Modelling Intraday Trading Volume

CHRISTOPHE AGOPIAN

MSc by Research in Pattern Analysis and Neural Networks

ASTON UNIVERSITY

October 2004

ASTON UNIVERSITY

Modelling Intraday Trading Volume

CHRISTOPHE AGOPIAN

MSc by Research in Pattern Analysis and Neural Networks, 2004

Thesis Summary

As a brokerage activity, some investment banks provide volume-weighted average price (VWAP) orders to their clients. In order to build VWAP trading strategies, one crucial step is to gain knowledge of the intraday trading volume patterns as well as to track tick prices for order submissions. This thesis describes the deterministic modelling of the volume behaviour coupled with the stochastic modelling of the volume dynamics. We also investigate whether we can gain a better insight by using a whole sector approach rather than considering just a single stock. Afterwards, we discuss further research that could be carried out.

Keywords: volume-weighted average price trading, financial time series prediction, pattern recognition, Bayesian inference, kernel methods.

Acknowledgements

I am particularly grateful to my project supervisors, Dr Mehdi Azzouzi and Prof Ian Nabney, for accepting to supervise a finance oriented project as I wished. Their refined knowledge and deep expertise have provided me with enlighting guidance.

I wish to thank the Neural Computing Research Group for allowing me to benefit from its tremendous knowledge and familial atmosphere, especially my course's lecturers Prof David Lowe, Prof David Saad, Prof Ian Nabney and Dr Manfred Opper. I would like to thank also Dr Laura Rebollo-Neira for the pleasant late night discussions and Dr Davide D'Alimonte for his kindness and enthusiasm. I wish to thank Ms Vicky Bond for her efficiency concerning administrative matters.

Thanks to my course's fellows for the good time we shared, Youssef, Thomas, Youenn, Kiriko, Dharmesh, and Pierre, and the PhD students, Ben and Dan. I would like also to thank Kleopatra for having been such a pleasant flatmate and friend.

Thanks to my parents, who provide me financial support to make this year possible.

Special thanks to Marie for her understanding, support and encouragement when it was the most required.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Foreword

This thesis is the report of research work carried out from January to August 2004 at Aston University's Neural Computing Research Group, as part of a Master of Science by Research in Pattern Analysis and Neural Networks course.

All the work has been done in close collaboration with Dr Mehdi Azzouzi (Azzouzi 2004) for the account of a leading investment bank in London, and all the research was driven by a real-life problem.

The goal of the project is to model intraday trading volume patterns and dynamics for it to be used to build Volume-Weighted Average Price (VWAP) trading strategies.

1.2 Volume-weighted average price

As a brokerage activity, some big investment banks provide volume-weighted average price – commonly called VWAP – orders for its clients. Therefore, the aim of the traders is to deal better prices than the VWAP over the trading horizon. VWAP can be regarded as a proxy for the real market price of the asset considered.

Following the notation in (Azzouzi 2003a), a given trading period of length N consists of $1, \ldots, N$ time windows of the same length, measured as the number of seconds or minutes. We define V_t as the total trading volume, defined by the number of transactions, of a given asset during the time window t. Similarly, p_t denotes the execution price of the security from time t to $t+1$ (if t is sufficiently small to represent any transaction, to be more precise). Hence a formal definition of the VWAP is

$$VWAP_N = \frac{\sum_{t=1}^{N} V_t p_t}{\sum_{t=1}^{N} V_t}. \tag{1.1}$$

Currently, VWAP orders are mainly executed manually, although some big investment banks claim to have developed VWAP engines. Due to the competitiveness of the business, banks keep their processes secret. These supposed automates take advantage of the newest method of achieving

VWAP, which is to use trading strategies to participate proportionately throughout the trading day, trading as intelligently as possible and with minimal market impact. For example, orders can be split up for execution over the day in accordance with the historical volume 'smile' or pattern. Hence, the volume shape as function of the intraday time has a crucial role in VWAP strategies.

