Liquidity risk and valuation: Valuing and assessing risk in an illiquid market

by

Simen Eide

Thesis
for the degree of
Master of Science

(Master i modellering og dataanalyse)

Faculty of Mathematics and Natural Sciences
University of Oslo

November 2014

Faculty of Mathematics and Natural Sciences
University of Oslo

ABSTRACT

The object of the thesis is to investigate, measure and analyse the impact of liquidity on portfolio value, risk and execution.

We consider the formalism of [Acerbi and Scandolo, 2008] to value portfolios in markets exposed to illiquidity through the use of Marginal Supply Demand Curves. We show that future portfolio returns become fat-tailed when liquidity risk is introduced.

Further, we investigate the market impact model of [Almgren et al., 2005], who estimates supply curves on equity instruments by considering a large database of executed orders. Since such data are highly confidential, we propose to use transaction data to estimate the same supply curves. This may enable more market participants to assess their liquidity risks and costs. Transaction data does not contain the same information as order data. To bridge the information gap between the two data sets, we introduce a 'strategy identifier'. By using regression and filtering techniques we show that using transaction data together with the strategy identifier give results comparable to using order data.

Finally, we combine the formalism of [Acerbi and Scandolo, 2008] with the supply curves of [Almgren et al., 2005] and expand the notion of Marginal Supply Demand Curves to a stochastic object to model future liquidation prices. We find that a portfolio owner required to liquidate a large position will be faced with a trade off between liquidating fast to a high liquidity cost, versus liquidating over a longer time span but with higher market and liquidity risk.

ACKNOWLEDGEMENT

I would like to thank my supervisor Frank Proske for his guidance since I first came up with the idea. He embraced the idea, and has been enthusiastic with suggestions whenever we have met for discussions.

I also want to thank my colleagues at Oslo Clearing (now SIX X-clear) and my fellow students in B802 at the University of Oslo and at École Polytechnique Fédérale de Lausanne for giving both practical and theoretical feedback.

CONTENTS

1. BACKGROUND

The subject of liquidity risk has been absent in most mathematical finance literature. The main focus has relied on market risk, and extensive theories have been developed to explain the behaviour of the market dynamics. That may have been rightly: most of the financial risk stems from what we loosely call market risk, and the golden theory of mathematical finance, the Black&Scholes formula, manages to value options quite correctly without taking into account any liquidity risk.

However, liquidity risk is still an important contributing factor to financial risk. During the financial crisis, several institutions did not go bankrupt due to the market risk, but due to liquidity risk. The absorption of liquidity in the market decreased and they were forced to sell assets at large discounts to raise cash for their obligations. In general, a portfolio thought to be worth its mark-to-market value may be found to be worth significantly less when liquidating it in a market, and especially in a stressed market environment.

When speaking of liquidity risk we distinguish between three main sorts of risks:

- the risk of running out of money,
- the risk of trading in a illiquid market,
- the risk of decreased liquidity in the economy.

We will focus on the first two elements of liquidity risk, although the third is still relevant.

A promising approach is given by [Acerbi and Scandolo, 2008] and [Cetin et al., 2004]. Both articles questions the mark-to-market paradigm where prices are market to best bid or best ask. They argue that the effective price an investor will buy (sell) by closing down a large positon in a security will be higher (lower) than the observed best ask (bid). To correctly value a portfolio, the authors introduce supply and demand-curves for equities. [Acerbi and Scandolo, 2008] further assumes that a portfolio holder is held under some liquidity constraint, with a possible need to liquidate his portfolio for cash within a relatively short time horizon. The risk of having to liquidate a portfolio on short notice should clearly reduce the value of the portfolio. This is a reasonable assumption: a hedge fund may be confronted with a large unexpected withrawal and an insurance company may be confronted with a large unexpected insurance claim.

By assuming the existence of supply curves and a liquidity constraint, [Acerbi and Scandolo, 2008] shows that the value of a portfolio is now an optimization problem: Given that one may be required to raise liquidity in the portfolio, the investor would need to choose which securities to liquidate to minimize some 'liquidity cost'. Fortunately, [Acerbi and Scandolo, 2008] also shows that under reasonable and weak assumptions, finding the portfolio value can be formulated as a convex optimization problem.

The above framework builds upon the assumption of a well defined supply and demand curve. However, it makes no attempt to specify such curves other than specifying a large class of functions that will work.

[Gatheral, 2010] shows that if market impact is permanent, then no-dynamic-arbitrage[1] imposes that impact is symmetric between buy and sell.

[Almgren et al., 2005] estimates a model for the supply and demand curves from a large dataset from Citigroup's equity trading desk. They separate market impact into a permanent and a temporary element, and regress their data on historical buy and sell orders. Their findings is that the permanent impact is linear in number of shares, while the temporary impact follows a 3/5 power law.

The thesis is organized as follows: In Section 2 we review the formalism of [Acerbi and Scandolo, 2008] and include some proofs not present in the original work. In Section 3 we estimate Supply Curves through the use of transaction data from Oslo Børs over a period of about 2 years in some selected equity instruments. Section 4 combines the two above sections and considers how liquidity and market risk affect the portfolio value and execution horizon decision. Then, we give some general conclusive remarks in Section 5. The R code used to generate all the numerical results are found in the Appendix.

[1] Not possible to buy and sell over some time without a positive expected cost. See [Gatheral, 2010] for formal definition.

2. LIQUIDITY VALUATION

2.1 The Acerbi-Scandolo formalism

This section is a summary of the formalism put forward by [Acerbi and Scandolo, 2008] and introduces the most important concepts and results found in the mentioned article. The definitions, propositions and theorems will usually have their counterpart in [Acerbi and Scandolo, 2008], but we have added proofs and explanations where the original authors have omitted this.

The main concept to be understood is that there is no such thing as one price per asset. As observed in the market, a market participant will receive a higher price if he sells a small quantity of assets than if he sells a large quantity of assets. In general, the larger quantities of securities being traded by a participant the less favourable price will be given for the securities. This is characterized by the Marginal Supply Demand Curve (MSDC).

Definition 1 (Marginal Supply Demand Curve)**.** *An asset is a good traded in the market. The price of an asset at any point in time can be characterized by the Marginal Supply Demand Curve: a map $m : \Re \to \Re$ satisfying*

1. $m(x_2) \leq m(x_1)$ *if* $x_1 < x_2$
2. m *is cadlag for* $x < 0$ *and ladcag for* $x > 0$.

where x is the total amount of asset sold. $x < 0$ will be characterized as buying the asset.

The only general restriction on the prices is that you get less favourable prices as you buy or sell more of the asset.

A natural concequence of the definition of the MSDC's is that by selling or buying a quantity s of an asset, the monetary value gained equals the integral of m(x) from 0 to s. We will say that a market consists of one fully liquid asset we will call the Euro ($m_0 \equiv 1$), and N illiquid assets with MSDCs as in definition 1.

Definition 2. *Let a portfolio be defined as a vector $\mathbf{p} = (p_0, p_1, ..., p_N) \in \Re^{N+1}$ where p_0 is the position in the fully liquid asset, and p_i is the position in asset i. One may add a scalar to the portfolio vector $\mathbf{p} + c$, where $c \in \Re$. The meaning of such an operation will be the addition of euros to a given portfolio i.e. $\mathbf{p} + c = (p_0 + c, p_1, ..., p_N)$.*

After defining the market, we can define the first notion of a value of the portfolio as the value recieved if the portfolio was liquidated straight away:

Definition 3. *The Liquidation Mark-to-Market Value of a portfolio* $\mathbf{p}$ *is defined as*

$$L(\mathbf{p}) = \sum_{i=0}^{N} \int_0^{p_i} m_i(x)dx = p_0 + \sum_{i=1}^{N} \int_0^{p_i} m_i(x)dx \tag{2.1}$$

This definition is in contrast to the definition of a mark-to-market value of a portfolio when it is marked to the quoted bid/ask prices. Then, the portfolio is priced given prices of best bid ($m^+ = m(0^+) = lim_{x \searrow 0} m(x)$) and best ask ($m^- = m(0^-) = lim_{x \nearrow 0} m(x)$) prices.

Definition 4. *The Uppermost Mark-To-Market Value of a portfolio* $\mathbf{p}$ *is defined as*

$$U(\mathbf{p}) = p_0 + \sum_{i=1}^{N} [p_i m_i^- \mathbb{1}_{p_i>0} - p_i m_i^+ \mathbb{1}_{p_i<0}], \tag{2.2}$$

where $\mathbb{1}_x = 1$ *if* x *is true, and 0 otherwise.*

A natural definition of liquidity cost can then be defined as the difference between marking the portfolio to bid/ask prices, and liquidating the portfolio immidiately:

Definition 5. *The Liquidation Cost of a portfolio* $\mathbf{p}$ *is defined as*

$$C(\mathbf{p}) = U(\mathbf{p}) - L(\mathbf{p}) \tag{2.3}$$

It is easy to see that $C(\mathbf{p})$ is non-negative, and the natural interpretation is that a positive value is the cost of liquidation, measured in euro.

Theorem 1. *The functions L, U and C are continuous in P. They are also concave, concave and convex, respectively. Further, for* $\lambda \geq 1$

- $L(\lambda\mathbf{p}) \leq \lambda L(\mathbf{p})$
- $U(\lambda\mathbf{p}) = \lambda U(\mathbf{p})$
- $C(\lambda\mathbf{p}) \geq \lambda C(\mathbf{p})$

Proof. The continuity is easy to see from the definitions. We refer to [Acerbi and Scandolo, 2008] for proofs of the concavity and convexity. Hence we will consider the rescaling of the functions:

$$\begin{aligned} L(\lambda\mathbf{p}) &= \lambda p_0 + \sum_{i=1}^{N} \int_0^{\lambda p_i} m_i(x)dx \\ &= \lambda p_0 + \sum_{i=1}^{N} \int_0^{p_i} m_i(\lambda z)\lambda dz \\ &\leq \lambda p_0 + \lambda \sum_{i=1}^{N} \int_0^{p_i} m_i(z)dz = \lambda L(\mathbf{p}), \end{aligned}$$

where the equalities follow from the definition and transformation $x = \lambda z$, while the inequality follows from the monotonicity assumtion on m.

The scaling for U follows simply by linearity.

$$\begin{aligned} C(\lambda \mathbf{p}) &= U(\lambda \mathbf{p}) - L(\lambda \mathbf{p}) \\ &\geq \lambda U(\mathbf{p}) - \lambda L(\mathbf{p}) = \lambda C(\mathbf{p}), \end{aligned}$$

where we have used the property for U and L. □

We will call a portfolio $\mathbf{q}$ attainable from $\mathbf{p}$ if $\mathbf{q} = \mathbf{p} - \mathbf{r} + L(\mathbf{r})$ for some $\mathbf{r}$, and write $\mathbf{q} \in Att(\mathbf{p})$. Then the portfolios in $Att(\mathbf{p})$ are all portfolios that can be obtained from starting with $\mathbf{p}$ and liquidating some positions $\mathbf{r}$ in the market. Further, it can be shown by considering the derivatives of U that if $\mathbf{q} \in Att(\mathbf{p})$ then $U(\mathbf{q}) \leq U(\mathbf{p})$ (Lemma 4.11 in [Acerbi and Scandolo, 2008]).

One of the main innovations in [Acerbi and Scandolo, 2008] is the concept of a liquidity policy. It is a concept that is separate for each investor holding a portfolio, and can influence the value of a portfolio considerably as we will see.

Definition 6. *A liquidity policy $\mathcal{L}$ is a closed convex set of portfolios satisfying*

1. $\mathbf{p} \in \mathcal{L} \Rightarrow \mathbf{p} + a \in \mathcal{L}, a \geq 0$
2. $\mathbf{p} = (p_0, p) \in \mathcal{L} \Rightarrow (p_0, 0) \in \mathcal{L}$

Hence we can always add cash to the portfolio, and there is never any problem in reducing the position in a illiquid asset.

A liquidity policy should be understood as a requirement imposed on a portfolio by its owner's liquidity needs.

For instance, a hedge fund may be obliged to refund one of its investors within a short notice. We would then say that the hedge fund must have a portfolio $\mathbf{p}$ that can be liquidated to a portfolio $\mathbf{q}$ that satisfies a liquidity policy $\{p_0 \geq F\}$, where F is the investor's withrawal. Such a policy is called a cash liquidity policy:

Example 1 (Cash Liquidity Policy)**.** *A special type of liquidity policies is the cash liquidity policies $\mathcal{L}(a)$, defined as*

$$\mathcal{L}(a) = \{\mathbf{p} | p_0 > a\}, a \in \Re$$

It is clear that if $(p_0, p) \in \mathcal{L}$ and $b > 0$ then $(p_0, p) + b \in \mathcal{L}$ as $a < p_0 < p_0 + b$. Further, the cash liquidity policy does not depend on the position in any illiquid asset.

A portfolio is not meant to satisfy a liquidity policy at all times. For example, a hedge fund may be obliged to refund one of its investors on a very short notice, and hence will be required to find an amount of cash quickly through liquidation (a cash liquidity policy). Hence, we will require that a portfolio $\mathbf{p}$ can be liquidated into portfolio $\mathbf{q} \in \mathcal{L}(a)$ quickly. We will say that the value of the portfolio $\mathbf{p}$ for an investor with liquidity policy $\mathcal{L}$ is the best possible partly liquidation of $\mathbf{p}$ so that the new portfolio satisfies the liquidity policy. Formally,

Definition 7. *The value of a portfolio* $\mathbf{p}$ *under the liquidity policy* $\mathcal{L}$ *is the function* $V^{\mathcal{L}}$ *defined as*

$$V^{\mathcal{L}} = sup\{U(\mathbf{q}) | q \in Att(\mathbf{p}) \cap \mathcal{L}\} \tag{2.4}$$

Going back to our hedge fund as an example, it is not fair for the hedge fund management to announce that the value of their portfolio is 100m if they, after being forced to repay 1m, only can claim to be worth 95m. Seen this way, it is clear that the notion of value found in the mark-to-market value function can be improved. The mark-to-market function does not take any liquidity concerns when valuing the portfolio. The concept of liquidity policies takes liquidation and liquidity into account and incorporates it into the value function:

Corollary 1. *We can rewrite equation* (2.4) *in terms of the liquidating portfolio* $\mathbf{r}$*:*

$$V^{\mathcal{L}}(\mathbf{p}) = sup_r\{U(\mathbf{p} - \mathbf{r}) + L(\mathbf{r}) | r \in \mathcal{C}_{\mathcal{L}}(\mathbf{p})\} \tag{2.5}$$

where

$$\mathcal{C}_{\mathcal{L}}(\mathbf{p}) = \{\mathbf{r} | \mathbf{p} - \mathbf{r} + L(\mathbf{r}) \in \mathcal{L}\}$$

If $\mathcal{C}_{\mathcal{L}}(\mathbf{p}) = \emptyset$ *then we define* $V^{\mathcal{L}}(\mathbf{p}) = -\infty$*.*

A result of practical importance is also found in [Acerbi and Scandolo, 2008]. It gives us a straightforward recipe to value a portfolio constrained under a cash liquidity policy:

Theorem 2. *Let* $\mathcal{L}(a)$ *be a cash liquidity policy. Assume* $m_i(s)$ *are strictly decreasing and continuous on* $\Re$ *for all* $i = 1, .., n$*. Then there exists a unique solution* r^a *to the problem in equation* (2.5) *which is given by*

$$r_i^a = \begin{cases} m_i^{-1}(\frac{m_i(0)}{1+\lambda}) & \text{if } p_0 < a \\ 0 & \text{if } p_0 \geq a \end{cases} \tag{2.6}$$

where λ *is determined by*

$$L(\mathbf{r}^a) = a - p_0 \tag{2.7}$$

2.2 Summary of [Cetin et al., 2004]

[Cetin et al., 2004] defines an illiquid market in much the same ways that [Acerbi and Scandolo, 2008] does. Their market modeling assumptions are similar if not equal. [Cetin et al., 2004] goes on to use their market to hedge claims and places much stricter assumptions on their price processes than [Acerbi and Scandolo, 2008]. We will review the modeling assumptions made in [Cetin et al., 2004].

2.2.1 Comparison between assumptions of supply curves

In [Cetin et al., 2004] the authors explore hedging of claims in a market where prices are not only dependent on time and state of the world, but also on the traded amount x. The approach is also taken by [Acerbi and Scandolo, 2008], and although different notation, there are several similarities in the assumption taken on the price processes. In [Acerbi and Scandolo, 2008] one defines the Marginal Supply Demand Curves $m(s)$ which gives the marginal price of an asset given a sell ($s > 0$) or a buy ($s < 0$). On the other hand, [Cetin et al., 2004] models the Supply Demand curves $S(t, x, \omega)$ directly with respect to a buy ($x > 0$) or a sell ($x < 0$).

The transformation between the two notations is simply

$$S(t, x, \omega) = \frac{1}{x} \int_0^x m(-s)ds \tag{2.8}$$

and is also identified as what [Acerbi and Scandolo, 2008] calls the Supply-demand curve.

For comparison, we can transform the assumptions of the MSDCs found in 1 and 2 to find the equivalent properties for 2.8:

Theorem 3 (Acerbi Supply Curve)**.** *Assume that the MSDC satisfies the assumptions in 1, 2 and is restricted to be an asset. Then, a supply curve S_a defined as in 2.8 will satisfy the following properties:*

1. $S_a(x) \geq 0$,
2. $S_a(x)$ *is strictly increasing: if* $x_2 > x_1$ *then* $S_a(x_2) > S_a(x_1)$,
3. S_a *cadlag og lagcad,*
4. S_a *is continuous in x.*

Let us call functions satisfying the above conditions Acerbi Supply Curves $\mathcal{S}_a$.

Proof. The first property follows as m is assumed to be positive. The second property follows as m(-s) is strictly increasing in s: the average (from zero) of an increasing function is increasing. Third property follows from continuity of limit. Continuity of S follows from continuity of the integrand. □

Similarly, the assumptions on S in [Cetin et al., 2004] is:

Definition 8 (Cetin Supply Curve)**.** *A Cetin Supply Curve $S_c \in \mathcal{S}_c$ satisfies the following properties:*

1. *$S_c(t, x, \cdot)$ is $\mathcal{F}_t$-measureable and non-negative,*
2. *$x \to S_c(t, x, \omega)$ is non-decreasing in x a.s.,*
3. *S_c is C^2 in x, and its first two derivatives w.r.t. x are continous in t,*
4. *$S_c(\cdot, 0)$ is a semi-martingale,*
5. *$S_c(\cdot, x)$ has continuous paths for all x.*

Comparing S_a and S_c we can see that the two frameworks are similar. Both functions are increasing in x, although S_a is strictly increasing. Both functions are required to be positive. S_c is required to be C^2, a stronger assumption than cadlag og lagcad of S_a. Assumptions about the probability space is only loosely defined in [Acerbi and Scandolo, 2008].

The assumptions of [Cetin et al., 2004] regarding the supply curves are much stricter than the equivalent assumptions made by [Acerbi and Scandolo, 2008]. This is due to the fact that the former shall be used to develop hedging strategies. However, [Cetin et al., 2004] does not account for any liquidity policies in their valuations, a feature that may alter the end result of their hedging strategies. A possible investigation would be to see how the hedging strategy will work under the constraint of a liquidity policy.

2.3 Motivational example of portfolio value under exponential MSDCs

In this section we will present a simple model of the exponential type proposed in [Acerbi and Scandolo, 2008] where the MSDC of an asset i takes the form $m_i(x) = A_i e^{-k_i x}$. The parametric form of the MSDC is simple and tractable and there exists a closed-form solution of the value function, as shown in [Acerbi and Scandolo, 2008]. We will aim at showing some stylized facts about the model, by assuming stochastic dynamics on the variables A_i and k_i. The variable A_i will model the market volatility much in the same way that standard finance litterature models ut, while k_i will be our liquidity parameter. A large k_i will result in a less liquid asset, and the value of the position will be negatively related to the position size.

By using Proposition 6.1 in [Acerbi and Scandolo, 2008] we can show that the optimal solution for the exponential model is [1]

[1] [Acerbi and Scandolo, 2008] must have assumed $p_0 = 0$. We have included this term for completeness

$$\lambda = \frac{a - p_0}{\sum_{i=1}^{N} A_i/k_i - a + p_0} \tag{2.9}$$

$$r_i^a = \frac{1}{k_i} \log(1 + \lambda) \tag{2.10}$$

$$V^L a(\mathbf{p}) = U(\mathbf{p} - \mathbf{r}^a) - L(\mathbf{r}^a) = \sum_{i=1}^{N} A_i(p_i - r_i) + a + p_0 \tag{2.11}$$

There are several interesting points regarding the model:

- The model assumes no restrictions or costs on short selling.

- The optimal liquidation policy r^a is independent of the current position-vector (except p_0). This is a consequence of the no short selling restriction.

- The portfolio value is linear in each p_i, $i \neq 0$. This is because all position not liquidated to satisfy the cash policy is valued at the mark-to-market value.

Let us now introduce some simple stochastic elements in the model. Assume that we wish to evaluate a portfolio in a future time and that we know that the distribution of A and k are gaussian: $A \sim N(10, 0.04)$ and $k \sim N(0.001, 10^{-7})$. By simulating values of A and K we get a distribution of MSDC curves as seen in figure 2.1.

In 2.1, each line corresponds to the MSDC function with a different simulated (A, k) pair. The MSDC corresponding to the expected value of A and k is the thick black line. Due to our gaussian assumption on A there is no guarantee that the prices are positive. This is a limitation to the model, but by having sufficient low standard deviations on A we can ensure that the probability of negative prices is effectively zero.

Also note that the price has a significantly larger spread for buying positions ($pos < 0$) than for selling positions ($pos > 0$). The effect is due to the nature of the exponential function, and is in our model determined by the variable k.

We wish to investigate the value and risk of a portfolio in this model, and how it changes with different assumptions on A and k. Assume we have a portfolio consisting of a single asset. Following the idea of Acerbi (2007) we will consider four different scenarios:

1. A normal and $k = 0$. The gaussian model with no liquidity risk

2. A normal and $k \geq 0$ fixed. It is a model with static liquidity risk.

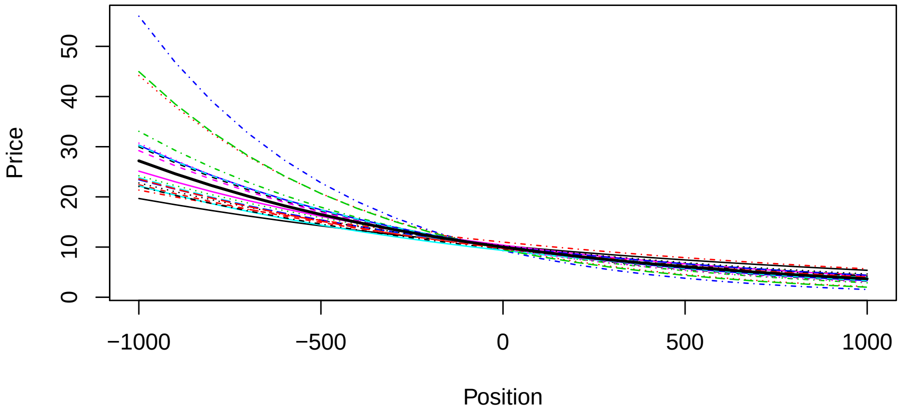


Fig. 2.1: Simulated MSDCs for different values of A and k. The black line is the mean MSDC.

3. A, k joint normal with zero correlation. i.e. it assumes independence of market and liquidity risk.

4. A and k joint normal with a negative correlation. The model assumes that as asset prices fall, liquidity is likely to decrease.

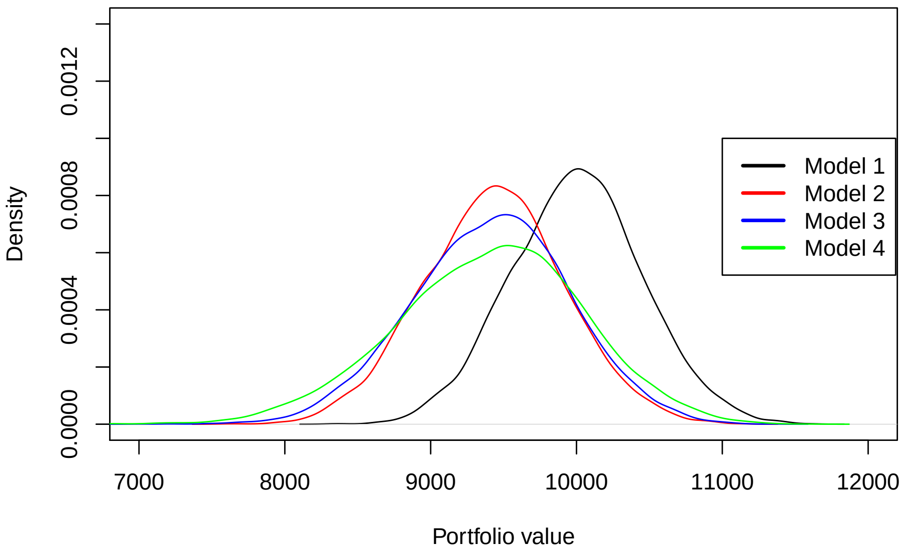
Assume parameters as given above. The uppermost mark-to-market value of this portfolio is simply $V(P) = m(0) * P = 10 * 1000 = 10000$. By performing a Monte Carlo simulation on the MSDCs with the parameters given above, we are able to approximate the distributions of the different models. The resulting distributions are plotted in 2.3.

The most clear result is that our first model is gaussian with mean at the uppermost value of the portfolio. By introducing liquidity risk in the form of a positive parameter k in model 2-4, the expected value of the portfolio decreases. In fact, it seems like model 2-4 have shifted their peak probabilities to a new point around 9500, which is the centre of the (A,k)-distribution (which is common for model 2-4). [2] The introduction of liquidity risk decreases the general value of the portfolio in this model, as should have been expected.

More interesting is the comparison between model 2-4. They are each special cases of the model $(A, k) \sim N(\mu, \Sigma)$, where μ and Σ are a general mean vector and covariance matrix. Comparing the empirical distributions, it is qualitatively clear that the fat tails increases from model 1 to 4. The fact can be accompanied by testing different risk measures on the portfolios.

[2] The value of a portfolio is 9 433 at the point $A = 10, k = 0.001$.

Fig. 2.2: Portfolio distributions of model 1 to model 4



In our Monte Carlo simulation, we can extract estimates of the standard deviation, 5 pct. Value at Risk and 5 pct. Expected Shortfall. Let us define these risk measures in the usual way and as the distance from the mean of the portfolio[3]. The results are found in table 2.1.

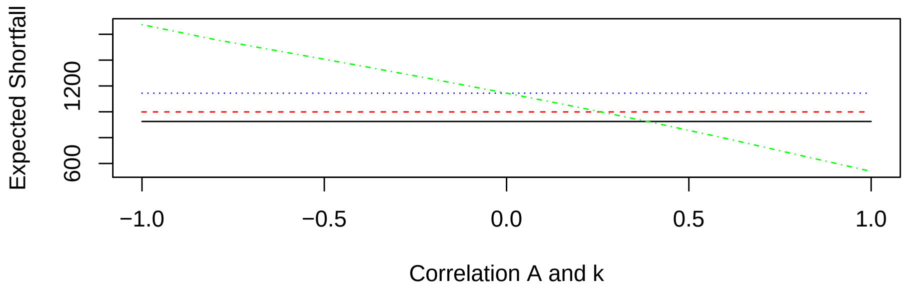
	Portfolio Value	Std.Dev	VaR	ES
Model 1	10003	453	743	925
Model 2	9434	486	800	999
Model 3	9413	541	913	1144
Model 4	9405	644	1100	1407

Tab. 2.1: Risk statistics for the four exponentially MSDC models that are being tested. Numbers denominated in cash value.

We see that with all risk measures, the risk increases from Model 1 to Model 4. Although introducing liquidity risk in Model 2 decreased the (average) value of the portfolio significantly, the model suggests that adding stochastic dynamics to the available liquidity has a significant impact on the risk measures used. Secondly, the correlation between the price variable A and the liquidity variable k is important, as seen from the differences between Models 3 and 4, which has a correlation of 0 and 0.5 respectively.

[3] Formally defined in chapter 3.

Fig. 2.3: How cor(A,k) affects the expected shortfall. We can see that only model 4 depends on the correlation between the two models, and that the portfolio risk is higher when there is a strong correlation between A and k. Model 1 (black), Model 2 (red), Model 3 (blue) and Model 4 (green).



The correlation effect is significant when executing the model with a correlation varying from -1 to 1, as seen in Figure 2.3, where we modified Model 4 and left the other model outputs with the original correlation assumptions.

There are several stylized results that can be drawn from the example above. We see that the value of the portfolio decreases as one introduces liquidity in the valuation model. The effect is an expected result of the new valuation model. However, it is not before we introduce stochastic liquidity in the later models that we see a large effect on the tails of the distributions. The portfolio becomes significantly more heavy-tailed when stochastic liquidity is introduced. In addition, a negative correlation between price return and liquidity has the largest effect on the heavy-tailedness.

The results motivate us to investigate stochastic liquidity-models where we try to estimate the size of the liquidity effect and impose reasonable assumptions on the distribution of the different market and liquidity variables.

3. MARKET IMPACT MODELLING

3.1 Overview Market Impact

The liquidity value model in Section 2 takes the Marginal Supply-Demand Curves (MSDC) as given in their valuation of asset portfolios. However, the forms of the MSDCs are of high importance in the models. This research area investigates the market impacts of agents executing large trading orders. The focus in the litterature is on supply/demand curves, which consider the average realized executed price instead of the marginal price studied in the previous section. There is a 1 to 1 relationship between the two, and we will bridge the notations in Section 4 when we combine the two research areas.

The literature is mainly focused on estimating expected market impacts as the data are highly noisy due to the more significant market volatility. Most established models make distinctions between temporary and permanent impact, as shown in the Deutsche Bank presentation [Ferraris, 2008]. The temporary impact happens as liquidity is taken out of the 'active' market, and the investor is temporary distorting the supply/demand equilibrium. However, the temporary impact will decay over time as the market restores itself to equilibrium. This is when the second impact effect will play a role: the permanent impact. It can be justified by an information argument, assuming that there are informed investors among the market noise: The execution of a large order provides new information to the market place, and the market will therefore establish at a new equilibrium. An illustration of the different impacts is shown in Figure 3.1.

[Gatheral, 2010] investigates "no-dynamic" arbitrage principles on a generalized class of different market impact models. The authors show that the permanent impact has to be linear in size and symmetric around 0 to satisfy some general no-arbitrage assumptions.

[Almgren et al., 2005] uses a permanent/temporary impact model where the permanent impact is linear and the temporary impact follows a 3/5-power law. It is a discrete model that considers each order as a separate event, as opposed to a continuous time model where one operates with a decay factor.

In this section we will base our analysis on the model suggested by [Almgren et al., 2005], and use various techniques to estimate their model to market data. The techniques, such as the strategy identifier and filtering theory, are generalizeable to other models.

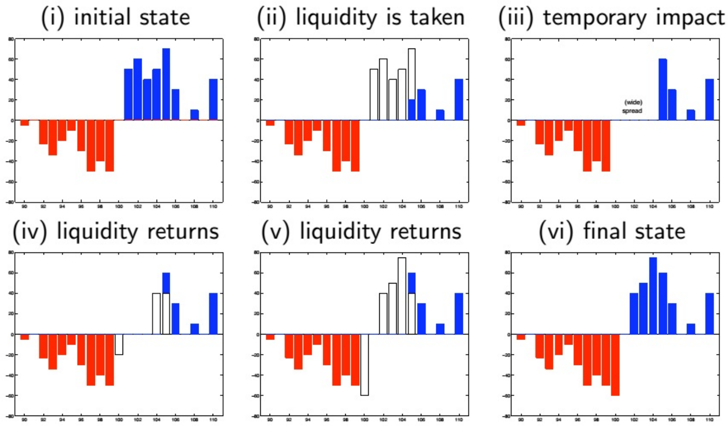


Fig. 3.1: Illustration of how an execution first drains the market of liquidity before the market again finds an new equilibrium different from the initial state. Borrowed from [Ferraris, 2008].

One of the main innovations in the section is the introduction of a 'strategy-identifier' described in Section 3.4. [Almgren et al., 2005] obtains their data from a large history of executed orders in Citibank. However, such data is not available to most market participants, and therefore limits the use of their analysis. The strategy-identifier tries to patch this gap by identifying trading patterns in transaction history that are more often publicly available. The remaining analysis shows that results comparable to those of [Almgren et al., 2005] can be obtained using this strategy-identifier.

3.1.1 Model selection

In testing the error of the various models, we will use the Mean Absolute Error (MAE) instead of the more widely used Root Mean Square Error (RMSE). There are arguments in favor of both estimators, see e.g. [Willmott and Matsuura, 2005] for arguments in favor of MAE as estimating average prediction power. We believe that in a highly noisy environment with extreme movements, RMSE would put an unfair high weight on the extreme "non-expected" events that we are not supposed to measure in the first place. Therefore, MAE is a more natural choice.

3.2 The Almgren-framework

[Almgren et al., 2005] quantifies market impact from large trades. They utilize a database from Citigroup, a large bank, containing equity trade orders between December 2001 and June 2003 (19 months). The authors postulates a model describing the effect on stock prices from a large trade. They distinguish between two types of market impact:

Definition 9 (Market Impact)**.** *Permanent (I) and realized (J) market impact are defined as follows:*

$$I = \frac{S_{post} - S_0}{S_0} \tag{3.1}$$

$$J = \frac{\bar{S} - S_0}{S_0}, \tag{3.2}$$

where S_{post} is the price observed 30 minutes after end of trading, S_0 is the price observed just before trading, and $\bar{S} := \frac{\sum_j x_j S_j}{\sum_j x_j}$ is the average price realized through trading x_j for price S_j for all transactions j in an order.

Further, as we will utilize and compare these functions over a longer time span, whenever we speak of the impact functions we will actually consider the volatility normalized impact functions $I \leftarrow \sigma I$ and $J \leftarrow \sigma J$, where σ is an estimate of the current volatility level.

Both of these variables can be observed in the market, and the hypothesis is that a large trade will affect the market in two ways: Firstly, it will temporarily affect the price as the agent demands liquidity over a (short) period of time. Secondly, a large order will distort the general view of the price and permanently affect the price.

[Almgren et al., 2005]'s model assumes that all orders are executed at a constant (normalized) trade rate v throughout the order's time span, and this affects the drift of the stock price. In order to normalize trades over a longer time span we will normalize the trade by the daily volume. Therefore, we will use $v := \frac{X}{VT}$ where X is the total order size, V is an estimate of the daily volume and T is the trade duration.

The dynamics of the stock price S_t is

$$dS_t = S_t g(v) d\tau + S_t \sigma dB_t \tag{3.3}$$

where v is defined as the normalized trade rate of the agent, g is the permanent impact function, τ is the volume time, B_t is a standard Brownian motion and the other variables are as usual.

In addition to a permanent price impact, [Almgren et al., 2005] assumes a separate temporary impact of the trade, giving an average price of execution slightly worse than an indenpendent observer would see in the market. The price received from our trades is assumed to be modeled as

$$\tilde{S}_\tau := S_\tau + S_0 h(X/T) \tag{3.4}$$

where $h(v)$ is the temporary impact function and S_τ is the process in Equation (3.3).

[Almgren et al., 2005] shows that in this framework, we can express the normalized impact functions in the following way:.

Modeling Assumption 1. *The normalized impact functions, defined by Definition 9, in the model described by Equations* (3.3) *and* (3.4) *can be expressed as*

$$I = \sigma T g(\frac{X}{VT}) + < noise > \tag{3.5}$$

$$J = \frac{I}{2} + \sigma h(\frac{X}{VT}) + < noise > \tag{3.6}$$

where
σ is the market volatility of the security, T is execution horizon, X is the trade size, V is daily market volume, g is the permanent market impact function, h is the temporary market impact function, and $< noise >$ is some white noise coming from market risk with zero expected value.

Further, [Almgren et al., 2005] proposes the analytic forms $g = \pm\gamma|v|^\alpha$ and $h = sgn(v)\eta|v|^\beta$. [1]

Proof. We will show how to obtain the permanent model. A proof of the temporary equation can be found in [Almgren et al., 2005].

By applying Ito's formula on equation 3.3 we get

$$S(\tau) = S_0 exp(g(v)\tau - \frac{1}{2}\sigma^2\tau + \sigma\beta_\tau)$$

where $t = \tau$ is the post-trade period and $t = 0$ is the start of the order period. Then, using definition 9,

$$\begin{aligned} I &= exp(g(v)\tau - \frac{1}{2}\sigma^2\tau + \sigma\beta_\tau) - 1 \\ &= exp(g(v)\tau) - 1 + exp(-\frac{1}{2}\sigma^2\tau + \sigma\beta_\tau) \\ &\approx g(v)\tau + exp(-\frac{1}{2}\sigma^2\tau + \sigma\beta_\tau) \end{aligned}$$

where we have used the approximation $exp(y) - 1 = y$ for small y in the last step.

We see that the last step is a so-called "noise"-element with zero expectation. To get the volatility-normalized version of the equation simply normalize the impact by $I \leftarrow \sigma I$. □

[1] Note that we have omitted the cross sectional term in g, as our analysis focuses at each stock separately for permanent impact.

By a Gauss-Newton optimization [Almgren et al., 2005] found the exponents to be $\alpha = 0.891 \pm 0.1$ and $\beta = 0.600 \pm 0.038$, and postulated that the exponents $\alpha = 1$ and $\beta = 3/5$ could be used in further analysis. We will (naively) adopt this assumption.

Then, [Almgren et al., 2005] regresses Equation (3.1) onto the models in Equation (3.5) and (3.6) to estimate γ and η. Their estimates are significant and has R^2 values of less than 1%. The explanatory power is expected to be low as the noise of the model is the market volatility. However, they show that the residuals are independent and have zero means. They do however have fatter tails than the assumed Gaussian distribution.

3.3 Strategy Identifyer

The largest advantage of the data set in [Almgren et al., 2005] compared to our data set is that the former has well identified orders with starting and ending time points. This enables the authors to know when an order starts and when it ends.

Unfortunately, large quantities of order data of this form are not publicly available. This causes limitations to the use of [Almgren et al., 2005]'s model, and one could argue that their model is based on a weak form of 'insider information'. For the results to be useful, it requires a practitioner either to blindly trust the model and parameters found, or to have access to large quantities of order data in a suitable form.

In comparison, our dataset contains only transaction information. Such data are often made available by the different exchanges on an exchange member level, and is in specific available for Oslo Børs. [2] Therefore, if one is able to use publicly available data to reproduce the results found in [Almgren et al., 2005], the approach would be far more promising.

The main difference between order data and transaction data is that in the latter one does not know when the agent started or stopped his order. Therefore, in order to identify orders we need to find out when an order starts and when it ends. We will propose one method which considers the largest difference in position of each agent per day and then identifies this as one order:

Definition 10 (Largest Daily Trade strategy)**.** *Let us assume that we can observe the position h_t of an agent in a particular stock over the trading day. We define the maximum and minimum position of the agent as*

$$h_{t_{max}} := max_t h_t$$
$$h_{t_{min}} := min_t h_t$$

Then, the trade order is identified as

[2] See e.g. "http://www.oslobors.no/markedsaktivitet/stockOverview?ticker=SEVDR" where "Kjøper" and "Selger" is identified per transaction for the stock SEVDR.

- *If $t_{min} < t_{max}$ we call the event a 'buy order' with size $h_{t_{max}} - h_{t_{min}}$.*
- *If $t_{min} > t_{max}$ we call the event a 'sell order' with size $h_{t_{min}} - h_{t_{max}}$.*

Further, if the trade size is below the trade size threshold $|h_{t_{min}} - h_{t_{max}}| < K$, we assume that there has been no trades for the given agent on the given day.

Our strategy identifier will capture large orders that an agent executes over the trading day. It also filters out any unwanted 'noise' on those days that the agent is not actively trading any large quantities by defining the trade size threshold K.

However, we stress that our strategy identifier in Definition 10 is a large simplification. A large order may be executed over several days, and our identifier will not capture this effect. Estimating the timing of large orders may also be problematic with this approach. Consider an agent who buys a large order at the start of the day, then does nothing until afternoon when a small end-client requests a small amount of shares. In this case, the timing of the order will be recorded to be the whole day, and our trade rate v will be significantly underestimated.

There are therefore many possible improvements to our simple identifier function. The main complication of using the model is to correctly set the trade size threshold K. Choosing that K is too low will include large quantities of noise in the model, while setting it too high will remove all observation points.

3.4 Initializing data for analysis

This section handles all the preprocessing of data needed to perform the analysis. It describes the raw data, discusses assumptions and approximations and establishes necessary time series to normalize the data such as volatility and volume estimates.[3]

3.4.1 Data description

The data set consist of transaction data from October 2010 to January 2014 (39 months) on Oslo Børs (OB). For each transaction, it identifies the time of trade, identification of stock traded, the buyer, the seller, the quantity of shares and price agreed. The buyers and sellers are identified by the members of Oslo Børs: brokerage firms and investment banks. In the dataset, there is no information of bid-ask prices or spreads.

In the analysis we will conduct the numerical study on three equities listed on Oslo Børs: Statoil (STL), Yara International (YAR) and Schibsted

[3] The source code of the data initialization is found in section .1.2.

(SCH). They are in decreasing market volume and from low volatility to high volatility stocks respectively.

The main advantages of utilizing transaction data is that they are public and that they are in vast quantities. Models are allowed to be more flexible and can be updated faster to new observations. In Figure 3.2 we show the cumulative positions of some of the exchange members in one stock over time, calculated from transaction data.

There are three main issues using transaction data instead of inside order-information:

1. there is no information whether a trade is buyer- or seller initiated,
2. there is no information when a trader starts and stops her trading strategy,
3. one looks at aggregated behavior over several agents within the exchange member, and
4. there is no observation of trades executed outside of the exchange: OTC trading, future markets, dark pools, MTFs[4], etc.

Issue 1 clouds our visibility of who hits each transaction. However, it should not have a large impact on our analysis. The importance is that the agent is buying/selling a significant amount, distorting the assumed equilibrium. Whether the agent actively places an order close to the market cross or hit the bid/ask should not alter the fact that he is demanding or supplying large quantities of securities. Considering issue 3, one could argue that one is not considering real agents. However, the rest of the market cannot observe the agent behind the exchange member. Their perception will be the same whether it was one agent that bought x number of shares or whether there were two agents under the same member that bought $x/2$ each.

Issue 2 poses a bigger problem, and one would have to look at the transaction data to observe trends. This issue is addressed by our strategy identifier described in Section 3.3. Issue 4 is a real problem that is a limiting factor in our analysis. A developed model that is used to take trading decisions should be expanded to these additional sources of market into account. However, this should not affect the subsequent analysis, and we will proceed as if we have all the information.

3.4.2 Raw data treatment and slot specification

An institution that is observed in the transaction data may execute a large number of transactions during very short periods of time. We are not interested in these micro level details, and we will therefore sample transaction

[4] MTF: Multilateral Trading Facilities. Alternative market places that may trade the same securities

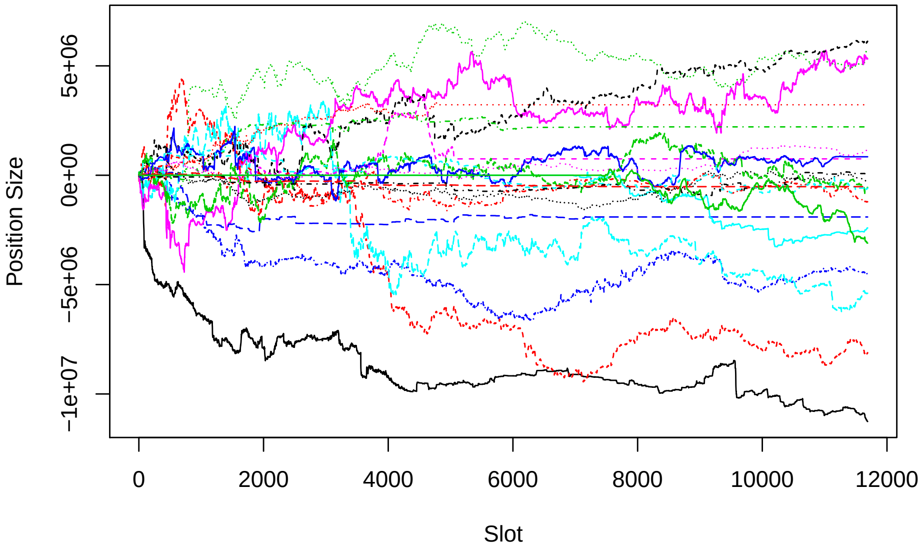


Fig. 3.2: The cumulative position for some of the exchange members in one stock over the observed slot periods. The position is calculated from transaction data and starts at zero.

data into 30 minute slots over the trading day: the first slot starting at 09.15-09.45 and the last slot being 15.45-16.15. Currently, the opening hours of Oslo Børs are 09.00-16.25, while the opening hours were 09.00-17.30 before 6. August 2012 [5]. This implies that we do not observe trading occurring in the opening and ending periods at Oslo Børs. This is reasonable, as trading in these time periods involves auctions which are significantly different in nature from the trading in between opening and close.

In addition to recording the transaction data we also collect supplementary data, such as volume and turnover traded in each time slot.

3.4.3 Time series related to stock: supportdata

In the framework given, we need to harvest various time series to explain the impact: volatility of the stock, the average volume traded per day, price of the stock and the average price each agent traded on in that time slot.