The commercial value of a VWAP machine must be measured in terms of profit and loss of the trading desk. If the machine beats the market, there would be a profit of 3 basis points[1] per order (considering charges of client and transaction costs). To give an idea of this emerging market, GOLDMAN SACHS receive VWAP orders for a daily average value of 1.5 billion euros in Europe. Assuming the same amount for the US and an execution price of 10 bps, this comes to a total of 750 million euros per year for the VWAP brokerage activity.

1.3 Motivation

A large number of studies deal with financial time series, but usually are concerned with stock prices and volatility. Much less attention has been devoted to the investigation of the dynamics and patterns of the number of trades of a given asset.

Knowing that almost no research has been carried out yet concerning trading volume prediction, and due to its importance for VWAP trading, the aim of this thesis is to study intraday trading volume.

1.4 High-frequency time series

Nowadays, datasets are available on the scale of seconds, which represents tens of thousands of transactions or posted quotes[2] in a single day, time stamped to the nearest second. It means that any statistical model, or theoretical idea, can and must be tested against available data.

1.4.1 Properties

High-frequency data have the following characteristics (Engle and Russell 2002):

irregular temporal spacing: virtually all transaction data are inherently irregularly spaced in time,

discreteness: price changes must fall on multiples of the smallest allowable price change called a 'tick',

patterns: for most stock markets volatility, frequency of trades, volume, and spreads all typically exhibit a U-shaped pattern over the course of the day; the time between trades tends to be shortest just after the opening and just prior to the closing of the market,

[1] 1 basis point (bp) is equal to 0.01%.
[2] A quote is a collection of data relative to a buy or a sell of a security on a stock exchange.

temporal dependence: high-frequency data tends to exhibit volatility clustering; large price changes tend to follow even larger price changes.

1.4.2 Tick-time models

Due to the fact that one of the most salient features of high-frequency transaction data is that transactions do not occur at regularly spaced time intervals, a large amount of statistical literature has been produced studying and applying (marked) point processes. The research community proposed the Autoregressive Conditional Duration (ACD) model (Engle and Russell 1998) and variants. Let $x_i = t_i - t_{i-1}$ be the duration between two bid-ask quotes occurring at times t_{i-1} and t_i. The assumption introduced in (Engle and Russell 1998) is that the time dependence in the durations can be subsumed in their conditional expectations, $\Psi_i \equiv \mathrm{E}[x_i|\mathcal{F}_{i-1}]$, in such a way that $x_i/\mathrm{E}[x_i|\mathcal{F}_{i-1}]$ is independent and identically distributed (iid), where $\mathcal{F}_{i-1}$ denotes the information set available before time t_i, which includes at least the past durations. Thus, the ACD model specifies the observed duration as

$$x_i = \Psi_i \epsilon_i, \tag{1.2}$$

where the ϵ_i are positive iid random variables. A second equation specifies an autoregressive model[3] for the conditional durations Ψ_i:

$$\Psi_i = \omega + \sum_{k=1}^{p} \alpha_k \epsilon_{i-k} + \sum_{k=1}^{q} \beta_k \Psi_{i-k}. \tag{1.3}$$

Another approach is to aggregate high-frequency data into time series of five-minute intervals, and hence to use the widely used Autoregressive Conditional Heteroskedasticity (ARCH) type models first introduced by (Engle 1982) for modelling the predictive variance for UK inflation rates. An ARCH model with order $p(\geq 1)$ is defined as

$$x_i = \sigma_i \epsilon_i, \tag{1.4}$$

where ϵ_i is zero mean unit variance iid random variable, and

$$\sigma_i^2 = \omega + \sum_{k=1}^{p} \beta_k x_{i-k}^2. \tag{1.5}$$

Engle was awarded The Bank of Sweden Prize in Economic Sciences in Memory of Alfred Nobel 2003 "for methods of analyzing economic time series with time-varying volatility (ARCH)".