[5] http://www.oslobors.no/Oslo-Boers/Om-oss/Presserom/Nyheter-fra-Oslo-Boers/Oslo-Boers-reduserer-aapningstiden

Volatility

We apply an intraday volatility estimate per time slot. To estimate volatility, we use the Exponentially Weighted Moving Average (EWMA) estimator with $\lambda = 0.99$, as made popular by [J.P.Morgan, 1995]. Since we are discounting on a slot-to-slot basis, the daily λ value will be closer to 0.86. [6] Hence, our estimator is quickly adapting to new regimes and short periods of high volatility.

Our volatility estimate is as follows:

$$\sigma_k^2 = \lambda\sigma_{k-1}^2 + (1-\lambda)r_k^2 \tag{3.7}$$

where r_i is the log return at time k. The initial estimate is simply $\sigma_1^2 = r_1^2$. More complex initial values could have been assigned, but the estimates will normalize sufficiently fast.

Note that the volatility estimate is quite simple: it does not give any weights to the observed returns (by turnover, etc.), which would be a reasonable extension of the model.

To standardize the values, we scale the volatility estimate to a daily figure by multiplying σ by the square root of the number of slots in a day ($\sqrt{14}$).

Average daily volume

The daily volume estimate at timeslot k is estimated by the Exponentially Weighted Moving Average ([J.P.Morgan, 1995]) with $\lambda = 0.985$. By the same reasoning as in Section 3.4.3, it gives an effective daily λ of 0.78.

$$V_k = \lambda V_{k-1} + (1-\lambda)v_k \tag{3.8}$$

where v_k is the total number of shares traded in slot k.

To standardize the values, we scale the volume estimate to a daily figure by multiplying V by the number of slots in a day (14).

The volume estimator is simple. It does not account for e.g. systematic intraday differences.

Transaction price

We would like to estimate the realized price (3.4) an agent receives from trading in a time slot. Since we know both the agents' total value traded and the total number of shares traded in each slot, we can find the implied realized price by dividing the two quantities in each time slot for each agent.

The approach causes an issue for some time slots when an agent has a net change in holding of close to zero and has a profit or loss on its trading. It

[6] Found by solving $x : 0.99 ** 14 * 14 * 0.01 = x * (1-x)$, assumes constant signal in the slot-type estimator.

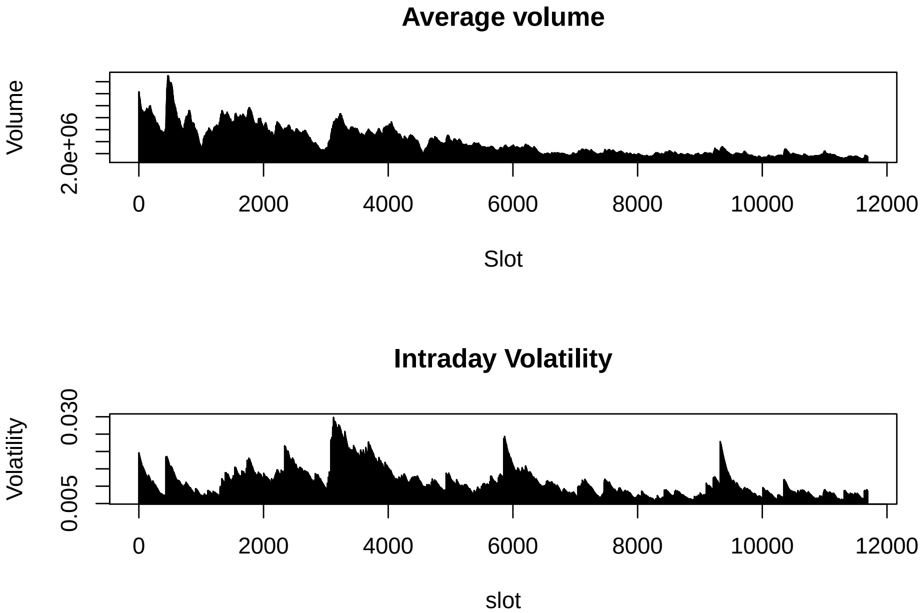


Fig. 3.3: Volume and Volatility estimators for the Statoil stock over time (measured by slots).

creates a delicate "divide-by-zero" problem which we will solve by removing all slot trades less than 5 shares in either direction. The tweak of the data will not affect the result substantially, as 10 shares has little impact on large order.

3.4.4 Strategy identifier: Transform transaction data to orders

We apply the strategy identifier in Definition 10 to all cumulative positions of all the identified agents in our dataset. The trade size threshold is determined in the data cleaning set in Section 3.4.5. The value is determined by first finding all trade strategies when $K = 1$, and then increasing K until only some percentage of the data is remaining.

As an example, if the position is 0 at time slot 1, 1000 at time slot 4 and -1000 at time slot 10, then the strategy is between 4 and 10 (identifying a sell order of 2000 shares).

In Figure 3.4 we can observe the accumulated positions of a member (bars). The grey vertical lines identify different days. The red lines identify the strategies. We can see that the upper plot gives several non-interesting strategies (which will fall below the trade size threshold), while the lower plot gives us something that looks like reasonable start and stop periods of a strategy.

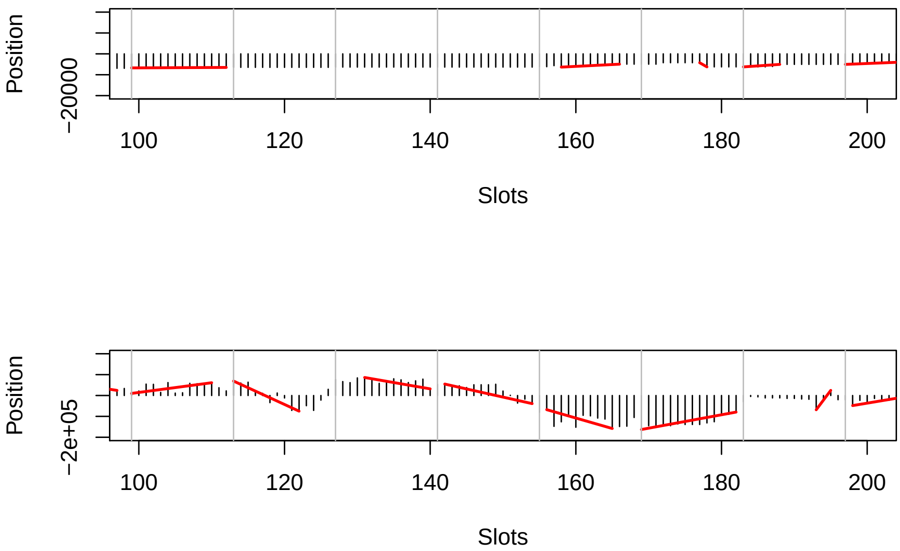


Fig. 3.4: Examples of identifying strategies. Positions over time and identified strategies (red). The vertical lines (grey) separate different days.

3.4.5 Data Cleaning

Before we can start to execute statistical analysis of obtained data, we need to limit our strategies. So far, we have included all strategies with a non-zero position. However, limiting both too small trades and outliers will be done before we start the data analysis.

Trade size Threshold

It is natural to expect that small position strategies will both be abundant and irrelevant to the analysis. The impact of small trades is expected to be mainly white noise, and thus the trades brings little new information to the market.

We will only conduct an analysis on the 6% largest strategies identified. As an example, this excludes roughly all trades that are less than 5% of the daily volume. The trade size threshold is arbitrary, but should not be so high that we do not have sufficient data to analyze. We have observed that the explanatory power of the below regressions is greater the higher threshold we set. This has an upper limit until the dataset is too scarce, and where noise becomes the dominant factor. Therefore, the choice of 6% was done as a trade-off between quality and quantity.

Outliers

We are attempting to model a process consisting of a large amount of noise. One should expect large outliers in the data set, and assume they are normal. However, extreme events in the dataset may just as likely be due to misspecifications and treatment of the data. After careful examination of the data, we have chosen some thresholds to remove the most unnatural data points.

We will remove some data points that have a very high 'order size-to-impact' ratio. Specifically, we will remove the 1% data points with highest 'order size-to-impact' ratio for both realized and permanent impact (0.5% of either side). The level was chosen by investigating the data and ensuring that only the very extreme points were removed. Note that neglecting outliers could accidentally remove tail events that one wants to measure.

The data cleaning above was done using all the raw data each time. The same data point is likely to be in both of the cleaning criteria. For example, a large order with very little impact may be in both outlier categories.

3.5 Regression

After identifying the investment strategies in Section 3.2 we want to estimate the liquidity gradients η and γ in (3.5) and (3.6). Assuming that these parameters are time-independent, we may use a regression approach. The permanent and temporary impact are regressed using the data obtained in Section 3.4. In the following section we will use the analysis obtained from STL to examine the regression results, unless otherwise stated. Then, we will present the main statistics using the same techniques examine the other equity instruments [7].

3.5.1 Parametric Regression

Loyal to the model specified in Section 3.2, we regress the impact functions to find the liquidity gradients η and γ. To test the predictability of the models, the data has been randomly grouped into training- and test data, where 70% is categorized as training data. Then, the liquidity gradients η and γ were estimated using the training data, and predicted on the input variables in the test data. The prediction power is benchmarked against the zero-hypothesis $\{\eta = 0, \gamma = 0\}$, and mean absolute errors of the models are computed relative to the null hypothesis.

$$\mathrm{MAE} = \frac{\sum_{i=1}^{N} |\text{predicted Impact}_i - \text{Observed Impact}_i|}{\sum_{i=1}^{N} |\text{Observed Impact}_i|} \tag{3.9}$$

for all test data 1 to N.

[7] The source code used to produce the results in this section is found in section .1.3.

```
##
## Call:
## lm(formula = I ~ I(XV * sigma) - 1, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.08134 -0.00937  0.00059  0.01107  0.05610
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## I(XV * sigma)    1.578      0.458    3.45  0.00058 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0195 on 1338 degrees of freedom
## Multiple R-squared:  0.00881,Adjusted R-squared:  0.00807
## F-statistic: 11.9 on 1 and 1338 DF,  p-value: 0.000581
```

Fig. 3.5: Regression output in R by regressing the training data onto the permanent model in Equation 3.5 for Statoil.

Test/Train data

We divide the data randomly into a training set and a test set, giving a 70%. probability that a data point is categorized as training data. By fitting the impact models to the training data and then testing the error on the test data, the approach should give us a good unbiased estimate of the error of the model.

The permanent model regression on Statoil reported in Table 3.5.1 shows that the regression analysis estimates a positive coefficient, as expected. The estimate is about two standard deviation of the standard error, and is weakly 'significantly different' from zero. The regression explains around 1% of the deviance in the data set.

The temporary impact regression on Statoil reported in Table 3.5.1 shows that there is a significant positive coefficient estimate, and that the R^2 value is around 0.6%.

When testing the model on the test data, the explained deviance naturally decreases. We will benchmark the mean squared errors (MAE) of the model with the null model as described in 3.9. Hence, our MAE ratios are 0.997 and 0.995 for the permanent and temporary models, respectively. Both the permanent and temporary models seem to have R^2 values between 0.5% to 1.0%. The results are in line with what is found by [Almgren et al., 2005].

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## I(J - I/2) ~ h_dat - 1
##
## Parametric coefficients:
##       Estimate Std. Error t value Pr(>|t|)
## h_dat   0.1806     0.0652    2.77   0.0057 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## R-sq.(adj) =  0.00533   Deviance explained = 0.565%
## GCV = 7.5478e-05  Scale est. = 7.5422e-05  n = 1352
```

Fig. 3.6: Regression output in R by regressing the Statoil data onto the temporarly model in Equation 3.6. Note that the variable h_{dat} is the parameterized function h.

Similar results are found for YAR and SCH, as shown in Table 3.1 and 3.2.

	gamma	Stddev	R-sq	MAE
STL	1.580	0.460	0.009	0.997
YAR	1.410	0.440	0.008	0.994
SCH	0.230	0.210	0.002	0.999

Tab. 3.1: Regression coefficients from training data on permanent impact.

We see that both STL and YAR have better out-of-sample performance than SCH. The main reason of this is probably the availability of data. STL and YAR have significantly more trading than SCH over the same time horizon. This makes the SCH analysis very prone to noise.

The data seems to be too volatile relative to the amount of data collected to get reliable liquidity gradient estimates when dividing into training and test data. By changing the seed of the train/test separation sampling, different data points will be assigned to train and test. If there is a sufficient degree of randomness in the data set, it may not be possible to correctly estimate a prediction error: The observations assigned to the less abundant test set may not display the underlying trend of the data. In such a case we may have calibrated a good predictive model, but will not be able to realize when gauge the model against the 'flawed' test data. To show the

	eta	Stddev	R-sq	MAE
STL	0.180	0.070	0.005	0.995
YAR	0.100	0.050	0.003	0.996
SCH	0.100	0.040	0.006	1.007

Tab. 3.2: Regression coefficients from training data on temporary impact.

effect, we have changed the seed of the permanent impact model of STL. As seen in Table 3.3 the estimated liquidity gradients spans over a wide range, depending on which values go into the training data. For example, seed 5 seems to estimate half the gradient of the other seeds. Similarly, the MAE ratio changes significantly in the different runs, ranging from 99.4% of the null model for the "best" prediction to as poor as the null model.

	gradient	stddev	MAE
1	0.519	0.454	0.997
2	0.539	0.455	0.996
3	0.165	0.472	0.999
4	1.099	0.450	0.996
5	1.319	0.444	1.001
6	1.016	0.445	0.994

Tab. 3.3: Liquidity gradients with different seeds

Similar results are found for the temporary impact and for the other equity instruments. The results indicate that the data set is too small to accurately both predict and estimate the prediction error using a test/training method. Further analysis of changing seeds would be similar to the approach of cross validation, and we propose this as a good method to estimate the prediction error.

Cross validation

By applying cross validation to estimate our prediction error, we are using a 10-fold approach where all data is randomly divided into 10 groups. For each group, we train a model using all data except the data points in this group. Then, the prediction error is benchmarked to the null model. [8] The cross validated coefficients for STL have mean estimates of 1.066 and 0.208 for γ and η.

The prediction and error analysis of the 10-fold cross validation method can be seen in Table 3.4 and 3.5.

When employing the algorithm, we tested different seeds to create different groupings. The gradient coefficients and prediction errors variability

[8] For more information on cross validation, see [Hastie et al., 2009].

	gamma	MAE
STL	1.070	0.994
YAR	0.920	0.995
SCH	0.180	0.998

Tab. 3.4: Regression coefficients from cross validation on permanent impact

	eta	MAE
STL	0.210	0.994
YAR	0.140	0.996
SCH	0.050	1.001

Tab. 3.5: Regression coefficients from cross validation on temporary impact

obtained in 3.5.1 were not found with the cross validated approach. This suggests that the cross validated figures are more reliable. We see that the models can explain about 0.5% of the market noise for STL and YAR. The SCH data seems to be too scarce to predict a good model, even with a cross validation approach.

Estimating Temporary Impact across stocks

One may assume that temporary impact is not stock specific, but equal for all stocks. Then one may regress (3.6) across all three datasets: STL, YAR and SCH. The approach allows for a larger dataset, but may decrease the precision since there may be stock specific features of the temporary impact. However, the permanent impact I will still be stock-dependent, as we regress on $J - \frac{1}{2}I$.

	Estimate	Std. Error	t value	Pr(>|t|)
h_dat	0.0979	0.0239	4.09	0.0000

Tab. 3.6: Temporary regression on data from all stocks

In (3.6) we estimated η by standard regression. We see that the regression identifies a significant positive coefficient.

If we employ the same 10-fold cross validation approach to estimate the errors, the estimated model has an out-of-sample mean absolute error of 0.997 compared to the null model. Mean absolute errors on each of the individual stocks are reported in Table 3.7.

Compared to the results in Section 3.5.1 the common temporary impact model is an improvement for SCH, no change for YAR and a worsening for STL. The common temporarly impact model seems to have predictive power for all equities, while the former only managed to predict the temporarly liquidity gradients of STL and YAR with reasonable precision.

	ticker	MAE
1	STL	0.996
2	YAR	0.996
3	SCH	0.997

Tab. 3.7: MAE using a common temporary impact model

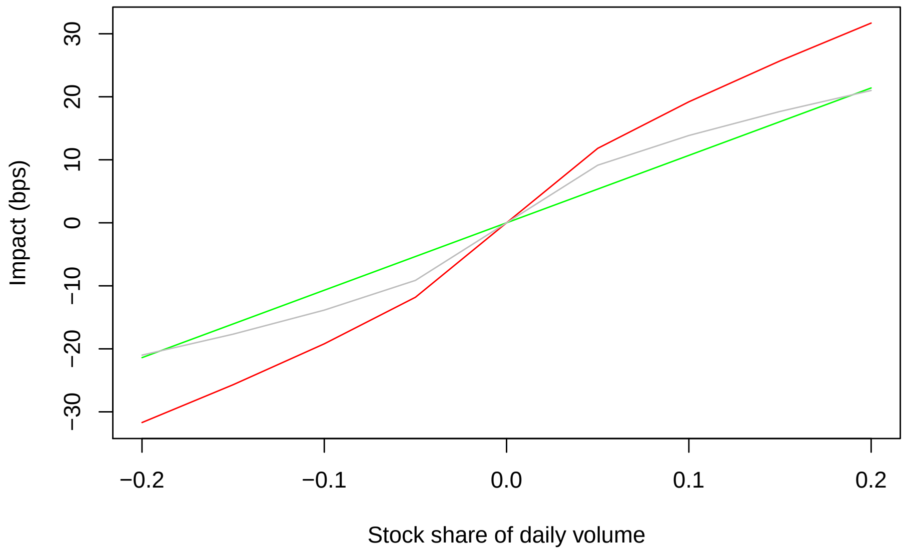


Fig. 3.7: Liquidity Impacts for STL. Realized (red), Permanent (green), Temporary (grey), 10 pct. volatility

Example of Impact costs

Given daily volatility of 10% and a trading time of 0.2 days and the results found above we calculate the expected liquidity impacts on STL, as seen in Figure 3.7.

We see that by buying 10% of the daily volume the expected price increase is about 20 bps. This is a small price difference, but in line with what [Almgren et al., 2005] and other literature found. [Almgren et al., 2005] estimate that the realized cost of buying 10% over 0.2 days of IBM and DRI is about 25 bps. and 43 bps.

In [Ferraris, 2008] the authors investigate their own market impact model over an order database from January 2007 to March 2008. They find that the realized impact ranges from -30 bps to around 90 bps with volumes ranging from 0.05% of daily volume to 10% of daily volume. Although difficult

	XV	Ibp	Jbp
1	-0.20	-21.40	-31.70
2	-0.15	-16.05	-25.70
3	-0.10	-10.70	-19.20
4	-0.05	-5.35	-11.82
5	0.00	0.00	0.00
6	0.05	5.35	11.82
7	0.10	10.70	19.20
8	0.15	16.05	25.70
9	0.20	21.40	31.70

Tab. 3.8: Impact as a function of order size

to compare, these figures seems to be on the upper end of our findings. However, this should be expected as they operated in a more volatile market environment.

3.5.2 Smoothing regression

The section above considers a specific, parametric form of the Supply Curve. The analytically tractability of such models is desireable, but limits the possible prediction power of the data. In this section, we will investigate non-parametric forms of the supply curves. The basis will be the set of Generalized Additive Models, where we smooth the explanatory variables with thin plate splines. Thin plate splines smooths a d-dimensional surface by minimizing the distance plus a penalizing term given by the square of the second order derivatives. The smoothing factor will be found using cross validation within the training set.

We will perform the analysis using the temporary impact of STL as an example. Two non-parametric models were regressed against $J - \frac{1}{2}I$:

- A 1 dimensional thin plate spline of the original covariate $\frac{X}{VT}$, and

- A 4 dimensional thin plate spline using all our relevant covariates: Position size (X), Daily Traded Volume (V), daily volatility (σ) and execution time (τ).

Using a train/test approach, as described in Section 3.5.1, we find the out-of-sample prediction power in Table 3.5.2. We see that the original linear model have the same prediction power as the 1 dimensional smooth model with the same input variable variable. The four dimensional smoothing model seems to give an improved prediction power, at the expense of being a significantly more complicated model to work with outside the scope of prediction. In addition, the four dimensional smoothing model bears the risk

Model	MAE
Original	0.997
1-dim Smooth	0.997
4-dim Smooth	0.990

Tab. 3.9: Out of Sample Mean absolute errors on test data for the one dimensional smoothing model and the four dimensional smoothing model when predicting temporary impact on STL.

of attaching non-explainable effects to the calibrating values of the model. It has a significant smoothing term and a high in-sample R^2 value, as seen in the following R summary:

```
Family: gaussian
Link function: identity

Formula:
I(J - I/2) ~ s(X, V, sigma, tau, bs = "tp")