1.5 Thesis outline

Chapter 2 We describe the dataset used to carry out the research and its preprocessing, especially the outlier detection.

[3]This model is the ACD(p,q) where p and q refer to the order of the lags.

Chapter 3 Assuming a model combining a deterministic part and stochastic part for intraday volume, we review the methods and the models of the deterministic part. The main techniques used are splines and we highlight the link between these and Bayesian priors which finally leads us to the use of kernel methods, especially Gaussian processes.

Chapter 4 We tackle the stochastic modelling. We compute residuals from the deterministic fit from the previous chapter. We fit the empirical density and evaluate the autocorrelation of these residuals. Then, we focus on the one-step ahead prediction of the residuals using leading edge techniques in a phase space reconstruction framework.

Chapter 5 Instead of considering a single stock, we use a whole sector view, such as the UK bank sector, in order to gain some insight into the underlying forces that drive the dynamics studied.

Chapter 6 We end the thesis by summarising the work presented and providing some future directions of research.

Figure 1.1 represents the logical flow of the thesis.

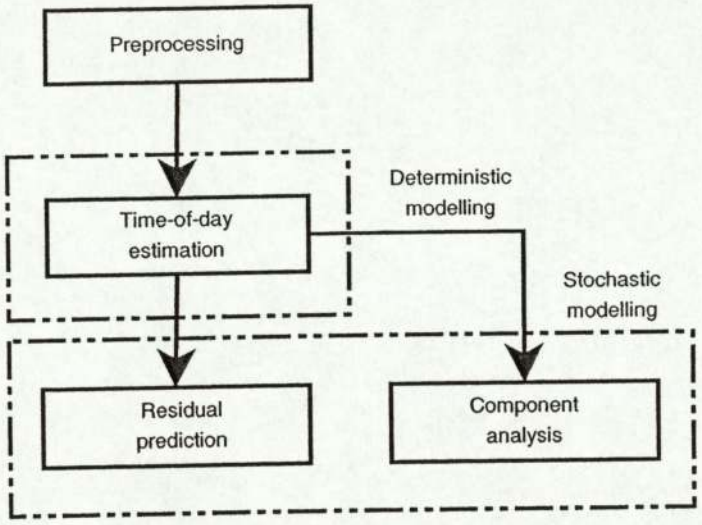


Figure 1.1: Thesis logical flow.

Chapter 2

Datasets and Preprocessing

Since this research is carried out in an empirical manner, understanding and preprocessing the dataset are important.

2.1 Data description

Our datasets concern:

- the forty stocks compounding the French CAC40 index. These stocks are the forty largest French firm capitalisations, hence they are expected to be very liquid,
- seventeen stocks of the French SBF120 index, which is compounded of illiquid stocks,
- ten stocks of the UK bank sector.

French market consist of 5-minute tick volumes, collected from 8:00 till 16:30 (UK time) during 935 days from the 4th of January 2000 to the 30th of September 2003[1].

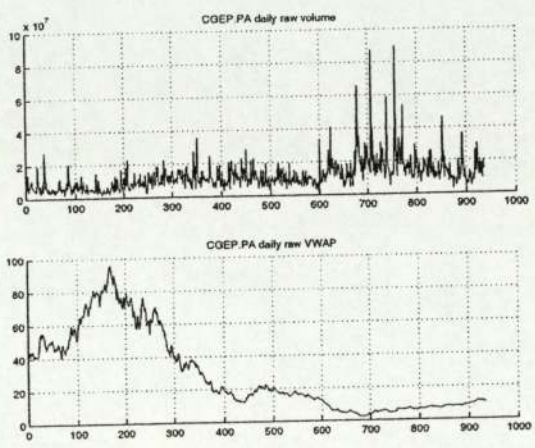
Figure 2.1 plots the daily volume and the daily VWAP of different stocks over the period. It clearly demonstrates some outliers, examples which did not (or are thought not to have) come from the assumed population of examples, which should be discarded as we will see later (see Section 2.2.2).