Parametric coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.0006495  0.0002230  -2.913  0.00365 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
                   edf Ref.df     F p-value
s(X,V,sigma,tau) 60.42  77.37 3.265  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =   0.15   Deviance explained = 18.8%
GCV = 7.0031e-05  Scale est. = 6.683e-05  n = 1344
```

However, the train/test approach suffers, as we demonstrated in Section 3.5.1, from unstable results. Therefore, we conduct a 10-fold cross validation to get a more relieable estimate of the prediction power of the four dimensional smoothing model.

The out-of-sample MAE ratio on the four dimensional smoothing model were found to be 99.1%. This indicates that the four dimensional smoothing model outperforms our original linear model.

The result indicates that the 3/5 power law described in Model 1 is ill-posed, and that there may exist functional forms that performs better. Finding a functional form is important, as the four dimensional smoothing model is intuitively complicated and difficult to work with when confronted with follow up calculations such as combining it with the formalism put forward by [Acerbi and Scandolo, 2008].

3.5.3 Conclusions and improvements on regressions

We have conducted regression analysis of the models postulated in Model 1. All regressions predict positive liquidity coefficients for both temporary and permanent impact. The liquidity gradients are in the same order of magnitude as [Almgren et al., 2005] and [Ferraris, 2008] finds in their analysis of the Citigroup and Deutsche Bank order data, respectively. The results therefore suggest that the strategy identifier defined in Section 3.3 is a good bridge from transaction data to a full order history, which the original model was calibrated to.

However, analysis of the train/test data shows that there are high uncertainties in the results. Therefore, a careful cross validation is done to increase the use of the data and reliability of the results. Also, a simplified version of the temporary model is designed which does not distinguish between the different stocks, but regresses the temporary model on all three stocks. The out-of-sample MAE ratios of this model performs averagely well among the three other models. Since the simplified parametric model has less assumptions on the temporarily impact (no assumptions that the impact is different), it may be correct to conclude that the stock-independent temporary impact model is more suitable. Stock specific elements should then be captured in the permanent impact model.

A non-parametric regression analysis using a multivariate smoothing spline outperform the simpler parametric models. This suggests that the form of our impact functions can be improved by changing the impact functions g and h. However, due to the difficulty in handling such smoothed functions analytically in future analysis, we will keep to the more analytically tractable Model 1.

There are several possible improvements of the model. For the above analysis, we have chosen to keep each stock separate for the permanent impact. This may be one of the main drawbacks, as it reduces the possible scale of the dataset by a third. By including the 'inverse turnover'-variable in [Almgren et al., 2005] a higher precision model could possibly be created.

We have assumed the same impact functions g and h as [Almgren et al., 2005] in the analysis. Specificly, we did not investigate the exponents found in [Almgren et al., 2005], $\alpha = 1$ and $\beta = 3/5$. These exponents were found using different equity instruments in different markets. The non-parametric regression analysis in Section 3.5.2 suggests that other forms and exponents should be investigated. Implementing an algorithm to estimate the exponents could give different estimates and improved out-of-sample results.

The 'simplest' solution to any regression problem with large quantities of noise is to use more data. It is also reasonable that if more data is available, either by an extension in time period or inclusion of alternative market places or the futures market, prediction power would increase.

There are several ways of buying exposure to a stock, and a natural extension of the above analysis would be to include the futures and forward market. Buying a futures would give you the same exposure as buying the underlying asset, while at the same time not necessarily stress the underlying market. Care has to be taken to examine how buying a future contract affects liquidity. If you are buying from a market maker that would hedge himself in the spot market, little is gained by adding this liquidity.

Including more market places that trades the same equities would improve the dataset. Oslo Børs does not have market monopoly on its instruments, and STL and YAR are traded on alternative exchanges. Harmonizing trade data from the different exchanges and aggregating it on the same brokers is an obvious path to obtain better data.

However, caution must be taken, as stocks are not stationary companies but evolve over time. In the regression analysis we have assumed that $\gamma_t = \gamma$ and $\eta_t = \eta$ are time independent. It is reasonable to assume that a company evolves with time: the project risks a company is exposed to and the risk factors determining its stock price may be completely different after some time. For example, the daily volume of STL on Oslo Børs has decreased from 10m shares in 2010 to about 2m shares in 2013. Also, in 2011 Statoil bought Brigham Explorations for \$4.4 billion, about 5% of the value of Statoil, to exploit oil shale in North Dakota. These are factors that make it difficult to model Statoil as the same company over time. A significant source of error could therefore be time dependency. Classical regression approaches do not accommodate for time dependency easily, and alternative solutions should be evaluated. Using time series analysis is a possible approach and state space models could possibly be used to estimate the liquidity gradient coefficients.

In addition, using randomized train/test data in a regression distort the prediction power of the model. The test data are evenly spread out over the investigated time period, and will be predicted using data points obser ed in time. Therefore, the information in the test data may have already been given to the model.

Finally, a relevant point is that we have not been able to compare the results in this section with order data from the same stocks in the same period. Including this would the the proper test to see whether it is possible to use transactional data for liquidity cost analysis.

3.6 Liquidity Gradients via filtering theory

3.6.1 The Kalman Filter

State-space models were first introduced by [Kalman, 1960] in aerospace-related research. The models estimate an unobservable stochastic underlying process from noisy observations of a different process. The liquidity gradients can be considered to be such unobservable processes. The given observations are equity returns which include a large amount of market noise.

Among state space models there are both discrete and continuous methods. As the current data is a discrete and finite set of orders, we choose to focus on a discrete Kalman filter to model the liquidity gradients. Further, we use a time-dependent version of the model.

Let us first define some necessary assumptions and notation:

Definition 11 (State Space)**.** *We wish to produce estimators of the unobserved signal $x_t \in \mathbb{R}^p$ given some observations $Y_{t-1} = \{y_1, ..., y_t - 1\}$ where $y_s \in \mathbb{R}^q$ is an observable vector. We assume that the dynamics of x_t and y_t are*

$$x_t = x_{t-1} + w_t \tag{3.10}$$

$$y_t = A_t x_t + v_t, \tag{3.11}$$

where A_t is a $q \times p$ matrix giving a linear map on how y_t is affected by x_t,
$w_t \sim N(0, Q_t)$ is the random noise of the unobservable process x_t with zero-mean and covariance matrix Q_t, and
$v_t \sim N(0, R_t)$ is the noise of the observable process y_t with zero-mean and covariance matrix R_t.

Furthermore, we define

$$x_t^s := E(x_t | Y_s) \tag{3.12}$$

$$P_t^s := E[(x_t - x_t^s)(x_t - x_t^s)^\top], \tag{3.13}$$

where $s \leq t$.

Then, we can state the main result:

Algorithm 1 (The Kalman Filter)**.** *Given the state space in Definition 11, with initial conditions $x_0^0 \in \mathbb{R}^p$, $P_0^0 \in \mathbb{R}^p$ and covariance matrices $R_t \in \mathbb{R}^{q \times q}$,$Q_t = Q_0 \in \mathbb{R}^{p \times p}$. Then, the minimum mean-squared error[9] of all linear*

[9] See [Shumway and Stoffer, 2011] for a formal definition

estimators is found by applying the following algorithm: for $t \geq 1$*:*
Prediction step:

$$x_t^{t-1} = x_{t-1}^{t-1}$$
$$P_t^{t-1} = P_{t-1}^{t-1} + Q_t$$

Filter step:

$$x_t^t = x_t^{t-1} + K_t \epsilon_t$$
$$P_t^t = [I - K_t A_t] P_t^{t-1},$$

where

$K_t = P_t^{t-1} A_t' [\Sigma_t]^{-1}$ *is called the Kalman Gain,*
$\epsilon_t = y_t - A_t x_t^{t-1}$ *is the prediction errors (innovations), and*
$\Sigma_t := var(\epsilon_t) = A_t P_t^{t-1} A_t' + R_t$ *is the covariance matrix of the prediction errors*

A proof of the optimality of the algorithm can be found in [Shumway and Stoffer, 2011].

3.6.2 A Kalman Filter for a market impact model

Assume we have historical orders executed over a time period between 1 to T. We want to estimate unobservable liquidity gradients on permanent and temporary impact. By constructing the problem as a filtering problem we may extract the liquidity gradients as a time series, which can incorporate changes in impact over a period of time.

We assume that the permanent liquidity gradient γ and the temporary liquidity gradient η have the dynamics

$$\gamma_t = \gamma_{t-1} + \epsilon_t^{\gamma}$$
$$\eta_t = \eta_{t-1} + \epsilon_t^{\eta},$$

where ϵ_t^{γ} and ϵ_t^{η} and zero-mean gaussian noise. Then, we can define a two dimensional column vector

$$z_t := \begin{bmatrix} \gamma_t \\ \eta_t \end{bmatrix} = z_{t-1} + w_t,$$

where $\begin{bmatrix} w_t^{\gamma} \\ w_t^{\eta} \end{bmatrix} := w_t \sim N(0, Q)$. The first element of z_t is the permanent liquidity gradient, while the last element of z_t corresponds to the temporary liquidity gradient.

Further, we can write the model $y_t := \begin{bmatrix} I_t \\ J_t - 0.5 I_t \end{bmatrix}$ as

$$y_t = A_t z_t + v_t \tag{3.14}$$

where

$$A_t := \sigma_t \begin{bmatrix} \frac{x_t}{V_t} \\ \mathrm{sgn}(x_t) T_t |\frac{x_t}{V_t T_t}|^{\frac{3}{5}} \end{bmatrix}$$

$$v_t \in \mathbb{R}^q \sim N(0, R_t) \text{ and}$$

$$R_t = \sqrt{T_t} \begin{bmatrix} R_{11}, R_{12} \\ R_{12}, R_{22} \end{bmatrix} \in \mathbb{R}^{2\times 2}$$

Using the notation above, we can implement the Kalman Filter in Algorithm 1, repeating all the observations $t = 1, ..., T$ to estimate a time-series of the liquidity gradients z_t.

3.6.3 Implementing the Kalman Filter

We investigate three stocks over 835 observation days. One issue that arises in the implementation of the Kalman Filter is that there may be several orders starting in the same slot, across all participants and stocks. Therefore, if there are several orders on the same slot we choose to use the largest trade, on the assumption that it contains more information than the smaller order. The data extracted in Section 3.2 is mapped to the variables as defined in Section 3.6.2.

Initial parameters

An extensive attempt of fitting the maximum likelihood initial parameters using the whole path of the Kalman Filter was done by implementing a modified Newton Method. However, the algorithm tended to fail, or the variables tended to explode in value. For example, when setting R_0 and Q_0 by maximum likelihood, the algorithm needed to ensure that both matrices were positive definite. As there is no such mechanism in the Newton Method, the algorithm used the method of [Higham, 2002] to ensure we were handling a positive definite matrix. However, the inclusion of such steps altered the stepwise modification and was not helpful in the Newton Method. Also, as we observed that the developing algorithm predicted unrealistic high volatility parameters, boundaries were imposed. However, such boundaries only resulted in the algorithm finding the optimum at the boundary.

The initial parameters were instead set by considering the results found in the regression section, and the uncertainty parameters were set to be relatively stable for the whole observation period.

Temporary impact

We estimate the temporary impact time series using Algorithm 1 [10], with modification of $z_t = \eta_t$. Data from all three stocks is used as demonstrated in Section 3.5.1, and sorted by the starting slot. If there are more than one order starting in the same time slot, we only consider the largest order, measured by the assumed temporary impact ($\sigma \frac{X}{VT}$). The method will thus filter out only the most dominant orders observed at the same time horizon, and we rely on the assumption that there are more information in larger trades.

The results show the temporary liquidity gradient to be in the same range as found in Section 3.5, varying around 0.1 . The predicted standard deviations $\sqrt{P_t^s}$ are on average 0.04, and show a decreasing trend from the initial value of 0.044 to 0.036. This gives a weak predictor with a large uncertainty, as seen in Figure 3.8, but the model suggests that the variable is non-zero by a 95% confidence level most of the time.

We calculate the mean absolute error by comparing the filter model prediction at time t (x_t^{t-1}) against the observed temporary impact at time t. The Mean Absolute Error ratio for the temporary filter model across all stocks is 0.992 compared to the null model. The figure is well below the MAE value of 0.997 found by the equivalent regression model in Section 3.5.1. The deviation could be due to the decrease in the number of observations in the dataset when requiring only one observation per slot. In fact, by running the Kalman filter naively on the larger dataset, the filter model seems to perform as well as the time independent regression model.

An interesting feature of the filter model is how it apparently is calibrating itself to better performance over time. By dividing the prediction into four equally large groups we find that the MAE is higher in the first period than the succeeding three. In the first period, the filter model performs worse than the null model. In the succeeding periods, the filter model is performing better, an indication that the parameters are calibrating during the first period.

Based on the above analysis, we conclude that it is appropriate to model the liquidity gradient as a time-dependent state space variable. Although

[10] Source code found in Appendix .1.4.

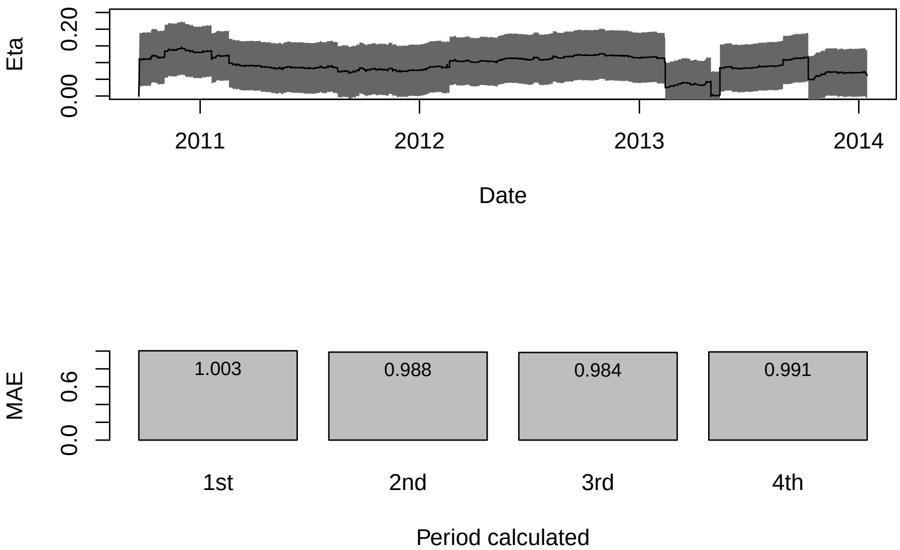


Fig. 3.8: Above: Estimate of temporary liquidity gradient. The grey area is the 95 pct. confidence interval. Below: Mean absolute error (MAE) in different time intervals. The filter model seems to improve as it recalibrates its own values.

the filter model is not outperforming the regression model overall, it seems to calibrate itself and explains a somewhat increasing amount of data in the later time periods. The model also predicts non-zero temporary liquidity gradients at a 95% confidence level over the time period.

3.6.4 Possible improvements of the implemented Kalman Filter

Many of the arguments put forward for the regression analysis are also relevant for the Kalman Filter. In addition, we have some filter-relevant issues that will be dicussed in this section.

The initial values of the filter play a large role in the estimation. As an example, when the above implementation was run with a different initial z_0 the filter did not adapt quickly to the 'correct' level. This is due to the assumed volatility parameters of z_0 in Q, which restrict how fast the parameters can adapt. By choosing initial parameters more carefully, for instance by improving the method outlined in Section 3.6.3, a model with better prediction power may be obtained.

The implemented Kalman Filter is a discrete filter which is updated whenever a significantly large trade occurs. The discrete version of the Kalman Filter was chosen because the underlying permanent/temporary impact model by [Almgren et al., 2005] gives discrete observations. However, if a continuous market impact model was chosen such as those described by [Gatheral, 2010], a continuous filtering technique, described in e.g. [Øksendal, 2010], could be used. Non-linear filtering techniques may also be used to take into account the results found in Section 3.5.2.

It is natural to assume that the change in liquidity among securities have a common driver. Hence, if one expands the variable z_t to include more stocks one could assume or estimate correlations between the liquidity gradients of the different stocks. Then, by assuming or estimating correlations between the stocks one can get an updated state variable of stock A by observing a signal on stock B. Such a noise reduction technique may improve the filter.

4. RISK MANAGEMENT IN ILLIQUID MARKETS

In Section 2 we investigated how to value a portfolio under liquidity constraints. Taking some marginal supply and demand curves (MSDCs) as input, we found that the actual value of a portfolio held by an owner with some liquidity constraint is lower than the mark-to-market price. In Section 3 we estimated some supply-demand curves from market data. In this Section we will show the one-to-one relationship between the supply-demand curves and the MSDCs. During the analysis in Section 3, we observed both a large amount of noise and an indication that the liquidity gradients are stochastic themselves. We combine the work done in the previous sections and define Stochastic Marginal Suppy Demand Curves (SMSDC) that is used to evaluate both market and liquidity risk in portfolios. Further, we investigate how the introduction of liquidity risk and execution horizon affects the portfolio value distribution. An implementation of the model is performed and numerically investigated. [1]

4.1 Coherent measures of risk

The concept of a coherent risk measure was introduced by [Artzner, 2009]. They proposed four axioms that any risk measure should satisfy for all types of financial risk. A risk measure should be a real-valued function mapping a random variable representing the portfolio value distribution to a unit stating a financial loss in a given reference instrument (e.g. Euro). Loosely speaking, we should be able to say that "under risk measure ρ the risk of portfolio X is ten million Euros". For simplicity and without lack of generality, we will call the reference instrument "Euro".

Definition 12 (Coherent Measure of Risk)**.** *Let X, Y be random variables representing portfolio values. Then, a real valued function $\rho : \mathbb{M} \to \Re$ is called a coherent risk measure if it satisfies four axioms:*

(T) Translational Invariance $\rho(X + e) = \rho(X) - e$

(M) Monotonicity $\rho(X) \leq \rho(Y)$ *if* $X \geq Y$

(S) Subadditivity $\rho(X + Y) \leq \rho(X) + \rho(Y)$

[1] The source code of this section can be found in Appendix .1.5

(PH) Positive homogeneity $\rho(\lambda X) = \lambda\rho(X)$

The first two axioms are not controversial and straight forward intuitive: The Translational Invariance property (TI) states that if you add Euro to your portfolio, the risk decreases linearly. Monotonicity (M) assumes that if portfolio Y is worth less than portfolio X, then the risk in portfolio Y is higher.

Subadditivity (S) is also intuitive, but not that easy to fulfill. It states that for two portfolios, the risk of the portfolios separately will be higher than if the risk of the two together. It captures the well accepted diversification effect. However, the widely used Value at Risk measure does not fulfill (S) at all times.

Positive Homogeneity (PH) states that if you increase the portfolio value by a factor, the risk of the portfolio will increase by the same factor. The axiom has been widely debated when considering liquidity and concentration risk. The argument is that risk may not be linear in position: if a position in an asset is doubled, the risk will be more than doubled as the market will be impacted when liquidating the position.

The argument has given rise to a weakened set of risk axioms, called 'convex risk measures' and proposed by e.g. [Follmer and Schied, 2002]. The convex risk measures replace (PH) and (S) with a convexity requirement (C) on the risk measure:

$$\rho(\theta X + (1-\theta)Y) \leq \theta\rho(X) + (1-\theta)\rho(Y) \tag{4.1}$$

where $\theta \in (0, 1)$.

A convex risk measure still takes into account diversification, as a diversified portfolio $\rho(\theta X+(1-\theta)Y)$ will have lower risk than the two separately. However, it does not guarantee that the risk will increase linearly with an increase in position.

[Acerbi and Scandolo, 2008] counter this weakening by arguing that it is not the risk measure axioms that are wrong, but merely the notion of value. They argue that the real deficiency is not due to the coherent risk measure, but the notion of value. In the same way the critics of coherent risk measures argue that portfolio risk is not linear in position, [Acerbi and Scandolo, 2008] argue that it is the portfolio value that is not linear in position.

It is common to assume a linear relationship between position and value as in Definition 3. However, [Acerbi and Scandolo, 2008] argue in favor of a portfolio value under a liquidity policy, which we introduced in Section 2.1. They conclude that the risk of a portfolio is linear in portfolio value. The non-linearity must come from the value function.

We will focus on two examples of risk measures: The Value at Risk and the Expected Shortfall. Value at Risk is a widely used risk measure in the industry first introduced by [J.P.Morgan, 1995].

Definition 13 (Value at Risk)**.** *Given a random value X representing the portfolio value, a probability measure P and a quantile $\alpha \in (0,1)$, the Value at Risk is defined as*

$$VaR_\alpha(X) := -F^{\leftarrow}(\alpha) \tag{4.2}$$

where $F_X^{\leftarrow}(p) := inf\{x|F(X) \geq p\}$ is the inverse of the cumulative distribution funtion $F_X(x) := P(X < x)$.

Value at Risk is an intuitive risk measure which is simply the quantile of the portfolio value distribution. It is therefore easy to coin the measure as "the largest loss that can occur with a certainty of 99%". The measure is widely used and popular among regulators. However, the measure fails to fulfill the subadditivity axiom, and is therefore not a coherent risk measure.

Remark 1. *Value at Risk is not a coherent risk measure.*

Proof. We will show this by counterexample. Consider two assets X, Y independent and identical distributed that are subject to occasional, independent shocks:

$$X = \begin{cases} 0 & \text{with probability } 0.991 \\ -1 & \text{with probability } 0.009 \end{cases} \tag{4.3}$$

Then, $VaR_{1\%}(X) = 0$. However, the probability that $X + Y = -1$ is

$$\begin{aligned} P(X + Y = -1) &= P(X = 0, Y = -1) + P(X = -1, Y = 0) \\ &= 0.991 * 0.009 + 0.009 * 0.991 \\ &= 0.018 \end{aligned}$$

so $VaR_{1\%}(X + Y) = -1$. □

Definition 14 (Expected Shortfall)**.** *Given a random value X representing the portfolio value, a probability measure P and a quantile $\alpha \in (0,1)$, the Expected Shortfall is defined as*

$$ES_\alpha(X) = -\frac{1}{\alpha}\int_0^\alpha F_X^{\leftarrow}(p)dp \tag{4.4}$$

where $F_X^{\leftarrow}(p) := inf\{x|F(X) \geq p\}$ is the inverse of the cumulative distribution funtion $F_X(x) := P(X < x)$.

Remark 2. *Expected shortfall is a coherent risk measure.*

Proof. **Translational Invariance:** Assume we have $e \in \Re$. First observe that $F_{X+e}(x) = P(X + e < x) = P(X < x + e) = F_X(x - e)$. We also have that

$$\begin{aligned} F_{X+e}^{\leftarrow}(\alpha) &= inf\{y|F_{X+e}(y) \geq \alpha\} \\ &= inf\{y|F_X(y - e) \geq \alpha\} \\ &= F_X(\alpha) + e \end{aligned}$$

Where the last step follows from substitution $x = y - e$ and the linearity of inf. With these results, it is easy to see that

$$\begin{aligned} ES_\alpha(X+e) &= -\frac{1}{\alpha}\int_0^\alpha F^{\leftarrow}_{X+e}(p)dp \\ &= -\frac{1}{\alpha}\int_0^\alpha F^{\leftarrow}_{X}(p) + edp \\ &= ES_\alpha(X) - e \end{aligned}$$

Monotonicity: First we observe that if $X \geq Y a.s.$ then $F_Y(z) \geq F_X(z) \forall z$ and by taking the inverse we have that $F^{\leftarrow}_Y(p) \leq F^{\leftarrow}_X(p) \forall p \in (0,1)$.
Then,

$$\begin{aligned} ES_\alpha(X) &= -\frac{1}{\alpha}\int_0^\alpha F^{\leftarrow}_{X}(p)dp \\ &\leq -\frac{1}{\alpha}\int_0^\alpha F^{\leftarrow}_{Y}(p)dp = ES_\alpha(Y) \end{aligned}$$

Subadditivity: Subadditivity can be shown by using an finite order statistic and taking the limit as in [Alexander J. McNeil, 2006] or by a more direct version as in [Acerbi and Tasche, 2002]. Both methods require some more notational work, and we omit it here.

Positive homogenity: Let $\lambda \in \Re$. Observe that

$$\begin{aligned} F^{\leftarrow}_{X\lambda}(\alpha) &= inf\{z|P(X\lambda \leq z) < \alpha\} \\ &= inf\{z|P(X \leq \frac{z}{\lambda}) < \alpha\} \\ &= \lambda F^{\leftarrow}_X(\alpha) \end{aligned}$$

Then it follows that

$$\begin{aligned} ES_\alpha(X\lambda) &= -\frac{1}{\alpha}\int_0^\alpha F^{\leftarrow}_{X\lambda}(p)dp \\ &= -\frac{1}{\alpha}\int_0^\alpha \lambda F^{\leftarrow}_{X}(p)dp \\ &= \lambda ES_\alpha(X) \end{aligned}$$

$\square$

4.2 Liquidity risk under illiquid pricing

In [Acerbi and Scandolo, 2008], the value of a portfolio depends on the liquidation prices when a partial liquidation of the portfolio occurs. The authors assumes an immediate liquidation given the respective MSDCs. However, immediate liquidation is only possible if you liquidate to the prices present in the order book of the exchange. In reality, portfolios are liquidated over some time period often ranging from a few minutes to a full day. During this time, the portfolio is exposed to market and liquidity risk. These risks can be modeled as a Stochastic Marginal Supply Demand Curve (SMSDC, Definition 15). The value concept of Section 2 should therefore be extended to a function of the Stochastic Marginal Supply Demand Curves. It implies that the value of a portfolio is no longer deterministic.

In the following section we will first bridge the notation of the MSDC and the Supply curves used. Then, we will define the SMSDC. Using an example we investigate the value distribution of a portfolio. Finally, we investigate how the liquidation horizon affects the expected value and the risk in a portfolio.

4.2.1 Bridging notation between the Marginal Supply Curves and the Supply Curves

We have used the concept of Marginal Supply Demand curves in Section 2, which is different from the more familiar Supply Curve that we have used in Section 3. The MSDC can be seen as the derivative of the total cost of an order, while the supply curves give the average cost. In this Section we will bridge the notational gap between the MSDC and the Supply Curve.

Assuming the framework in Section 3.2 we obtain the following model by substituting Equation 3.5 and 3.6 into Equation 3.1:

Modeling Assumption 2. *The realized price of an order of X shares executed over a period of length τ is given by*

$$\bar{S}(X) = S_0(1 + \frac{1}{2}\gamma\sigma\frac{X}{V} + \eta\sigma sgn(X)|\frac{X}{V\tau}|^{3/5} + \sigma\epsilon), \tag{4.5}$$

where S_0 is the price before the order, γ and η are given liquidity gradients, V is the daily volume traded, σ is the daily volatility and ϵ is some white noise with mean zero (standardized by the return volatility).

We have reduced market risk into the single variable ϵ. However, as we have seen in Section 3, it is the main driver of $\bar{S}$, and should not be ignored.

Modeling Assumption 3. *The realized price of an order in Model 1 can be expressed as a Marginal Supply Demand Curve with respect to an order size $s \neq 0$:*

$$m(s) = S_0 + S_0\epsilon\sigma - sgn(s)S_0\sigma(\gamma\frac{|s|}{V} + \frac{8}{5}\eta|\frac{s}{V\tau}|^{3/5}), \tag{4.6}$$

where s is the amount sold ($s < 0$ is buying), $\tau > 0$ is the execution horizon, S_0 is the price before the order, γ and η are given liquidity gradients, ϵ is the market risk varaible, V is the daily volume traded, and σ is the daily volatility.

To validate the model above we need ensure that the following statements are true: The above formula is correct and that $m(s)$ is a MSDC according to Definition 1.

By assuming that $x_2 > x_1$ it can be shown that $s \to m(s)$ is a decreasing function:

$$\begin{aligned} m(x_1) - m(x_2) =& S_0\sigma(\frac{\gamma}{|V|}(\text{sgn}(x_2)|x_2| - \text{sgn}(x_1)|x_1|) \\ &+ \frac{\eta}{|V\tau|^{3/5}}\frac{8}{5}((\text{sgn}(x_2)(|x_2|)^{3/5} - \text{sgn}(x_1)|x_1|^{3/5}))) \end{aligned}$$

is positive for all combinations $x_2 > x_1 > 0$, $x_2 > 0 > x_1$ and $0 > x_2 > x_1$. Since $m(s)$ is also continuous, $m(s)$ satisfies the definition.

The cost of buying X shares is $C(X) = \bar{S}(X)X$. First, we substitute $X = -s$ to obtain the same sign for buy and sell as in Section 2. Then, the cost function reads $C(s) = \bar{S}(-s)|-s|$. The total cost function is the infinite sum over all prices from 0 to X. Hence, the Marginal Supply Demand Curve is

$$\begin{aligned} m(s) &= \frac{d}{ds}C(s) \\ &= \frac{d}{ds}S_0|s| + S_0|s|\sigma\epsilon - S_0\sigma\text{sgn}(s)(\frac{1}{2}\gamma\frac{s^2}{V} + \eta|\frac{s^{8/5}}{V\tau^{3/5}}| \\ &= S_0 + S_0\sigma\epsilon - \text{sgn}(s)S_0\sigma(\gamma\frac{|s|}{V} + \frac{8}{5}\eta|\frac{s}{V\tau}|^{3/5}) \end{aligned}$$

as required.

Then, by applying Theorem 2 to the MSDC and using the results in Theorem 2 and Equation (4.7) we obtain an example of an explicit solution for the value of a portfolio:

Remark 3. *Assume there exist $i = 1, ..., n$ MSDCs $m_i(s)$ in the form of Equation (4.6). Given a cash liquidity policy $\mathcal{L}(a)$ there exists a unique solution to the problem*

$$V^{\mathcal{L}}(\mathbf{p}) = sup\{U(\mathbf{p} - \mathbf{r}) + L(\mathbf{r}) | r \in \mathcal{C}_{\mathcal{L}}(\mathbf{p})\} \tag{4.7}$$

which is given by

$$r_i^a = \begin{cases} m_i^{-1}(\frac{m_i(0)}{1+\lambda}) & \text{if } p_0 < a \\ 0 & \text{if } p_0 \geq a, \end{cases} \tag{4.8}$$

where λ is determined by

$$L(\mathbf{r}^a) = a - p_0 \tag{4.9}$$

and m_i is given by Equation (4.6).

There is no explicit solution of the inverse of Equation (4.6), but as we have shown it is one-to-one and can therefore be inverted numerically.

Solving Equation (4.7) requires us to implement an algorithm performing two nested root findings: First we need to find the inverse of Equation (4.6), and secondly we need to solve for λ in Equation (4.9). The power of the above result is that we now can value a portfolio with a cash constraint over some short time horizon τ for any market- and liquidity conditions (η, γ, ϵ).

An investor's free variable, given a large market order, is the liquidation horizon he execute his order over. His natural aim is to have the largest possible value of his portfolio after the execution, and he may wish to maximize the expected value of his post-trade portfolio. Considering liquidity alone, the best strategy would be to execute the trade over a long period of time, reducing the market impact. At the same time the investor is exposed to market risk, which, if assuming a standard Brownian motion, is proportional to the square root of time. There is a trade-off between market risk and available liquidity in the market.

Example 2 (One stock universe with cash liquidity policies)**.** *Assume that the value of a portfolio is given by Equation* (4.7)*. Consider an investor that may be required to fund NOK X m. on short notice. If there are no short-selling restrictions, we can for simplicity assume that he has no open positions in the market, and will short-sell to fund his liquidity needs. This is not a restrictive assumption: As seen in Remark 3, the optimal liquidation strategy* $\mathbf{r}$ *is independent of the investor's current positions* $\mathbf{p}$*.*

We will assume that there exists one stock in the market, and use calibrated values from Section 3 for the stock STL. Using Remark 3 we can find portfolio values given different cash liquidity policies. In Figure 4.1 we can see the effect on an increasing cash liquidity policy. When the cash liquidity policy is close to zero the portfolio value is close to zero, as expected (mark-to-market value of a zero position). As the cash liquidity policy increases, the portfolio value decreases concavely. This is due to the fact that the investor's market impact increases as he is selling larger quantities of the stock to fund his liquidity needs. We can also observe that the numerical first derivative of the portfolio value is linear.

The effect can also be seen in Figure 4.2, which shows how the liquidation strategy $\mathbf{r}$ *(amount of stocks sold in STL in our case) depends on the liquidity policy. Clearly, when* a *increases, so does* $\mathbf{r}$*. In addition,* $\mathbf{r}$ *has an increasingly positive first derivative with respect to* a*. When the total cash liquidity policy is low, the investor is selling close to the mark-to-market price. However, for large cash liquidity policies he is selling at a significantly lower price and has to sell more stocks for the same amount of liquidity (in our example approximately 4% more stocks than at the mark-to-market price).*

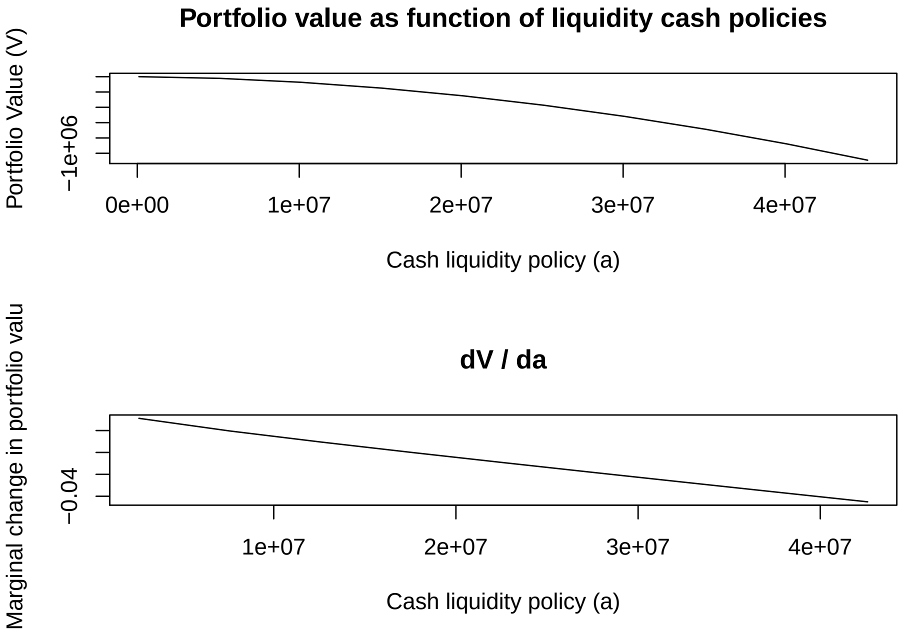


Fig. 4.1: Portfolio value and incremental increase in optimal liquidation policy given different cash liquidity policies

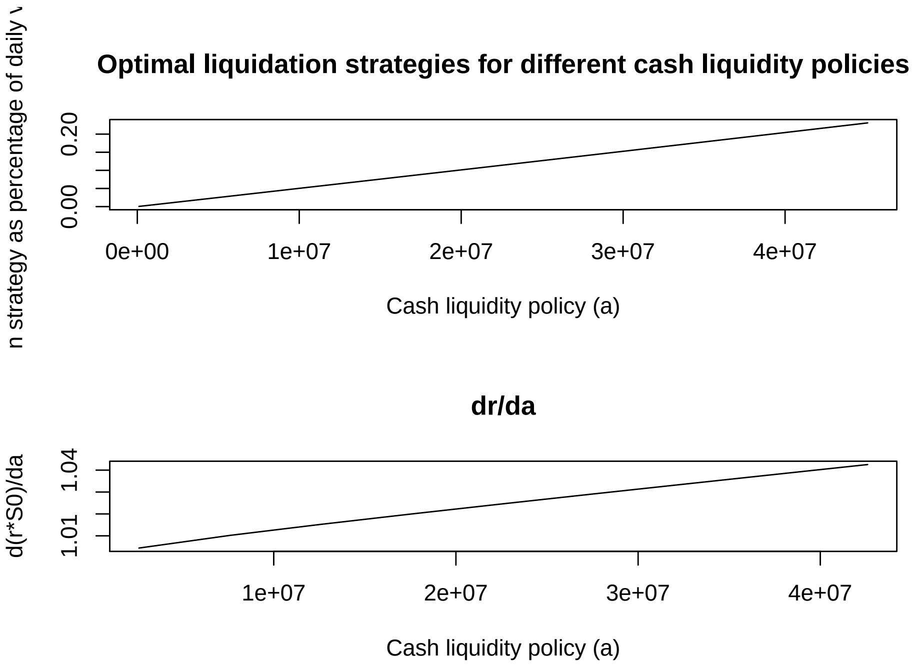


Fig. 4.2: Incremental extra need of liquidating stocks given incremental increase in cash liq. policy

4.2.2 Stochastic modeling of market- and liquidity risk

So far we have considered the deterministic dynamics of the value function. The Supply Curves in Section 3 assume stochastic variables in the price level return ϵ and the liquidity gradients η,γ. They also require the owner of a portfolio to liquidate any position over an execution horizon τ. By assigning stochastic dynamics to the risk factors in the MSDC function, we can find the portfolio value distribution for any execution horizon.

Definition 15 (Stochastic Marginal Supply Demand Curve)**.** *Assume that $\chi_t = (\eta_t, \gamma_t, \epsilon_t)$ is a stochastic process ($\chi_0 = (0,0,0)$) defined on a probability space $(\Omega, \mathcal{F}, P)$, where the process χ_t is $\mathcal{F}_t$ adapted and $\mathcal{F}_t$ its natural filtration.*

We will assume that χ_t is a diffusion of the form

$$d\chi_t = AdB_t \tag{4.10}$$

where B_t is an m-dimensional Brownian motion and A is a $3 \times m$ matrix. Note that A determines both the level of volatility of the liquidity itself and the correlation among the χ_t variables.

Then, we can define a Stochastic Marginal Supply Demand Curve (SMSDC)

$$m(s,\tau) = S_0 + S_0\epsilon_\tau\sigma - sgn(s)S_0\sigma(\gamma_\tau\frac{|s|}{V} + \frac{8}{5}\eta_\tau|\frac{s}{V\tau}|^{3/5}), \tag{4.11}$$

where s is the amount sold ($s < 0$ is buying), S_0 is the price before the order, ϵ_τ, γ_τ and η_τ are stochastic processes determining market and liquidity risk, V is the daily volume traded, and σ is the daily volatility.

We have emphasized the execution time τ in the function, as it is a variable that the agent has the power to control.

The model is a Gaussian model where $\chi_t = AB_t$. Hence, the volatility of the process increases by a factor of $\sqrt{t}$ over time. An example of a SMSDC can be seen in Figure 4.3.

We are assuming that the liquidity gradients are adaptable, but this should probably be relaxed in an extended model. It is clear from Section 3 that it is questionable whether the liquidity gradient processes are observable. Also, the simple Gaussian model in Equation (4.10) can be extended to include mean reversion by using an Ornstein-Uhlenbeck process or similar. We have purposely avoided the drift in Equation (4.10) as we believe it is of lesser importance. This could also be extended in a future model.

We will investigate the impact on both liquidity valuation and liquidity risk for the overall risk of the portfolio by an example.

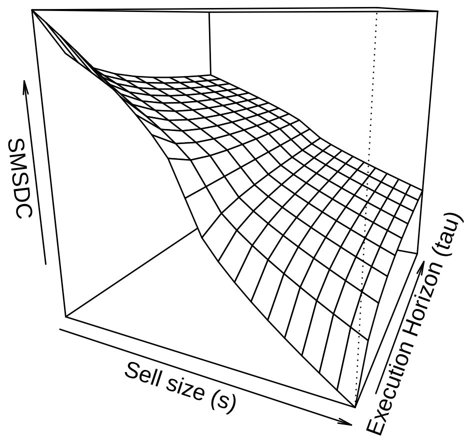


Fig. 4.3: Example of an expected Stochastic Marginal Supply-Demand Curve. It can be seen as a function of both trade size and execution horizon. Note that for the same trade size, the prices are more extreme for short execution horizons

Example 3 (Distributional properties of an illiquid portfolio)**.** *Assume η, γ and ϵ follow the dynamics in Equation* (4.10)*. Then, given the filtration at time t,*

$$\chi_t := (\eta_t, \gamma_t, \epsilon_t)^T \tag{4.12}$$

$$\chi_{t+1} - \chi_t \simeq N(0, \Sigma_t) \tag{4.13}$$

where Σ_t is a suitable covariance matrix.

The portfolio holds initially no positions, but has the possibility of trading in one security. It also has a cash liquidity policy of 10% of the daily market volume of the security.

We will test four different models, similar to what we did for the exponential MSDC model:

1. *$\eta = \gamma = 0$. The Gaussian price model with no liquidity risk.*
2. *$\chi_{t+1} > 0$ but the covariance matrix has zero standard deviation for the liquidity gradients. It is a model with constant liquidity.*
3. *$\chi_{t+1} > 0$ and Σ_t is a diagonal matrix. The model assumes independence between market and liquidity risk.*
4. *$\chi_{t+1} > 0$,*

$$cor(\eta_{t+1}, \epsilon_{t+1}) = -0.5,$$
$$cor(\gamma_{t+1}, \epsilon_{t+1}) = -0.5, \text{ and}$$
$$cor(\eta_{t+1}, \gamma_{t+1}) = 0.5$$

   The model assumes negative correlation between the market risk and liquidity risk: As asset prices fall, liquidity is likely to reduce.

We perform a Monte Carlo simulation to find the value distribution of the portfolio given by Equation (4.7)*, using the SMSDC in Definition 15. Then, we apply the risk measures in Section 4.1 to estimate the risk in the portfolio for the different models.*

	Expected_value	Stddev	VaR95	CVar95
Model 1	0.0	0.0	0.0	0.0
Model 2	-1.2	0.1	0.2	0.3
Model 3	-1.2	0.2	0.4	0.5
Model 4	-1.3	0.3	0.5	0.6

Tab. 4.1: Different risk measures on the various market models with one asset. Figures are reported in percentage of cash liquidity policy (a)

Model 1 will value the sold securities at the same price as they were sold for. Therefore, it does not have a negative expected value. All models with liquidity risk have a negative expected value, while the models with stochastic liquidity gradients have heavier tails than the constant liquidity model (Model 2). The results are qualitatively in line with the exponential MSDC-model in Section 2.3.

We see that the introduction of liquidity (Model 2) decreases the value, while the introduction of liquidity risk (Model 3,4) mainly influences the risk of the portfolio.

These models assume that we can fairly accurately predict the present state of the SMSDC. This may not be the case, as seen in the difficulty of estimating the liquidity gradients in Section 3. It is likely that the return distributions would be even more heavy tailed should this assumption be relaxed.

Portfolio value for different liquidation horizons

An interesting problem in the SMSDC framework is to determine the execution horizon of a trade. Analyzing the dynamics of our illiquidity pricing formula in Remark 3, together with the stochastic dynamics of Equation (4.10), it is clear that the investor faces a trade-off between risk and market impact when deciding on the execution horizon. A longer execution horizon affects the portfolio value distribution in two ways: It decreases the market impact by (4.6), while at the same time increase market and liquidity risk through (4.10).

There are several ways that an investor would want to evaluate this trade-off. A risk neutral agent would be interested in the expected payoff, while a risk averse agent would weigh the larger losses higher than the profits. We will limit ourselves to a specific optimization problem where we assume that the investor is risk averse and wants to maximize his profit minus the risk contribution of the portfolio. Mathematically, the problem would be to find a τ such that we optimize the risk adjusted value of the portfolio:

$$E[V_\tau] - \rho(V_\tau) = max_{t\in[0,T]} E[V_t] - \rho(V_t) \tag{4.14}$$

where ρ is a risk measure.

Generalizations of (4.14) by e.g. weighing the risk measure differently is possible, and would follow through in the following example.

Example 4. *Assume that we are in Model 4 of Example 3. The investor is subject to a liquidity policy that states he has to raise an amount A Euros within a total time period of two days, $T = 2$. He still has to decide how he will execute the order that raises the amount needed. In our Gaussian/Brownian model the volatility increases with the square root of time,* *while the market impact decreases the longer he spends on the execution. Given that his utility function is as in* (4.14)*, he wants to decide his execution horizon* τ*, bounded above by two days.*

We have shown earlier that our value function (4.14) *is too complicated for analytical solutions. We will therefore resort to a Monte Carlo simulation to solve for the optimum.*

By creating a grid on $[0, 2]$ χ_t *can be simulated for different possible execution horizons. Then, by applying Remark 3 and using the SMSDC in Definition 15, the value distributions for the portfolio can be obtained. It is then possible to value the utility function* (4.14) *for different execution horizons.*

Figure 4.4 shows the result of the simulation. It shows the expected loss as a ratio of the cash liquidity policy (blue), and the sampled confidence intervals in grey (90% and 99%). As the expected value increases so does market uncertainty. When the portfolio is executed quickly, there is little uncertainty, but the execution cost is high. On the other hand, a long liquidation horizon gives a moderate expected loss, but a larger uncertainty on the loss size. The trade-off is clearly visible in the example. By choosing $\rho = VaR_{99\%}$ *in* (4.14) *we find that the investor's optimal strategy is to set his liquidation horizon to 6 hours (75% of a trading day).*

Note that in the example above we assume that the investor has to decide his execution horizon ex-ante. Another approach could be that he adapts his trade rate continuously during the trade. The problem can then be formulated as a stochastic control problem.

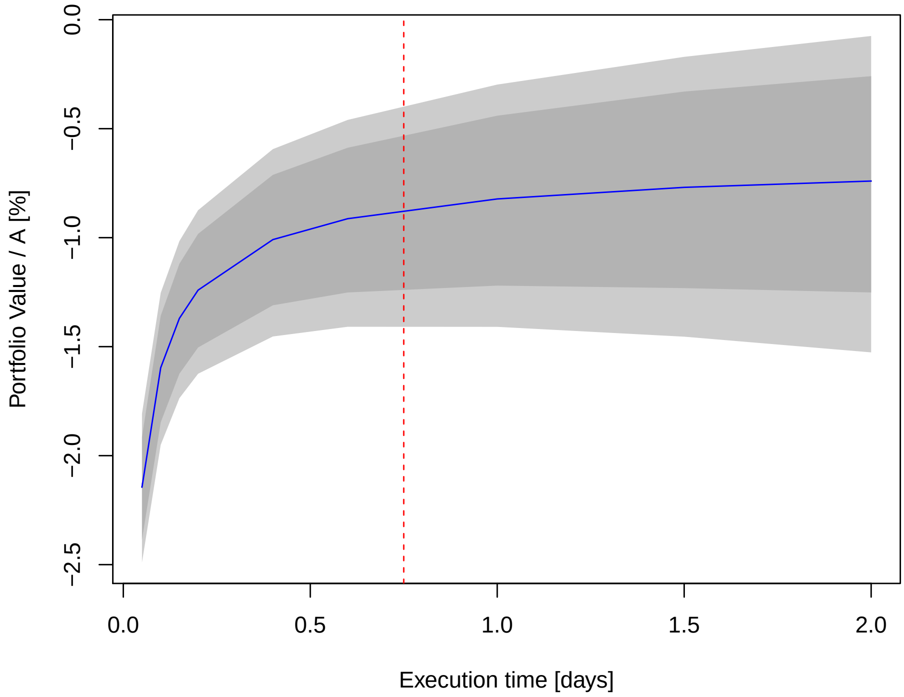


Fig. 4.4: Portfolio loss under a cash liquidity policy (A) for different execution periods. The grey areas indicate 90 pct. and 99 pct. confidence intervals (Value at Risk).

5. CONCLUSIONS AND FURTHER RESEARCH

Conclusion

In this thesis we have investigated the notion of portfolio value and risk in the presence of illiquidity. We adopted the formalism of [Acerbi and Scandolo, 2008] on valuing portfolios. They propose an alternative definition of value, that depends on the liquidity constraints of the owner. The authors show that this new definition of value can be defined in terms of a convex optimization problem under mild assumptions.

In Section 3 we explored the market impact model of [Almgren et al., 2005]. The authors estimate Supply Curves from historically executed orders obtained from Citibank. These data are strictly confidential and therefore unavailable to many institutions willing to construct such Supply Curves. We propose an alternative method of obtaining the same Supply Curve estimates by using publicly available transaction data. The method involves constructing a strategy identifier that can identify when an agent is trading large quantities of an asset. The results give market impact figures that are in the same order of magnitude as [Almgren et al., 2005] and with roughly the same prediction power. By using thin plate spline regression we seem to outperform the model described in [Almgren et al., 2005]. However, it should be noted that the prediction power is generally weak. General market risk acts as a highly noisy element in the analysis. We utilized cross-validation to obtain a more reliable out-of-sample prediction power.

In Section 4 we combined the illiquidity model of [Acerbi and Scandolo, 2008] with the market impact model of [Almgren et al., 2005]. We introduced the concept of a Stochastic Marginal Supply Demand Curve (SMSDC) as an extension of the MSDC. It is a stochastic function of both the position size and execution period, and will depend on future market conditions. We explored the distributional return properties for portfolios subject to SMSDCs and a cash liquidity policy. When illiquidity is added to the model, the value of the portfolio decreases. Including stochastic liquidity parameters makes the return distributions heavy tailed, as seen in Figure 2.3.

The introduction of SMSDCs illustrates that a portfolio can obtain a significantly higher expected value if some short-term market risk is accepted. We investigated a case where an agent needs to liquidate some positions within a certain time period. The investor then needs to choose to either liquidate his portfolio quickly, with high liquidation costs but low risks, or split his order over a longer period of time giving him lower liquidation costs but higher liquidity- and market risk. We propose to weigh this trade-off by considering a 'risk-adjusted portfolio value', and then finding the execution horizon that maximizes this adjusted value. We solve for the optimal execution horizon by a Monte Carlo simulation using the calibrated market liquidity model.

The formalism of [Acerbi and Scandolo, 2008] combined with the estimated Supply Curves of [Almgren et al., 2005] provides a promising model for valuing and assessing the risk of portfolios. The liquidity adjusted value defined by [Acerbi and Scandolo, 2008] provides a liquidity adjusted value of the portfolio, provides heavy-tailed price distributions and may be more informative than the mark-to-market pricing. We further believe the use of transaction data to estimate supply curves is a contribution to the market impact field independent of the formalism in [Acerbi and Scandolo, 2008].

The main drawbacks of the models are the increased computational difficulties and the loss of intuition. The liquidity adjusted value of a portfolio requires an optimization algorithm and several numerical inversion of functions. Estimating Supply Curves requires extensive data analysis on a large set of data. Using popular risk management techniques such as Monte Carlo simulations on top of these processes requires a large computational burden, as we experienced in Example 4. Analytical and intuitive tractability is lost when several of the steps have to be computed numerically even just to find the portfolio value. Therefore, finding sufficiently simple models that can increase the tractability and decrease the computational burden is necessary for the success of such a model.

Further research and improvements

There are several possible paths for further research in the area of liquidity. We have suggested improvements throughout the thesis, and will summarize some of them here.

In order to perform a full model validation of the transaction-based market impact model we described Section 3, we propose to test it against a data set of order data for the same period and the same equities. This could serve as a final test to see whether transactional data has the prediction power we observe in the analysis.

In addition, there are several possible improvements on the market impact models: Obtaining a complete overview of one security by incorporating alternative market places and derivative markets is likely to improve the regression results. Another consideration is to compare some alternative models to the ones we estimated in (4.6) and test the predictive power of each model. Such models may be continuous time models with discount factors and continuous time filtering techniques. Our analysis suggests that introduction of other regressional methods, such as non-parametric impact functions, may improve the prediction power of the models. Including jumps in the liquidity gradient variables is a realistic extension, as liquidity squeezes often come quickly and without warning. Further, more care in cleaning and initial filter parameters may give enhanced results for the methods.

As we proposed in Section 2.2, extending liquidity policies in option pricing theory could create interesting problems.

In Example 4 we consider how the value of a portfolio depends on the liquidation horizon. Determining the liquidation horizon requires the evaluation of a value vs. risk trade-off. This may be formulated as an optimal stopping problem where the investor can adjust the trade rate during liquidation.

BIBLIOGRAPHY

[Acerbi and Scandolo, 2008] Acerbi, C. and Scandolo, G. (2008). Liquidity risk theory and coherent measures of risk. *Quantitative Finance*, 8(7):681–692.

[Acerbi and Tasche, 2002] Acerbi, C. and Tasche, D. (2002). On the coherence of expected shortfall. *Working paper.*

[Alexander J. McNeil, 2006] Alexander J. McNeil, Rüdiger Frey, P. E. (2006). *Quantitative Risk Management: Concepts, Techniques, and Tools.* Princeton University Press.

[Almgren et al., 2005] Almgren, R., Thum, C., Hauptmann, E., and Li, H. (2005). Equity market impact. *Risk*, (July):57–62.

[Artzner, 2009] Artzner, P. e. a. (2009). Coherent measures of risk. *Mathematical Finance*, 9(3):203–228.

[Cetin et al., 2004] Cetin, U., Jarrow, R. A., and Protter, P. (2004). Liquidity risk and arbitrage pricing theory. *Finance and stochastics*, 8(3):311–341.

[Ferraris, 2008] Ferraris, A. (2008). Equity market impact models. (December).

[Follmer and Schied, 2002] Follmer, H. and Schied, A. (2002). Convex measures of risk and trading constraints. *Finance and Stochastics*, 6(4):429–447.

[Gatheral, 2010] Gatheral, J. (2010). No-dynamic-arbitrage and market impact. *Quantitative Finance*, 10(7):749–759.

[Hastie et al., 2009] Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning.* Springer.

[Higham, 2002] Higham, N. (2002). Computing the nearest correlation matrix - a problem from finance. *IMA journal of Numerical Analysis*, 22:329–343.

[J.P.Morgan, 1995] J.P.Morgan (1995). *RiskMetrics Technical Manual.* J.P. Morgan Bank.

[Kalman, 1960] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45.

[Øksendal, 2010] Øksendal, B. (2010). *Stochastic Differential Equations.* Springer.

[Shumway and Stoffer, 2011] Shumway, R. H. and Stoffer, D. S. (2011). *Time Series Analysis and Its Applications.* Springer.

[Willmott and Matsuura, 2005] Willmott, C. J. and Matsuura, K. (2005). Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Clim. Res.*, 30:79–82.

APPENDIX

.1 Source code used to analyse data

The thesis consist to a large degree of organizing and analysing data. In order to provide as much information as possible, we have appended the used source codes that implements the results. The files presented in this section are usually Sweave-files, which means it will be raw latex and raw R-code combined in the files. The syntax is such that R code is found between '«»=' and '@'. Outside these areas we are in standard tex-environment and one may see comments or text that explains the R code.

As there are extensive amount of code all is not necessarily well written with good comments. However, we hope that proividing these details will allow the interested reader to dig into the basis of the results found in the thesis.

.1.1 Example of exponential liquidity model

The following is the source code of the exponential model described in section 2.3. As this example is hard-coded into the thesis, it is merely the same section but includes the R code.