Although, the dataset is 5-minute tick frequency, it is possible to generate any volume and VWAP $5 \times N$-minute tick frequency time series. This study will stay in a 5-minute tick frequency framework since the forecast of interest is under this trading time horizon.

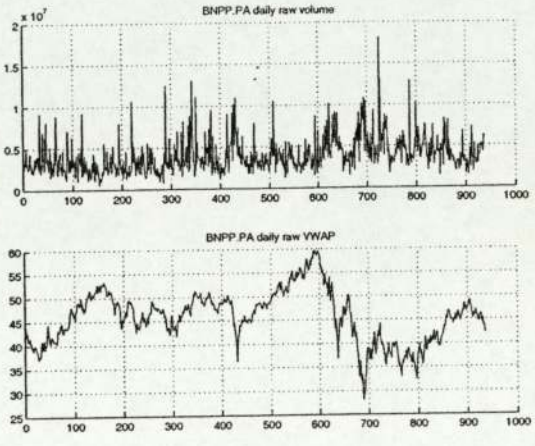
2.1.1 Summary statistics

Table 2.1 presents some statistics about the raw data. These statistics are calculated after having removed abnormal periods (see Section 2.2.1) of the dataset.

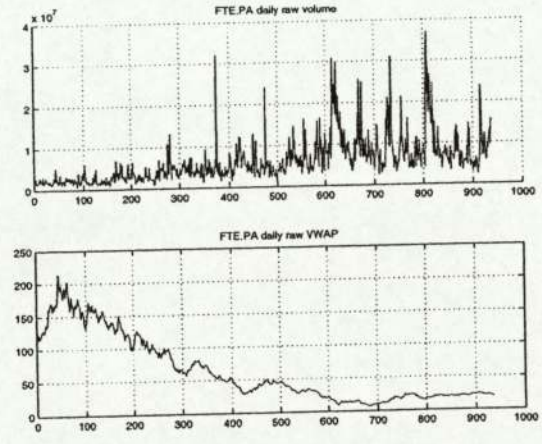
[1]Similar periods in the case of the UK market.



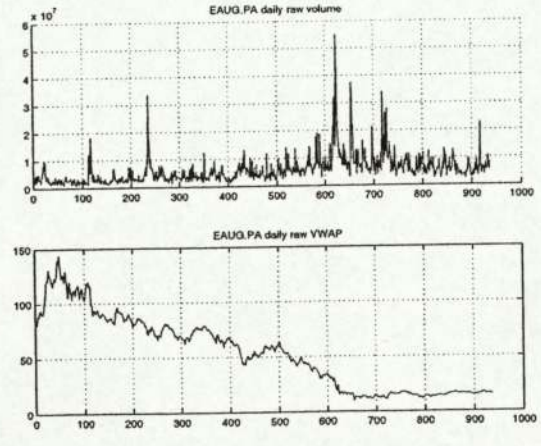
(a) Alcatel (CAC40)



(b) BNP-Paribas (CAC40)



(c) France Télécom (CAC40)



(d) Vivendi Universal (CAC40)

Figure 2.1: Raw volume and daily VWAP of different stocks during the period of our study.

Skewness and kurtosis, respectively, are calculated as follows:

$$\lambda_3 = \frac{\mathrm{E}[(X-\mu)^3]}{\sigma^3}, \tag{2.1}$$

$$\lambda_4 = \frac{\mathrm{E}[(X-\mu)^4]}{\sigma^4} - 3, \tag{2.2}$$

with $\mu = \mathrm{E}[X]$ and $\sigma^2 = \mathrm{E}[(X-\mu)^2]$ the expectation and the variance, respectively, of the random variable X. Skewness is a useful measure of the asymmetry of the probability density function (pdf). Clearly the skewness is zero for probability densities that are symmetric around their mean. This is not the case for this data. Kurtosis is the simplest statistical quantity for indicating the nongaussianity of a random variable. Random variables that have a Gaussian distribution have a zero kurtosis. Here, the data clearly has an enormous kurtosis (from 6 up to 38), which reflects the fact that the data distribution has a sharper peak and a longer tail[2] than the Gaussian pdf.