```
\Sexpr{set_parent('../../oppgave/main.Rnw')}


<<funtions, echo =FALSE>>=
library("MASS");library("xtable")

m.exp = function(x,A,k) A*exp(-k*x)
lambda.exp <- function(a,A,k) a * (sum(A/k)-a)**(-1)
r.exp <- function(a,A,k,p0 = 0) {
  if (a>p0) log(1+lambda.exp(a,A,k))/k
  else rep(0,A)
}
U.exp <- function(p,a,A,k,p0 = 0) sum(m.exp(0,A,k)*p) + p0
L.exp <- function(p,a,A,k,p0 = 0) sum(A/k*(1-exp(-k*p))) + p0
V.exp <- function(p,a,A,k,p0 = 0) {
  if (k>0) {
                r_a <- r.exp(a,A,k)
                U.exp(p-r_a,a,A,k,p0) + L.exp(r_a,a,A,k,0)
        } else if (k==0) sum(p*A)+p0
}
V.exp.direct <- function(p,a,A,k,p0 = 0)    sum(A* (p-r.exp(a,A,k
    )) ) + a


sample_one_asset <- function(sample_info,returner = "ES") {
  library("MASS")
  P = sample_info$pos
  a = sample_info$cashreq
  rho = sample_info$rho
  p0 = sample_info$cash
  A.hat = sample_info$A
```

```
  k.hat = sample_info$k
  sds = sample_info$sds
  D = list()


  # Scenario 1: k == 0, p = 0
  A.hat = 10
  cov0=diag(c(sds[1]**2,0))
  set.seed(1)
  AK <- mvrnorm(n,mu=c(A.hat,0),Sigma = cov0)
  D$pv1 <- sort(mapply(function(A,K) V.exp(P,a,A,K,p0), AK[,1],
      AK[,2]))

  # Scenario 2: k > 0 and fixed

  cov0=diag(c(sds[1]**2,0))
  set.seed(1)
  AK <- mvrnorm(n,mu=c(A.hat,k.hat),Sigma = cov0)
  AK[which(AK[,2]<0),2] = 0
  D$pv2 <- sort(mapply(function(A,K) V.exp(P,a,A,K,p0), AK[,1],
      AK[,2]))


  # Scenario 3: k > 0 and normal, p = 0
  cov0=diag(sds^2)
  set.seed(1)
  AK <- mvrnorm(n,mu=c(A.hat,k.hat),Sigma = cov0)
  AK[which(AK[,2]<0),2] = 0
  D$pv3 <- sort(mapply(function(A,K) V.exp(P,a,A,K,p0), AK[,1],
      AK[,2]))

  # Scenario 4: k > 0 and normal, p < 0
  cor0 = matrix(c(1,rho,rho,1),nrow=2)
  cov0 = sds%*%t(sds) * cor0
  set.seed(1)
  AK <- mvrnorm(n,mu=c(A.hat,k.hat),Sigma = cov0)
  AK[which(AK[,2]<0),2] = 0
  D$pv4 <- sort(mapply(function(A,K) V.exp(P,a,A,K,p0), AK[,1],
      AK[,2]))


  D = data.frame(D)

  q = floor(0.05*n)
  # Value at risk:
  VaR = D[q,]
  # Expected shortfall
  avg = apply(D,2,mean)
  ES =  avg-apply(D[1:q,],2,mean)

  if (returner=="ES")           {ES
  } else if (returner == "D")     {D
  } else if (returner == "avg") {avg
}
```

```
}

@

In this section we will present a simple model of the
    exponential type proposed in \cite{acerbi} where the MSDC of
    an asset i takes the form $m_i(x) = A_i e^{-k_ix}$.
The parametric form of the MSDC is simple and tractable and
    there exists a closed-form solution of the value function, as
     shown in \cite{acerbi}. We will aim at showing some stylized
     facts about the model, by assuming stochastic dynamics on
    the variables $A_i$ and $k_i$.
The variable $A_i$ will model the market volatility much in the
    same way that standard finance litterature models ut, while $
    k_i$ will be our liquidity parameter.
A large $k_i$ will result in a less liquid asset, and the value
    of the position will be negatively related to the position
    size.

By using Proposition 6.1 in \cite{acerbi} we can show that the
    optimal solution for the exponential model is \footnote{\cite
    {acerbi} must have assumed $p_0 = 0$. We have included this
    term for completeness}

\begin{align}
\lambda &= \frac{a-p_0}{\sum_{i=1}^N A_i/k_i-a+p_0} \\
r_i^a &= \frac{1}{k_i} \log(1+\lambda) \\
V^L{a}(\mathbf p) = U(\mathbf p- & \mathbf r^a) - L(\mathbf r^a)
     = \sum_{i=1}^N A_i(p_i-r_i) + a + p_0
\end{align}


There are several interesting points regarding the model:
\begin{itemize}
\item The model assumes no restrictions or costs on short
    selling. \\
\item The optimal liquidation policy $r^a$ is independent of the
     current position-vector (except $p_0$). This is a
    consequence of the no short selling restriction. \\
\item The portfolio value is linear in each $p_i$, $i \neq 0$.
    This is because all position not liquidated to satisfy the
    cash policy is valued at the mark-to-market value.
\end{itemize}

Let us now introduce some simple stochastic elements in the
    model.
Assume that we wish to evaluate a portfolio in a future time and
     that we know that the distribution of A and k are gaussian:
    $A \sim N(10,0.04)$ and $k \sim N(0.001,10^{-7})$.
By simulating values of A and K we get a distribution of MSDC
    curves as seen in figure \ref{fig:exp_sampled}.

<<label="parameter">>=
sample_info = list(
```

```
  pos = 1000,      # position of asset
  cashreq = 3000, # liquidity policy a
  cash=0,          # cash held p0
  k = 0.001,
  A = 10,
  rho=-0.5,          # correlation between A and k
  sds = sqrt(c(0.2,1e-7)) # standard deviations of A and k
)

@


<<exp_sampled, echo=FALSE, fig.height=4, fig.cap="Simulated MSDCs
    for different values of A and k. The black line is the mean
    MSDC.">>=
  cor0 = matrix(c(1,sample_info$rho,sample_info$rho,1),nrow=2) #
      correlation matrix
  #cor0 = diag(2)
  cov0 = sample_info$sds%*%t(sample_info$sds) * cor0
  set.seed(1)
  P = seq(-1000,1000,100)
  AK <- mvrnorm(20,mu=c(sample_info$A,sample_info$k),Sigma =
      cov0)
  msdc_values = outer(AK[,1],P,function(A,p,K) m.exp(p,A,K),AK
      [,2])
  matplot(P,t(msdc_values), type = "l", xlab = "Position",
          ylab = "Price")
  lines(P,m.exp(P,sample_info$A,sample_info$k), lwd = 2)
@

In \ref{fig:exp_sampled}, each line corresponds to the MSDC
    function with a different simulated $(A,k)$ pair.
The MSDC corresponding to the expected value of A and k is the
    thick black line.
Due to our gaussian assumption on A there is no guarantee that
    the prices are positive.
This is a limitation to the model, but by having sufficient low
    standard deviations on A we can ensure that the probability
    of negative prices is effectively zero.

Also note that the price has a significantly larger spread for
    buying positions ($pos<0$) than for selling positions ($pos
    >0$).
The effect is due to the nature of the exponential function, and
     is in our model determined by the variable k.

We wish to investigate the value and risk of a portfolio in this
     model, and how it changes with different assumptions on A
    and k.
Assume we have a portfolio consisting of a single asset.
    Following the idea of Acerbi (2007) we will consider four
    different scenarios:
\begin{enumerate}
  \item A normal and $k = 0$. The gaussian model with no
```

```
      liquidity risk
  \item A normal and $k \ge 0$ fixed. It is a model with static
      liquidity risk.
  \item A, k joint normal with zero correlation. i.e. it assumes
       independence of market and liquidity risk.
  \item A and k joint normal with a negative correlation. The
      model assumes that as asset prices fall, liquidity is
      likely to decrease.
\end{enumerate}

Assume parameters as given above.
The uppermost mark-to-market value of this portfolio is simply $
    V(P) = m(0)*P = 10*1000 = 10 000$.
By performing a Monte Carlo simulation on the MSDCs with the
    parameters given above, we are able to approximate the
    distributions of the different models.
The resulting distributions are plotted in \ref{fig:distr}.

\begin{figure}[h] \label{fig:distr}
  \caption{Portfolio distributions of model 1 to model 4}
<<echo=FALSE>>=
n = 10000
D = sample_one_asset(sample_info,"D")


plot(density(D$pv1), xlim=c(7000,12000), ylim = c(0,0.0014),main
     ="", xlab ="Portfolio value")
lines(density(D$pv2),col = "red")
lines(density(D$pv3),col = "blue")
lines(density(D$pv4),col = "green")

legend(11000,0.001, # places a legend at the appropriate place
       c("Model 1","Model 2","Model 3","Model 4"), # puts text
           in the legend
  lty=c(1,1), # gives the legend appropriate symbols (lines)
lwd=c(2.5,2.5),col=c("black","red","blue","green")) # gives the
    legend lines the correct color and width

@
\end{figure}

The most clear result is that our first model is gaussian with
    mean at the uppermost value of the portfolio.
By introducing liquidity risk in the form of a positive
    parameter $k$ in model 2-4, the expected value of the
    portfolio decreases.
In fact, it seems like model 2-4 have shifted their peak
    probabilities to a new point around 9500, which is the centre
     of the (A,k)-distribution (which is common for model 2-4). \
    footnote{The value of a portfolio is 9 433 at the point $A =
    10, k = 0.001$.}
The introduction of liquidity risk decreases the general value
    of the portfolio in this model, as should have been expected.
```

```
More interesting is the comparison between model 2-4.
They are each special cases of the model $(A,k) \sim N(\mu,\
    Sigma)$, where $\mu$ and $\Sigma$ are a general mean vector
    and covariance matrix.
Comparing the empirical distributions, it is qualitatively clear
     that the fat tails increases from model 1 to 4.
The fact can be accompanied by testing different risk measures
    on the portfolios.
In our Monte Carlo simulation, we can extract estimates of the
    standard deviation, 5 pct. Value at Risk and 5 pct. Expected
    Shortfall.
Let us define these risk measures in the usual way and as the
    distance from the mean of the portfolio\footnote{Formally
    defined in chapter \ref{sec:marketimpact}.}.
The results are found in table \ref{tbl:riskstats}.

<<echo = FALSE, results='asis'>>=

median = apply(D,2,median)
avg = apply(D,2,mean)
q = floor(n*0.05)
stddev <- apply(D,2,sd)
VaR = avg-t(D[q,])
ES = avg-apply(D[1:q,],2,mean)
pp <- data.frame(avg,stddev,VaR = VaR, ES)
colnames(pp) <- c("Portfolio Value","Std.Dev","VaR","ES")
rownames(pp) <-  c("Model 1","Model 2","Model 3","Model 4")

capStr = "Risk statistics for the four exponentially MSDC models
     that are being tested. Numbers denominated in cash value."
print(xtable(pp,digits=0,caption = capStr,label="tbl:riskstats")
    )
@

We see that with all risk measures, the risk increases from
    Model 1 to Model 4.
Although introducing liquidity risk in Model 2 decreased the (
    average) value of the portfolio significantly, the model
    suggests that adding stochastic dynamics to the available
    liquidity has a significant impact on the risk measures used.
Secondly, the correlation between the price variable A and the
    liquidity variable $k$ is important, as seen from the
    differences between Models 3 and 4, which has a correlation
    of 0 and 0.5 respectively.

The correlation effect is significant when executing the model
    with a correlation varying from -1 to 1, as seen in Figure \
    ref{fig:corES}, where we modified Model 4 and left the other
    model outputs with the original correlation assumptions.

\begin{figure} \label{fig:corES}
  \caption{How cor(A,k) affects the expected shortfall. We can
      see that only model 4 depends on the correlation between
      the two models, and that the portfolio risk is higher when
```

```
    there is a strong correlation between A and k. Model 1 (
    black), Model 2 (red), Model 3 (blue) and Model 4 (green).}
<<echo=FALSE, fig.height=3>>=
# See how ES varies with correlation of A and k
  rho.seq = seq(-1,1,0.2)
        ESS = mat.or.vec(nc=4,nr=length(rho.seq) )
        for (i in 1:length(rho.seq) ) {
                sample_info$rho = rho.seq[i]
                ESS[i,] <- sample_one_asset(sample_info)
        }

        matplot(rho.seq,ESS,type ="l", col = c("black","red","
            blue","green"),
          xlab = "Correlation A and k", ylab = "Expected
              Shortfall")


@
\end{figure}

There are several stylized results that can be drawn from the
    example above.
We see that the value of the portfolio decreases as one
    introduces liquidity in the valuation model.
The effect is an expected result of the new valuation model.
However, it is not before we introduce stochastic liquidity in
    the later models that we see a large effect on the tails of
    the distributions.
The portfolio becomes significantly more heavy-tailed when
    stochastic liquidity is introduced.
In addition, a negative correlation between price return and
    liquidity has the largest effect on the heavy-tailedness.

The results motivate us to investigate stochastic liquidity-
    models where we try to estimate the size of the liquidity
    effect and impose reasonable assumptions on the distribution
    of the different
market and liquidity variables.
```

.1.2 Data Initialization source code

Taking the raw data as input and then handling it as described in section 3.4. The different stocks are run through the same code just with different input file.

```
\documentclass{article}

\usepackage{mathtools}
\newcommand{\defeq}{\vcentcolon=}

\begin{document}
```

```
<<echo=FALSE>>=
 #parm = list()
 #parm$ticker <- "STL"


#load("../data/bigdiff_compiled_YAR_v4.Rdata")
#opts_chunk$set(echo=TRUE, warning=TRUE, results=TRUE, eval=TRUE)
@

\title{Initializing data for Market Impact analysis}
\maketitle

This document is only used for creating the results-vector that
    can be used by future analysis of the data.
It can be run on different time series, ticker specified above.

<<raw-data>>=

# RAW DATA TREATMENT OF TRANSACTION DATA
# PUT ALL TRANSACTION DOWN TO MEMBER LEVEL INTO TRADING SLOTS
    with 30 min intervals.

raw_old <- read.table(file.path("..","data",paste(parm$ticker,"_
    all.results",sep="")))
















raw_new <- read.table(file.path("..","data",paste("tradedata_",
    parm$ticker,"_3mnd.csv",sep="")))
```

```
raw <- rbind(raw_old,raw_new)

#raw <- raw_new
 tail(raw)

members <- sort(unique(raw$ACCOUNT_BUY))
masterdate <- sort(unique(as.Date(raw$TIME_OF_TRADE_EXECUTION)))
raw$TIME_OF_TRADE_EXECUTION <- as.POSIXlt(raw$TIME_OF_TRADE_
    EXECUTION)

# Assoicate slot for each trade
  parm$slots <- seq(9.25,16.25,.5) # e.g. slot 1 is traded after
       9:15 and before 9:45.
  parm$ns <- length(parm$slots)-1 # the last slot is associated
      with the ending auction and should not be considered.
  timeofday <- raw$TIME_OF_TRADE_EXECUTION$h + raw$TIME_OF_TRADE
      _EXECUTION$min/60
  raw$slotcat <- sapply(timeofday, function(x) sum(x >= parm$
      slots) )

# Remove trades that happened in start and end auctions
  remove <- which(raw$slotcat == 0 | raw$slotcat == parm$ns+1)
  prodata <- raw[-remove,]
  tot <- length(masterdate)
  N <- tot*parm$ns
  M <- length(members)
# Lag mastertick
  date <- match(as.Date(prodata$TIME_OF_TRADE_EXECUTION),
      masterdate)
  prodata$tick <- prodata$slotcat + (date-1)*parm$ns

        mastertick <- 1:N

                  #prodata$date <- as.Date(prodata$TIME_OF_TRADE_
                      EXECUTION)
# Lag supportdatatabell som følger tick'sene nedover:
  supportdata <- list()
  supportdata$tick <- 1:N
        supportdata$tickdate <- sort(rep(masterdate,parm$ns))
  supportdata$date <- ceiling(supportdata$tick/parm$ns)
  supportdata <- data.frame(supportdata)
  head(supportdata)


#### CREATE POSITIONS VECTORS PR SLOT ####
```

```
  #h <- mat.or.vec(nr=N, nc=M)
  h <- list() #array(data=0,dim=c(N,M,4), dimnames=list(tick =
      1:N,members,c("amount","value","price","cumpos")))
  h$amount <- mat.or.vec(nr=N,nc=M)
  h$value <- mat.or.vec(nr=N,nc=M)
  h$price <- mat.or.vec(nr=N,nc=M)
  h$cumpos <- mat.or.vec(nr=N,nc=M)


ind_buy <- match(prodata$ACCOUNT_BUY, members)
  ind_sell <- match(prodata$ACCOUNT_SELL, members)
turnovertick = rep(0,N); Vtick = rep(0,N)

# Lag pos-pr-medlem-vektor:
  for (i in 1:nrow(prodata)) {
    h$amount[prodata$tick[i],ind_buy[i]] <-  h$amount[prodata$
        tick[i],ind_buy[i]]  + prodata$AMOUNT1[i]
    h$amount[prodata$tick[i],ind_sell[i]] <- h$amount[prodata$
        tick[i],ind_sell[i]] - prodata$AMOUNT1[i]

    h$value[prodata$tick[i],ind_buy[i]] <-  h$value[prodata$tick
        [i],ind_buy[i]]  + prodata$AMOUNT1[i]*prodata$TRADE_PRICE
        [i]
    h$value[prodata$tick[i],ind_sell[i]] <- h$value[prodata$tick
        [i],ind_sell[i]] - prodata$AMOUNT1[i]*prodata$TRADE_PRICE
        [i]

    turnovertick[prodata$tick[i]] <- turnovertick[prodata$tick[i
        ]] + prodata$AMOUNT1[i]*prodata$TRADE_PRICE[i]
    Vtick[prodata$tick[i]] <- Vtick[prodata$tick[i]] + prodata$
        AMOUNT1[i]
  }

  # Create cumulative positions
  h$cumpos <- apply(h$amount, 2, cumsum)
@

<<supportdata_vol_volume_price>>=

# Price serie
  S = turnovertick/Vtick

          # If no price...: first try to find price forward in
              time, then backwards.
            ind = which(is.na(S))
            iter=0;max.iter=100
            while ( length(ind) > 0 & iter< max.iter) {
              S[ind] <- S[ind+1]
              ind = which(is.na(S))
              iter = iter+1
            }
            while ( length(ind) > 0 & iter< max.iter*2) {
              S[ind] <- S[ind-1]
              ind = which(is.na(S))
```

```
            iter = iter+1
          }

  supportdata$S <- S

#### Create (simple!) intraday volatility estimate ####
   r <- log(tail(S,-1)/head(S,-1))
   r[which(is.nan(r))] <- 0
#
#   var0 <- c(r[1]**2)
#     for (k in 2:N) {
#       var0[k] <- mean(r[max(1,k-28):(k-1)]**2)
#     }
#   supportdata$sigma <- sqrt(var0*parm$ns)

### ewma-intraday volatility estimate
ewma0 <- c(r[1]**2)
lambda = 0.99
    for (k in 2:N) {
      ewma0[k] <- lambda*ewma0[k-1] + (1-lambda)*r[k]**2
    }
ewma0[N] <- ewma0[N-1]
  supportdata$sigma <- sqrt(ewma0*parm$ns)


#### Create (simple!) 10-day average daily volume estimate ####
#   V <- c(Vtick[1])
#   for (k in 2:N) {
#     V[k] <- mean(Vtick[max(1,k-parm$ns*10):(k-1)])
#   }
#   supportdata$V <- parm$ns*V

### EWMA-volume
  Vewma <- c(Vtick[1])
  for (k in 2:N) {
    Vewma[k] <- 0.985*Vewma[k-1] + (1-0.985)*Vtick[k]
  }
  supportdata$V <- parm$ns*Vewma


#### various info, needed? ####
h$relamount <- h$amount/supportdata$V
h$relcumpos <- apply(h$relamount, 2, cumsum)


#### FIND TRANSACTION PRICE ####
h$price <-h$value/h$amount
ind0 <- which(!is.finite(h$price) | (h$amount >-10 & h$amount
    <10))
h$price[ind0] <- NA
@

<<STRATEGY-CREATOR>>=
```

```
strategies0 <- data.frame(start=c(),end = c(), member=c())

for (t in 1:tot) {
  for (m in 1:M) {
    # m = 4; t  = 2
  ind_t <- which(supportdata$date ==t)

 maks <- which(h$relcumpos[ind_t,m] == max(h$relcumpos[ind_t,m])
     )+parm$ns*(t-1)
  min <- which(h$relcumpos[ind_t,m] == min(h$relcumpos[ind_t,m])
      )+parm$ns*(t-1)
  # Regler dersom det er flere maks og min:
  if ((length(maks) > 1 |length(min) > 1)  & max(maks) < min(min
      )) maks = max(maks);min=min(min)
  if ((length(maks) > 1 |length(min) > 1)  & min(maks) >=max(min
      )) maks = min(maks);min=max(min)

  start <- min(maks,min)
  end <- max(maks,min)
  strat_temp <- data.frame(start=start,end=end,member=m)
  strategies0 <- rbind(strategies0, strat_temp)
  }
}

# Remove strategies with zero timelength:
remove <- which(strategies0$start== strategies0$end)
strategies <- strategies0[-remove,]
# interpret strategies:
tot.strat <- nrow(strategies)
@

\subsubsection{Extract market data from identified strategies}
Extract all needed variables (XV,sigma,tau,I,J) for each
    strategy.

<<extract_data_from_strategies>>=

resultater0 = data.frame(stratnr = c(),startslot =c(), startdate
     = c(), X=c(),V=c(),sigma=c(),tau=c(),I=c(),J=c(),sigma_delta
    =c())
for (i in 1:tot.strat) {
  strat <- strategies[i,]
  int <- strat$start:strat$end
  startdate <- supportdata$tickdate[strat$start]
  startslot <- strat$start

  S0 <- supportdata$S[max(1,strat$start-1)] # price before order
  Sp <- supportdata$S[min(N,strat$end+1)] # price after order
  Sbar <- sum(h$price[int,strat$m] * h$amount[int,strat$m],na.rm
      =TRUE) / sum(h$amount[int,strat$m])

  sigma <- mean(supportdata$sigma[int]) # per day
  sigma_delta <- supportdata$sigma[strat$end] - supportdata$
      sigma[strat$start]
```

```
  tau  <- (strat$end-strat$start)/parm$ns # per day

  X <- sum(h$amount[int,strat$m]) # no. stocks
  V <- mean(supportdata$V[int]) # [Average daily volume over
      trading horizon]
  I = Sp/S0-1
  J = Sbar/S0-1

  newline <- data.frame(stratnr = i,startslot,startdate, X,V,
      sigma,tau,I,J,sigma_delta)
  resultater0 = rbind(resultater0,newline)
}

  resultater0$XV <-resultater0$X/(resultater0$V) # stocks as
      percentage of daily volume. (STL/)
  resultater0$XVT <- resultater0$XV/resultater0$tau # stocks as
      percentage of volume during the trade

@


\subsection{Market Impact analysis}


\subsubsection{Outliers}
<<remove-zero-pos>>=
# Fjern alle små- og nullposser
meansigma <-  mean(resultater0$sigma,na.rm=TRUE)

p = 0.03
inliers <- quantile(resultater0$XV*resultater0$sigma/meansigma,
    probs=c(p,1-p),na.rm=TRUE) #c(-.05,.05)#
remove_inliers <- (resultater0$XV*resultater0$sigma/meansigma>
    inliers[1] & resultater0$XV*resultater0$sigma/meansigma<
    inliers[2])
plot(sort(with(resultater0,XV*sigma/meansigma)),ylim = inliers
    *10)
abline(inliers[1],0,col="red")
abline(inliers[2],0,col="red")

round(inliers,2)

# Remove too big positions
p = 0.005
outliers <- quantile(resultater0$XV,probs=c(p,1-p)) #c(-.05,.05)
    #
remove_outliers <- (resultater0$XV<outliers[1] | resultater0$XV>
    outliers[2])

plot(resultater0$XV, ylim = outliers*3)
abline(outliers[1],0, col = "red")
abline(outliers[2],0, col = "red")

# Remove strong signal in J
```

```
        vari <- with(resultater0,XV/(J*sigma/meansigma))
        highsignal <- quantile(vari,probs = c(0.005,0.995),na.rm
            =TRUE)

        plot(vari,ylim = 2*highsignal)
        abline(highsignal[1],0,col="red")
        abline(highsignal[2],0,col="red")

        remove_highsignal_J <- vari<highsignal[1] | vari>
            highsignal[2]

# Remove strong signal in I
        vari <- with(resultater0,XV/(I*sigma/meansigma))
        highsignal <- quantile( vari,probs = c(0.005,0.995),na.
            rm=TRUE)

        plot(vari,ylim = 2*highsignal)
        abline(highsignal[1],0,col="red")
        abline(highsignal[2],0,col="red")

        remove_highsignal_I <- vari<highsignal[1] | vari>
            highsignal[2]

# plot(resultater0$startdate,resultater0$I/(resultater0$XV*
    resultater0$sigma), main = "Remove liquidity signals above
    predefined threshold", ylab = "Gradient signal", xlab = "Date
    ",ylim = c(-50,50)*3)
# abline(50,0, col = "red")
# abline(-50,0, col = "red")


resultater <- resultater0[-which(remove_highsignal_J | remove_
    highsignal_I | remove_inliers),]
dim(resultater)

@



<<save_workspace>>=
  hfun <- function(XVT,sigma) sign(XVT)*sigma*abs(XVT)**(3/5)
  resultater$h_dat <- hfun(resultater$XVT,resultater$sigma)
save.image(file = paste("../data/bigdiff_compiled_",parm$ticker,
     "_v4.Rdata",sep=""))
@


\end{document}
```

.1.3 Regression analysis source code

Regressions on each stock

The following source code is used on each stock to identify permanent and temporary impacts through regression.

```
\documentclass{article}

\usepackage{mathtools}
\newcommand{\defeq}{\vcentcolon=}

\begin{document}
<<load-reg-data>>=
load("../data/bigdiff_compiled_STL_v4.Rdata")

#load(file = paste("../data/bigdiff_compiled_",parm$ticker, "_v4
    .Rdata",sep=""))


@


\subsubsection{Division into training and test data}

<<>>=
parm$traintestratio <- 0.7
set.seed(6)
draw <- runif(nrow(resultater))
ind_train <- draw < parm$traintestratio
ind_test <- draw > parm$traintestratio

train <- resultater[ind_train,]
test <- resultater[ind_test,]
@


\subsubsection{Analysis of permanent impact}
Instead of relying on \ref{eq:permanent_model}, we let ($I/\
   sigma$) be regressed nonparametrically on the variables using
    generalized linear models.

<<alternative_permanent_models>>=
par(mfrow=c(1,1))
rmseI=list()
rmseI$null <- sd(test$I)
maeI <- list()
maeI$null <- mean(abs(test$I))
require(mgcv)

reg_originalI <- gam(I ~ I(XV*sigma)-1, data=train)
summary(reg_originalI)
predI_original <- predict(reg_originalI,newdata=test)
rmseI$original <- sd(test$I-predI_original,na.rm=TRUE)/rmseI$
```

```
    null
maeI$original <- mean(abs(test$I-predI_original),na.rm=TRUE)/
    maeI$null

reg_original_smooth <- gam(I(I) ~ s(I(XV*sigma))-1, data=train,
    family = gaussian(link = identity))
summary(reg_original_smooth)
plot(reg_original_smooth)
predI_original_smooth <- predict(reg_original_smooth,newdata=
    test)
rmseI$original_smooth <- sd(test$I-predI_original_smooth,na.rm=
    TRUE)/rmseI$null
maeI$original_smooth <- mean(abs(test$I-predI_original_smooth),
    na.rm=TRUE)/maeI$null

reg_gam1 <- gam(I ~ s(X,V,tau,sigma)-1, data=train, family =
    gaussian(link = identity))
summary(reg_gam1)
predI_gam1 <- predict(reg_gam1,newdata=test)
rmseI$gam1 <- sd(test$I-predI_gam1,na.rm=TRUE)/rmseI$null
maeI$gam1 <- mean(abs(test$I-predI_gam1),na.rm=TRUE)/maeI$null

@

<<permanent-cross-validation>>=
# do a 10-fold cross validation
b <- nrow(resultater)
resultater_cv <- resultater[sample(1:b,b),]
pred_crossvalid <- c(); coeff_cv <- c()
for (i in 1:(b/10) ) {
  ind_testcv <- (10*(i-1)+1) : (10*(i-1)+10)
  train_cv <- resultater_cv[-ind_testcv,]
  test_cv <- resultater_cv[ind_testcv,]
  reg <- gam(I ~ I(XV*sigma)-1, data=train_cv)
  predI_cv <- predict(reg,newdata=test_cv)
  pred_crossvalid[i] <- mean(abs(predI_cv-test_cv$I))
  coeff_cv[i] <- reg$coefficients
}

plot(pred_crossvalid/mean(abs(resultater_cv$I)))
plot(coeff_cv)
mean(pred_crossvalid)/mean(abs(resultater_cv$I)) #prediction
    power
mean(coeff_cv) # predicted cv-coeff
sd(coeff_cv)

I_cv <- list(coeff = coeff_cv,
                          MAE = pred_crossvalid/mean(abs(resultater
                              _cv$I)))

@

<<results='asis'>>=
library("xtable")
```

```
xtable(t(as.data.frame(maeI)),digits=3,
       caption = "various  mean absolute errors for predicting
          permanent impact")
@


\subsubsection{Analysis of temporary impact}

<<temporary_impact_analysis>>=
rmseJ =list()
rmseJ$null = sd(test$J-test$I/2)
maeJ <- list()
maeJ$null <- mean(abs(test$J-test$I/2))
  reg_originalJ <- gam(I(J-I/2) ~ h_dat-1,data=train)
  summary(reg_originalJ)
  predJ_original <- predict(reg_originalJ,newdata=test)
  rmseJ$original <- sd(test$J-test$I/2 - predJ_original)/rmseJ$
      null
  maeJ$original <- mean(abs(test$J-test$I/2-predJ_original)) /
      maeJ$null

  reg_original_smooth <- gam(I(J-I/2) ~ s(XVT),data=train)
  summary(reg_original_smooth)
  plot(reg_original_smooth, main = "Temporary impact as
      Almgren05 but smoothed XVT")
  predJ_original_smooth <- predict(reg_original_smooth,newdata=
      test)
  rmseJ$original_smooth <- sd(test$J-test$I/2 - predJ_original_
      smooth)/rmseJ$null
  maeJ$original_smooth <- mean(abs(test$J-test$I/2-predJ_
      original_smooth)) / maeJ$null

  reg_gam1 <- gam(I(J-I/2) ~ s(I(X/V),sigma,tau,bs="tp")-1,data=
      train)
  summary(reg_gam1)
  predJ_gam1 <- predict(reg_gam1,newdata=test)
  rmseJ$gam1 <- sd(test$J-test$I/2 - predJ_gam1)/rmseJ$null
  maeJ$gam1 <- mean(abs(test$J-test$I/2-predJ_gam1)) / maeJ$null





plot(reg_original_smooth,xlim=c(-.5,.5), ylab = "Teporary Impact
    ",xlab="Smoothed XVT variable")

@

<<plot 4dim smooth model>>=
# Plot example prediction:.
  gam_all <- gam(I(J-I/2) ~ s(X,V,sigma,tau,bs="tp"),data=
      resultater)

  #gam_all <- gam(I((J-I/2*sign(-X))) ~ s(abs(X),V,sigma,tau,bs
```

```
    ="tp"),data=resultater)


xr = quantile(resultater$X, probs=c(0.05,0.95))

TAU = c(0.3,0.5,0.7,1)
plot(0,0,xlim=xr/mean(resultater$V),ylim=c(-.003,.003),type="n",
     xlab = "Position Size as percent of daily traded volume",
    ylab = "Temporary Impact")
legend("topright",legend=paste0("tau ",TAU),lwd=2,col=rainbow(4)
    )
for (i in seq_along(TAU)) {
example = data.frame (X = seq(-5e6,5e6,length.out=50),
```

```
temp_impacts = predict(gam_all,newdata=example)
lines(example$X/example$V,temp_impacts, type = "l", col =
    rainbow(4)[i])
}
@



<<results='asis'>>=
require(xtable)

xtable(t(as.data.frame(rmseJ)),digits=3,
       caption = "various root mean squared errors for
           predicting temp. impact")
@

<<temporary-cross-validation>>=
# do a 10-fold cross validation
b <- nrow(resultater)
resultater_cv <- resultater[sample(1:b,b),]
pred_crossvalid <- c(); coeff_cv <- c()
for (i in 1:(b/10) ) {
  ind_testcv <- (10*(i-1)+1) : (10*(i-1)+10)
```

```
  train_cv <- resultater_cv[-ind_testcv,]
  test_cv <- resultater_cv[ind_testcv,]

  #reg <- gam(I(J-I/2) ~ h_dat-1,data=train_cv)
  reg <- gam(I(J-I/2) ~ s(X,V,sigma,tau,bs="tp"),data=train_cv)
  predJ_cv <- predict(reg,newdata=test_cv)
  pred_crossvalid[i] <- mean(abs(test_cv$J-test_cv$I/2-predJ_cv)
      )
  #coeff_cv[i] <- reg$coefficients
}


J_cv <- list(coeff = coeff_cv,
                             MAE = pred_crossvalid/mean(abs(resultater
                                  _cv$J-0.5*resultater_cv$I)))

plot(pred_crossvalid/mean(abs(resultater_cv$J-0.5*resultater_cv$
    I)))
plot(coeff_cv)
mean(pred_crossvalid)/mean(abs(resultater_cv$J-0.5*resultater_cv
    $I))
mean(coeff_cv)
@



<<temp-using-only-J>>=
  reg <- gam(J ~ s(X,V,sigma,tau,bs="tp")-1,data=train)
        predJ <- predict(reg,newdata=test)

errors <- abs(test$J-predJ)#/test$J
 maeJ_gam <- mean(errors) / mean(abs(test$J))

summary(reg)
@



<<temporary-using-J-directly>>=
b <- nrow(resultater)
resultater_cv <- resultater[sample(1:b,b),]
pred_crossvalid <- c(); coeff_cv <- c()
for (i in 1:(b/10) ) {
  ind_testcv <- (10*(i-1)+1) : (10*(i-1)+10)
  train_cv <- resultater_cv[-ind_testcv,]
  test_cv <- resultater_cv[ind_testcv,]

  reg <- gam(J ~ s(X,V,sigma,tau,bs="tp"),data=train_cv)
  predJ_cv <- predict(reg,newdata=test_cv)
  pred_crossvalid[i] <- mean(abs(test_cv$J-predJ_cv))
  #coeff_cv[i] <- reg$coefficients
}

MAE_J_cv = pred_crossvalid/mean(abs(resultater_cv$J))

plot(MAE_J_cv)
```

```
mean(MAE_J_cv)

@



\subsubsection{Example}
Assuming that daily volatility of 10 pct. and a trading time of
    0.2 days, using the results found above we can calculate the
    expected liquidity impacts:

<<echo=TRUE>>=
par(mfrow=c(1,1))
gamma <- reg_originalI$coeff
eta   <- reg_originalJ$coeff
dailyvol = 0.010
tradingtime = 0.2

XV = seq(-0.2,0.2,0.05)

I = gamma*dailyvol*XV
K =  eta*hfun(XV/tradingtime, dailyvol)

J = I/2+K

data.frame(XV,Ibp=I*1e4,Jbp=J*1e4)

plot(XV,J*1e4, type = "l", col = "red",
     ylab = "Impact (bps)",
     xlab = "Stock share of daily volume",
     main = paste("Liquidity Impacts for",parm$ticker),
     sub  = "Realized (red), Permanent (green), Temporary (grey)
         , 10 pct. volatility"
     )
lines(XV,I*1e4, col = "green")
lines(XV,K*1e4, col = "grey")
@

An intersting feature of the calibrated model is that the
    realized expected impact is less than the permanent impact
    for trades over half of the daily traded volume.
Whether this is due to a misspecified framwork or problematic
    calibrations can be discussed.
It is at least a large extrapolation:
90\% of the trades in the dataset ranges from \Sexpr{round(
    quantile(resultater$XV,0.1),2)} to \Sexpr{round(quantile(
    resultater$XV,0.9),2)} of order sizes of daily volume.

\subsection{Conclusions}
The analysis shows significant positive coefficients $\eta$ and
    $\gamma$.
The $R^2$ value is small, around 1 and 3 pct., but the levels
    are at the same or higher levels than in \cite{almgren05}.
```

```
However, alternative models seems to give higher $R^2$'s.

Weakness: these regression models does not account for time
    variability. Should check by time series analysis: e.g.
    Kalman Filter.


<<save_results>>=
save.image(paste("../data/regression_analysis_",parm$ticker, "_
    v4.Rdata",sep=""))
@



\end{document}
```

Temporary impact regression using all stocks

The following code is used to find a temporary impact coeffcient across all stocks.