day	# days	μ	σ	min	max	λ_3	λ_4
Alcatel							
Monday	174	8.66	5.20	2.25	37.94	2.08	6.56
Tuesday	185	11.53	8.77	1.58	84.65	3.99	26.57
Wednesday	191	12.32	8.85	2.15	83.70	3.65	22.6
Thursday	193	11.35	6.60	1.94	47.72	1.86	6.51
Friday	176	11.14	6.79	2.34	52.86	2.18	8.34
all days	919	11.04	7.49	1.58	84.65	3.39	22.85
BNP-Paribas							
Monday	174	3.02	1.88	0.47	16.5	3.36	17.43
Tuesday	185	3.66	1.75	0.94	10.0	1.23	1.35
Wednesday	191	3.81	1.80	1.04	11.0	1.33	1.83
Thursday	193	3.84	1.83	0.85	12.8	1.55	3.55
Friday	176	3.52	1.70	0.97	9.93	1.30	1.83
all days	919	3.58	1.81	0.47	16.5	1.73	4.98
France Télécom							
Monday	174	4.91	4.42	0.62	31.0	3.19	13.02
Tuesday	186	5.96	4.86	0.89	31.0	2.42	7.80
Wednesday	191	5.97	4.63	0.84	33.7	2.26	8.09
Thursday	193	6.01	4.65	0.98	29.4	2.33	7.13
Friday	176	5.55	4.56	0.93	29.1	2.42	7.32
all days	919	5.69	4.64	0.62	33.7	2.49	8.41
Vivendi Universal							
Monday	174	4.37	3.23	0.73	19.7	2.50	7.46
Tuesday	186	5.73	5.17	1.03	53.5	4.94	38.41
Wednesday	191	6.10	5.06	0.98	41.5	3.14	15.01
Thursday	193	5.58	4.76	1.02	36.0	3.69	17.92
Friday	176	5.17	3.79	1.03	29.8	2.95	12.84
all days	919	5.41	4.52	0.73	53.5	3.92	25.54

Table 2.1: Statistics on Alcatel, BNP-Paribas, France-Télécom, Vivendi Universal raw volumes. Volume related quantities are expressed in millions of shares. Skewness and kurtosis are dimensionless.

The daily turnover as a measure of liquidity is shown in Table 2.2, in order to exhibit that the stocks we are looking at are very liquid.

[2]This kind of distribution is called supergaussian or leptokurtic.

Stock	Daily turnover mean
Alcatel	243.67
BNP-Paribas	161.05
France Télécom	229.60
Vivendi Universal	220.68

Table 2.2: Daily turnover statistic (expressed in millions of euros) on different stocks.

2.2 Preprocessing

2.2.1 Abnormal periods

The following two periods must be discarded in the dataset since they are too 'chaotic' to deal with:

auction periods: from 9h00 to 9h02 and from 17h25 to 17h30 (French time) for the French stock market (or Euronext),

triple witching Fridays: third Friday of each quarterly month (March, June, September, December) when option and future contracts expire simultaneously.

2.2.2 Outlier detection

To detect outliers, a trimmed standard deviation bandwidth filter is used on each time window. First, we discard the $2 \times \alpha\%$ most extreme values, where α is a threshold, of the dataset. Then, the trimmed mean, $\mu^\star_\alpha$, and the trimmed standard deviation, $\sigma^\star_\alpha$, are computed with respect to each time window t over the trimmed dataset. Finally, we apply a bandwidth, bw, on the scaled data, $\frac{X_t^{(i)} - \mu^\star_c(X_t)}{\sigma^\star_c(X_t)}$, where $X_t^{(i)}$ represents the value on the time window t of the day i and we replace any outliers by $\mu^\star_c(X_t) \pm bw \cdot \sigma^\star_c(X_t)$.