```
\documentclass{article}

\begin{document}

<<>>=
## Temporary impact using all stock data
setwd("C:/Users/OsloCleari/Dropbox/master1/almgren_filter")
## LOAD AND ORGANIZE INTO allresults
load("../data/regression_analysis_STL_v4.Rdata")
resultater$stock <- "STL"
resultater$gamma <- 1.07
allresults <- resultater
load("../data/regression_analysis_YAR_v4.Rdata")
resultater$gamma <- 0.92
resultater$stock <- "YAR"

allresults <- rbind(allresults,resultater)
load("../data/regression_analysis_SCH_v4.Rdata")
resultater$gamma <- 0.18
resultater$stock <- "SCH"
allresults <- rbind(allresults,resultater)

@




<<permanent-impact>>=

allresults$STL <- c(allresults$stock == "STL")*1
allresults$YAR <- c(allresults$stock == "YAR")*1

regI <- lm( I ~ I(sigma*XV) + STL+YAR-1,data=allresults)
```

```
summary(regI)

@


<<temp-on-all>>=
parm$traintestratio <- 0.7

random_effect <- data.frame(seed=c(),gradient=c(),stddev = c(),
    MAE=c())
for (k in 1:10) {
  set.seed(k)
  draw <- runif(nrow(allresults))
  ind_train <- draw < parm$traintestratio
  ind_test <- draw > parm$traintestratio

  train <- allresults[ind_train,]
  test <- allresults[ind_test,]

  reg_originalI <- lm(I(J-I/2) ~ h_dat-1, data=train) #am(I ~ I(
      XV*sigma)-1, data=train)
  predI_original <- predict(reg_originalI,newdata=test)
  null <- mean(abs(test$I))
  maeI <- mean(abs(test$I-predI_original),na.rm=TRUE)/null

  newline <- data.frame(seed=k,gradient = summary(reg_originalI)
      $coefficients[1],stddev = summary(reg_originalI)$
      coefficients[2], MAE = maeI)
  random_effect <- rbind(random_effect,newline)
}
random_effect <- data.frame(random_effect,row.names="seed")

xtable(random_effect,digits=3, caption = "Liquidity gradients
    with different seeds",label = "table:seedchanger")

@

<<temp-all>>=
  reg_alltemp <- lm(I(J-I/2) ~ h_dat-1, data=allresults) #am(I ~
       I(XV*sigma)-1, data=train)
summary(reg_alltemp)

#plot(with(allresults,J-I/2)/predict(reg))

@

<<temporary-part-cv>>=
# Estimate permanent impact
#allresults$I <- allresults$XV*allresults$sigma*allresults$gamma

# REGRESSION:
require(mgcv)
b <- nrow(allresults)
allresults_cv <- allresults[sample(1:b,b),]
```

```
pred_crossvalid <- c(); coeff_cv <- c()
for (i in 1:(b/10) ) {
  ind_testcv <- (10*(i-1)+1) : (10*(i-1)+10)
  train_cv <- allresults_cv[-ind_testcv,]
  test_cv <- allresults_cv[ind_testcv,]

  reg <- gam(I(J-I/2) ~ h_dat-1,data=train_cv)
  predJ_cv <- predict(reg,newdata=test_cv)
  pred_crossvalid[i] <- mean(abs(test_cv$J-test_cv$I/2-predJ_cv)
      )
  coeff_cv[i] <- reg$coefficients
}


J_cv_all <- list(coeff = coeff_cv,
                 MAE = pred_crossvalid/mean(abs(allresults_cv$J-0.5*
                     allresults_cv$I)))

plot(pred_crossvalid/mean(abs(allresults_cv$J-0.5*allresults_cv$
    I)))
plot(coeff_cv)
mean(pred_crossvalid)/mean(abs(allresults_cv$J-0.5*allresults_cv
    $I))
eta <- mean(coeff_cv)

ticker = c("STL","YAR","SCH")
error = c()
for (i in 1:3) {
ind0 <- which(allresults$stock == ticker[i])
error[i] <- with(
      allresults[ind0,],
      mean(abs(J-I/2-eta*h_dat)) / mean(abs(J-I/2))
     )
}

error_Jall_cv <- mean(pred_crossvalid)/mean(abs(allresults_cv$J
    -0.5*allresults_cv$I))


J_cv_all$temperrors <- data.frame(ticker = ticker, MAE = round(
    error,3))
xtable(temperrors,digits=3)

@



<<save>>=
save(reg_alltemp,J_cv_all,error_Jall_cv, file = "../data/
    temporary_effect_all_data.Rdata")
@



\end{document}
```

.1.4 Temporary filter source code

The code chunk is used to produce the temporary filter in section 3.6.4

```
\documentclass{article}

\begin{document}

<<load-needed-data>>=
#d = 1

RES <- list()
SUP <- list()

# Statoil
load('../data/bigdiff_compiled_STL_v4.Rdata')
RES[[1]] <- resultater[order(resultater$startdate,resultater$
    startslot),]
RES[[1]]$ticker <- "STL"
SUP[[1]] <- supportdata

# Schibsted
load('../data/bigdiff_compiled_SCH_v4.Rdata')
RES[[2]] <- resultater[order(resultater$startdate,resultater$
    startslot),]
# [-which(resultater$startdate == '2014-01-16'),]
SUP[[2]] <- supportdata[-which(supportdata$tickdate == '
    2014-01-16'),]
RES[[2]]$ticker <- "SCH"

# # Yara
load('../data/bigdiff_compiled_YAR_v4.Rdata')
RES[[3]] <- resultater[order(resultater$startdate,resultater$
    startslot),]
SUP[[3]] <- supportdata
RES[[3]]$ticker <- "YAR"

masterdate <- supportdata$tick
# masterdate <- masterdate0[-which(masterdate0 == '2014-01-16')]

RESA <- rbind(RES[[1]],RES[[2]],RES[[3]])

RESA = RESA[order(RESA$startslot),]

resultater <- RESA[1,]
for (t in 1:max(RESA$startslot)) {
        ind0 <- which(RESA$startslot == t)

                #if (length(ind0) == 0) #resultater[t,] <- 0
                # If multiple observation on same slot, choose
                    the biggest trade
```

```
                              if (length(ind0) >= 1) {
                                        ind0 <- ind0[which.max(abs(RESA$
                                            XVT[ind0]))]
                                        resultater[t,] <- RESA[ind0,]
                              }
}

# Remove zero pos
resultater <- resultater[-which(is.na(resultater$XV)),]

d = 1#length(RES)
tot <- length(masterdate)
N = tot

#
#resultater <- RESA

@


<<>>=
L = nrow(resultater)
A = list()
Q = c()
R = c()

n = 1
R[1] = resultater$sigma[1]**2
Q[1] = R[1]/100
A[[1]] <- 0

@


<<>>=
# start predicting in state k=2
xbarmin <- mat.or.vec(L,n)
xbar <- mat.or.vec(L,n)
xbar[1] <- 0.1
Pmin <- list()
P <- list()
P[[1]] <- 0.04**2
K = list()

for (k in 2:L) {
  A[[k]] <- resultater$h_dat[k]
          signal <- resultater$J[k]-0.5*resultater$I[k]
# Time update (predict)
xbarmin[k] <- max(0,xbar[k-1]) # Add floor at zero
Pmin[[k]] <- P[[k-1]] + Q[[k-1]]

# Measurement update (correct)

K[[k]] <- Pmin[[k]]%*%t(A[[k]]) %*% 1/(A[[k]]%*%Pmin[[k]]* t(A[[
     k]]) + R[k-1])
```

```
xbar[k] <- xbarmin[k] + K[[k]] %*% (signal- A[[k]]%*%xbarmin[k])
P[[k]] <- (diag(n) - K[[k]]%*%A[[k]]) %*% Pmin[[k]]
Q[[k]] <- Q[[k-1]] #var(xbar[1:k])
R[k] <- resultater$sigma[k]**2 * resultater$tau[k]
}




@

<<estimate-MAE-temp>>=

# Predicton
l = length(xbarmin)
nullsignals <- abs((resultater$J-0.5*resultater$I))
MAE_filter_null <- mean(nullsignals)
modelsignals <- abs(xbarmin*resultater$h_dat - (resultater$J-0.5
    *resultater$I))
MAE_filter_J <- modelsignals/MAE_filter_null

#library(zoo)

q = round(seq(1,l,length.out=6))

tickshare = list()
posshare <- c()
smoothMAE <- c()
for (i in 1:(length(q)-1)) {
smoothMAE[i] <- mean(modelsignals[q[i]:q[i+1]] / mean(
    nullsignals[q[i]:q[i+1]]))

tickshare[[i]] <- table(resultater$ticker[q[i]:q[i+1]])
tickshare[[i]] <- tickshare[[i]]/sum(tickshare[[i]])

posshare[i] <- mean(abs(resultater$XVT[q[i]:q[i+1]]))
}

smoothMAE

mean(smoothMAE)
mean(MAE_filter_J)

plot(unlist(K))

@

<<save_to_thesis>>=

temporary_filter <- data.frame(
  date = resultater$startdate,
  xbarmin = xbarmin,
  P = unlist(P),
  Q = unlist(Q),
```

```
  MAE = MAE_filter_J,
  modelsignals,
  nullsignals
  )

save(temporary_filter, file="../data/thesis_filter.Rdata")
@

<<plots-temporary>>=
par(mfrow=c(2,1))

plot(resultater$startdate,xbar, type = "l",  ylim = c(-.05,0.4)
    ,
     main = "Estimate of liquidity gradient",
     xlab = "Date",
     ylab = "eta estimate")
polygon(c(resultater$startdate, rev(resultater$startdate)), c(
    xbar+sqrt(as.double(P)), rev(xbar-sqrt(as.double(P)))), col =
     "grey40", border = NA)
lines(resultater$startdate,xbar, type = "l", main = "Estimate of
     liquidity gradient")

plot(unlist(P)**.5, type = "l", main = "Estimate of conditional
    error covariance", ylim = c(0,max(unlist(P)**.5)))

#plot((resultater$I/unlist(A)), type = "o", main = "Permanent
    Impact signal (y_t/A_t)")

#plot(unlist(K), type = "l", main = "Kalman Gain")
@

\end{document}
```

.1.5 MSDC surface example

The following code calculates the value of a portfolio given a msdc, and is used in section 2.

```


#### VALUE FUNCTION ####
#proceed <- function(pi,V,tau,sigma,gamma,eta,S0,ep = 0) S0*pi
    *(1+ep) - sign(pi)*sigma*(0.5*gamma*abs(pi**2/V) + V*tau*eta*
    abs(pi/(V*tau))**(8/5))

# proceed(2e6,15e6,0.5,0.2,1,0.2,100,ep=-.05) / (2e6*100)
#
# P = rep(1e6,3)
# a = 10e6

# Finds the acerbi-value of a portfolio given that p0<a:

#### Functions ####
```

```
# msdc
  msdc_full <- function(sv,tau,sigma,gamma,eta,S0) S0 + S0*sign
      (-sv)*sigma * (gamma*abs(sv) + 8/5*eta*abs(sv/tau)**(3/5))
  msdc_full_pars <- function(x,parsi) msdc_full(x,parsi$tau,
      parsi$sigma,parsi$gamma,parsi$eta,parsi$S0)
# inverse of msdc

  msdc_inv <- function(output,parsi) {
    m <- function(x) msdc_full(x/parsi$V,parsi$tau,parsi$sigma,
        parsi$gamma,parsi$eta,parsi$S0)-output
    tryCatch(uniroot(m,c(-parsi$V,parsi$V))$root,error=function(
        e) NA)
  }
  msdc_inv <- Vectorize(msdc_inv,"output")

# The proceed
  #proceed <- function(pi,pari) pari$S0*pi - sign(pi)*pari$sigma
      *(0.5*pari$gamma*abs(pi**2/pari$V) + pari$V*pari$tau*pari$
      eta*abs(pi/(pari$V*pari$tau))**(8/5))

proceed <- function(pi,pari) pi*pari$S0* (1-0.5*pari$gamma*pari$
    sigma*pi/pari$V - sign(pi)*pari$eta*pari$sigma*abs(pi/(pari$
    tau*pari$V))**(3/5))

# Given a lambda, find r-vector (used as internal function in
    optimization)
rootV <- function(lambda,pars,a,p0 = 0) {
  proceed_sum = 0
  r = c()
  for (k in seq_along(pars) ) {
    r[k] <- msdc_inv(pars[[k]]$S0/(1+lambda),pars[[k]])
    proceed_sum = proceed_sum + proceed(r[k],pars[[k]])
  }

  list(
    proceed_sum = proceed_sum,
    rootval = a-2*p0-proceed_sum,
    r = r)
}

# The Acerbi value function:
value <- function(pos,pars,a) {

# First find optimal lambda
  f <- Vectorize(function(l) rootV(l,pars,a)$rootval) # root
      function to find lambda
  # Find max possible lambda:
  lint = c(0,suppressWarnings(optim(0,f)$par))
  lambda <- uniroot(f,lint,tol=.Machine$double.eps**.5)$root
# Calculate r-vector:
  info = rootV(lambda,pars,a) # Round down to integer?

# Value remaining portfolio Mtm:
```

```
remaining_mtmpos <- pos - info$r
mtmprice <- sapply(pars,function(x) x$S0)
aval <- sum(remaining_mtmpos*mtmprice) + info$proceed_sum
mtm_liq_val <- sum(info$r*mtmprice)
mtmval <- sum(pos*mtmprice)

list(r = (info$r),
      lambda = lambda,
      acerbi_value = aval,
      mtm_value= mtmval,
      mtm_liq_val = mtm_liq_val)
}


#### INPUT PARAMETERS ####
# Base info on underlying stocks (NB: some parameters must be
    equal on all!)
STLinfo = list(
  S0 = 100,
  gamma = 1.1,
  eta  =0.1,
  V = 2e6,
  sigma = 0.1,
  tau = 0.2)

YARinfo = list(
  S0 = 100,
  gamma = 0.9,
  eta  =0.1,
  V = 0.5e6,
  sigma = 0.15,
  tau = 0.2)

SCHinfo = list(
  S0 = 100,
  gamma = 0.2,
  eta  =0.1,
  V = 1e5,
  sigma = 0.2,
  tau = 0.2)

pars = list(STLinfo=STLinfo,YARinfo,SCHinfo)
#### Test of functions ####
# Check that we actually get the inverse:
pari = pars$STLinfo
prices <- seq(99,101,0.1)
inv <- msdc_inv(prices,pari)
msdc_full(inv/pari$V,pari$tau,pari$sigma,pari$gamma,pari$eta,
    pari$S0)
plot(prices,msdc_inv(prices,pars$STLinfo),type="l")
```

```
# Test whether the derivative of the proceed is m numerically:
s = pari$V*seq(-.2,.2,0.05)
m <- msdc_full(s/pari$V, pari$tau, pari$sigma, pari$gamma, pari$eta,
    pari$S0)
proc <- proceed(pi = s, pari)
procderiv <- diff(proc)/diff(s)
s_num <- s[-1] - diff(s)/2 # the derivative is "best valued" at
    the between points in s
plot(s,m, type = "p")
lines(s_num, procderiv)
abline(0,1); abline(100,0)
# OK!


#### value plots ####
pars = list(STLinfo = STLinfo) #SCHinfo=SCHinfo, YARinfo=YARinfo
    , SCHinfo=SCHinfo
  a_vals <- seq(1e5,5e7,5e6) #c(1e6,5e6,7e6,1e7,3e7,5e7,8e7,1e8)
  pval=c();lambda_temp=c(); r=c()
val_temp <- list()
  for (i in seq_along(a_vals)) {
    val_temp[[i]] <- value(pos = rep(0e6, length(pars)),pars,a =
        a_vals[i])
    lambda_temp[i] <- val_temp[[i]]$lambda
    pval[i] <- val_temp[[i]]$acerbi_value#/val_temp$mtm_liq_val
    r[i] <- sum(val_temp[[i]]$r)
  }


plot(a_vals,r/pars$STLinfo$V, type = "l")
lines(a_vals,a_vals*1/100/pars$STLinfo$V)

par(mfrow=c(2,1))
plot(a_vals,pval, type = "l", main = "Increasing the need for
    Cash",
     xlab = "Cash liquidity policy", ylab = "Value of portfolio"
         )


r_deriv <- 100*diff(r)/diff(a_vals)
a_num <- a_vals[-1]-diff(a_vals)/2
plot(a_num,r_deriv,type = "l",
     main ="dr/da",
     sub = "Incremental extra need of liquidating stocks given
         incremental increase in cash liq. policy",
     xlab = "Cash liquidity policy",
     ylab = "dr/da [*100]")
par(mfrow=c(1,1))
# Comment: If needed liquidity is increased by one, how many
    extra stocks are needed?
#  It is not constant as your buying/selling price is getting
    worse and worse.
# The derivative shows it well. You start with a 100/1 ratio but
     ends up on almost 101/1 ratio.
```

```
deriv_pval <- diff(pval)/diff(a_vals)
a_vals_num <- a_vals[-1] - diff(a_vals)/2
plot(a_vals_num,deriv_pval,type="l",
     xlab = "Liquidity policy",
     ylab = "Marginal change in portfolio value",
     main = "Marginal decrease in portfolio value given
         incremental increase in cash liq. policy")

# Comment: When Cash liquidity Policy is increased, what is the
    value of the portfolio?
#        Observe: not linear due to two effects:
           # 1 - Your selling price is deteriorating
           # 2 - More and more stocks goes from being evaluated
               by the mtm price to the execution price (due to
               definition of value of portfolio)


deriv2_pval <- diff(deriv_pval)/diff(a_vals)[-1]
plot(r[-(1:2)],deriv2_pval,type = "l", main ="d**2(V) / da**2 :
    Second derivative is essentially zero")
# Comment: Second order is zero.

#### MONTE CARLO SIMULATION eta, gamma, sigma ####

# Assume a portfolio of one stock with mtm-value of 10k
# indexes:
# (sigma, gamma, eta)

# Stats is a named vector with (S0, gamma, eta) values on each
    row
acerbi_value <- function(stats,tau=0.2,pos = 0) {
  stock_temp = list(
    S0 = as.double(stats['S0']),
    gamma = as.double(stats['gamma']),
    eta  = as.double(stats['eta']),
    V = 2e6,
    sigma = 0.1,
    tau = tau)

  val_temp <- value(pos = pos,list(stock_temp),a = as.double(
      stats['a']))
  val_temp$acerbi_value}

N = 5000
A = 2e6*100*0.1
vol0 <- c(0.1,0.2,0.2)
initial_values <- c(100,1.1,0.1)
portfolio_values <- list()
library("MASS")

# No liquidity model (mtm-value)
cor0 <- diag(3)
```

```
cov0 <- outer(vol0,vol0) * cor0
returns <- mvrnorm(N,mu=rep(0,length(vol0)), Sigma=cov0)
statevar <- t(initial_values * t(1+returns))
R <- data.frame(statevar,a=A)
colnames(R) <- c("S0", "gamma", "eta","a")
portfolio_values$noliq <- R$S0*(0) # Sell without liquidity msdc

# With constant liquidity parameters
vol_const <- c(0.1,0,0)
cor0 <- diag(3)
cov0 <- outer(vol_const,vol_const) * cor0
returns <- mvrnorm(N,mu=rep(0,length(vol_const)), Sigma=cov0)
statevar <- t(initial_values * t(1+returns))
R <- data.frame(statevar,a=A)
colnames(R) <- c("S0", "gamma", "eta","a")

val <- apply(R,1,acerbi_value)
portfolio_values$noliqvol <- val

# With zero correlation
  cor0 <- diag(3)
  cov0 <- outer(vol0,vol0) * cor0
  returns <- mvrnorm(N,mu=rep(0,length(vol0)), Sigma=cov0)
  statevar <- t(initial_values * t(1+returns))
  R <- data.frame(statevar,a=A)
  colnames(R) <- c("S0", "gamma", "eta","a")
  val <- apply(R,1,acerbi_value)
  portfolio_values$zerocor <- val

# With positive correlation
# library("MASS")
# cor0 <- mat.or.vec(3,3)+.3 + diag(3)*(1-0.3)
# cov0 <- outer(vol0,vol0) * cor0
# returns <- mvrnorm(N,mu=rep(0,length(vol0)), Sigma=cov0)
# statevar <- t(initial_values * t(1+returns))
# R <- data.frame(statevar,a=A)
# colnames(R) <- c("S0", "gamma", "eta","a")
# val <- apply(R,1,acerbi_value)
# portfolio_values$poscor <- val


# With negative cor
  library("MASS")
  a = c(1,-.5,-.5)
  b = c(-.5,1,.5)
  c = c(-.5,.5,1)
  cor0 <- rbind(a,b,c)
  cov0 <- outer(vol0,vol0) * cor0
  returns <- mvrnorm(N,mu=rep(0,length(vol0)), Sigma=cov0)
  statevar <- t(initial_values * t(1+returns))
  R <- data.frame(statevar,a=A)
  colnames(R) <- c("S0", "gamma", "eta","a")
  val <- apply(R,1,acerbi_value)
  portfolio_values$negcor <- val
```

```
ES <- function(x,q=0.05) mean(x[which(x<=quantile(x,q))])
exp <- round(sapply(portfolio_values,mean)/A,3)
distr_stats <- data.frame(
Expected_value = exp
,Stddev = round(sapply(portfolio_values,sd)/A,3)
, VaR95 = exp-round(sapply(portfolio_values,function(x) quantile
    (x,0.05))/A,3)
,CVar95 = exp-round(sapply(portfolio_values,ES)/A,3)
)

rownames(distr_stats) <- paste0("Model ",c(1,2,3,4))
distr_stats
library(xtable)
xtable(distr_stats,digits=3)

par(mfrow=c(2,3))
sapply(portfolio_values,qqnorm)
par(mfrow=c(1,1))

lines(density(portfolio_values$noliq),xlim = c(-0.5e6,1.4e4))
lines(density(portfolio_values$noliqvol))
lines(density(portfolio_values$zerocor))
plot(density(portfolio_values$negcor))

  # Comments:
  # The mark-to-market model, with a constant msdc(x)=S0, gives
      the predicted mean(V) = 10k
  # When extending to a constant msdc-model, the mean value
      decreases.
  # The three 3-factor models (zerocor,poscor and negcor)...
  # When selling stock, a negative correlation between stock and
      liquidity gradients is the worst effect: seen in figures.
  # zero cor or small positive cor does not seem to have a big
      effect.


#### Portfolio value as function of execution time ####

taus <- c(0.05,0.1,0.15,0.2,0.4,0.6,1,1.5,2)
#taus <- c(0.01,0.1,0.4,0.6,1,2)

#taus <- c(0.05,0.5,0.8,1,2)
N = 1000
A = 2e6*100*0.1
vol0 <- c(0.03,0.3,0.3)
initial_values <- c(100,1.1,0.1)

a = c(1,-.5,-.5)
b = c(-.5,1,.5)
c = c(-.5,.5,1)
cor0 <- rbind(a,b,c)
cov0 <- outer(vol0,vol0) * cor0
```

```
portfolio_value_tau <- mat.or.vec(nr=N, nc=length(taus))
returns <- mvrnorm(N,mu=rep(0,length(vol0)), Sigma=cov0)

for (i in seq_along(taus)) {
  statevar <- t(initial_values * t(1+returns*sqrt(taus[i])))
  statevar[which(statevar <0)] <- 0.01
  R <- data.frame(statevar,a=A)
  colnames(R) <- c("S0", "gamma", "eta","a")

portfolio_value_tau[,i] <- apply(R,1, function(r) acerbi_value(r
    ,tau=taus[i]))
}

# probs <- seq(0.01,0.99,0.01)
# H <- apply(portfolio_value_tau,2,function(x) quantile(x,prob=
    probs))
# matplot(t(H)/A*100,type = "l")
# contour(H/A*100)
# library(rgl)
# persp(y = taus,x=probs,H/A*100,theta=-45,
#        zlab = "Portfolio value [pct]",
#        xlab = "Quantile",
#        ylab = "Execution time",
#        main = "Empirical CDFs of portfolio value for all
    execution times",ticktype="detailed",shade=TRUE
#        )
#


# distr_stats_tau <- data.frame(
# expected_value=apply(portfolio_value_tau,2,mean)
# ,var05 = apply(portfolio_value_tau,2,function(x) quantile(x
    ,0.1))
# ,var01 = apply(portfolio_value_tau,2,function(x) quantile(x
    ,0.01))
# ,var99 = apply(portfolio_value_tau,2,function(x) quantile(x
    ,0.99))
# ,var95 = apply(portfolio_value_tau,2,function(x) quantile(x
    ,0.90))
# )
# plot(taus,distr_stats_tau$expected_value/A*100,type = "l",ylim
    =range(distr_stats_tau/A*100))
# polygon(c(taus,rev(taus)),100/A*c(distr_stats_tau$var01,rev(
    distr_stats_tau$var99)),col="grey80",border=FALSE)
# polygon(c(taus,rev(taus)),100/A*c(distr_stats_tau$var05,rev(
    distr_stats_tau$var95)),col="grey70",border=FALSE)
# lines(taus,100/A*distr_stats_tau$expected_value, col = "blue")

save.image(file="../data/msdc_surface_example_computation.Rdata"
    )
```