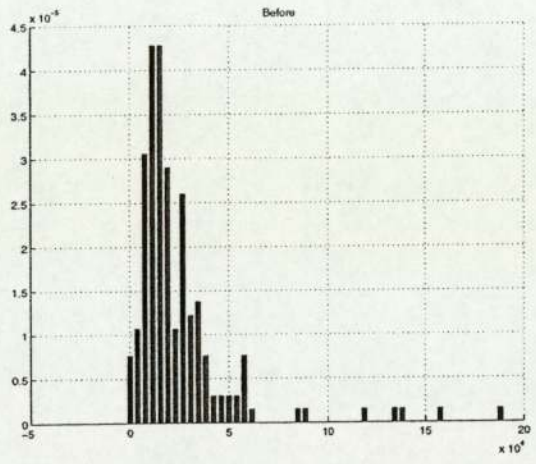
The advantage of using trimmed coefficients is to give a more robust estimate of the statistics, which is important due to the large kurtosis of the distribution.

Figure 2.2 depicts the distribution of data before and after having applied the filter. We clearly see the distribution properties that were highlighted in the raw volume statistics (see Table 2.1). Furthermore, in Figure 2.3, in the distribution of outliers, obviously the outliers are not time-dependent.

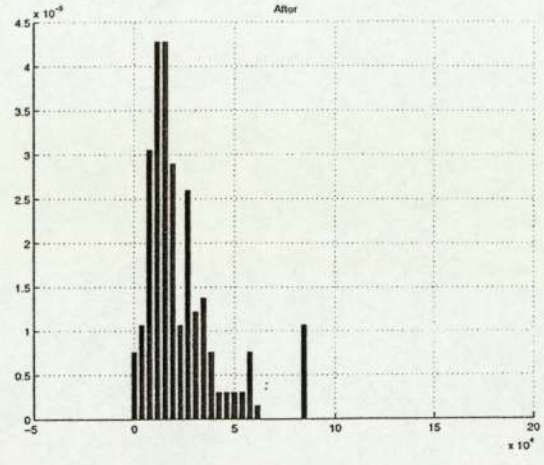
Table 2.3 shows the number of outliers modified with the use of this method. For the rest of the thesis, we will use 1% trimmed statistics since too many points are considered to be outliers if we increase α.

Stock	$\alpha = 1\%$	$\alpha = 5\%$
Alcatel	3.31%	6.26%
BNP-Paribas	3.18%	5.91%
France Télécom	3.43%	6.47%
Vivendi Universal	3.30%	6.46%

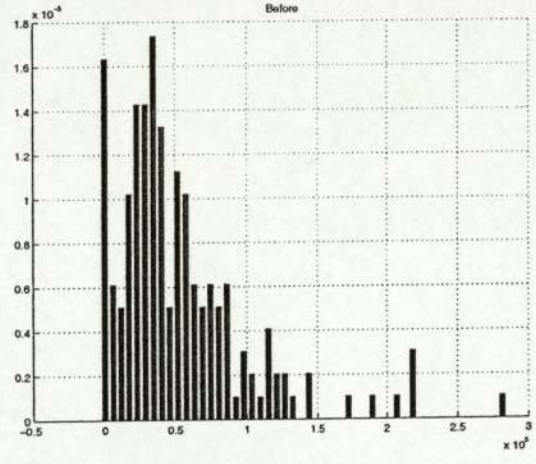
Table 2.3: Proportion of outliers with $bw = 3$ for 5 min tick-time volume.



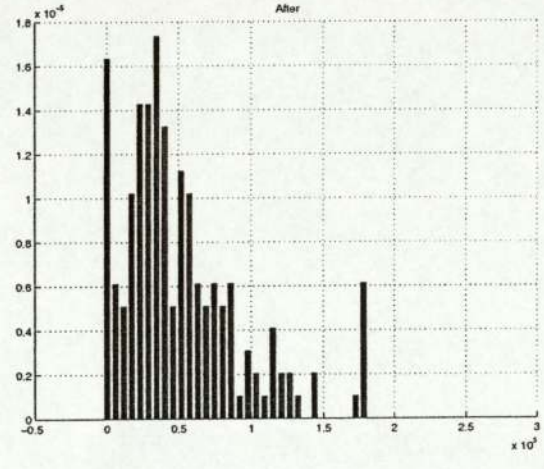
(a) 14:00 before



(b) 14:00 after



(c) 16:00 before



(d) 16:00 after

Figure 2.2: Distribution of 5 min raw volume at different hours before and after filtering ($bw = 3, \alpha = 1\%$) (BNP-Paribas on Mondays).

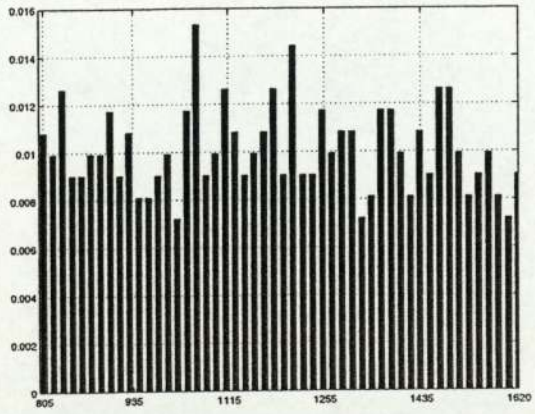


Figure 2.3: Distribution of the outliers by intraday time ($bw = 3, \alpha = 1\%$) (BNP-Paribas on Mondays).

2.2.3 Data transformation

We can also transform the time series (having removed the outliers) in order to obtain better properties. The two main possibilities are:

raw volumes: in other words no transformation,

relative volumes: given by $\tilde{V}_t := 100 \times \frac{V_t}{\sum_t V_t}$, with V_t, the raw volume.

The use of relative volume is driven by our main interest, which is the study of intraday patterns and their effect on the VWAP. Obviously, since the VWAP is a volume-weighted quantity, the formula (1.1) still holds with relative volume, $\tilde{V}_t$, instead of raw volume, V_t.

2.3 Preliminary statistical tests

Before moving on, two effects must be studied:

- day-of-week effect,
- stock effect.

In order to study these effects, we use the Analysis of Variance (ANOVA) statistical technique. ANOVA is a useful tool which helps the user to identify sources of variability from one or more potential sources, sometimes referred to as treatments or factors. This method is widely used in industry to help identify the sources of potential problems in the production processes and identify whether variation in measured output values is due to variability between various manufacturing processes, or within them. By varying the factors in a predetermined pattern and analysing the output, one can use statistical techniques to make an accurate assessment of the cause of variation in a manufacturing process.

The one-way ANOVA performs a comparison of the means of a number of replications of experiments performed where a single input factor is varied at different settings or levels. The object of this comparison is to determine the proportion of the variability of the data that is due to the different treatment levels or factors as opposed to variability due to random error.

It has to be remarked that we use the daily volumes as experiments in the ANOVA test, so we should not use a normalised measure such as the relative volume since it will be pointless to have all treatments equal to 1.

Stock	F-value
ACCP.PA	5.42
AGFP.PA	2.59
AIRP.PA	3.46
AVEP.PA	7.30
AXAF.PA	4.77
BNPP.PA	7.52
BOUY.PA	1.30
CAPP.PA	4.83
CARR.PA	8.54
CASP.PA	4.45
CGEP.PA	8.32
DANO.PA	1.33
EAUG.PA	5.18
EXHO.PA	6.66
FTE.PA	2.42
LAFP.PA	3.17
LAGA.PA	7.41
LVMH.PA	4.33
LYOE.PA	4.23
MICP.PA	5.26
OREP.PA	6.26
PERP.PA	3.20
PEUP.PA	9.04
PRTP.PA	4.09
RENA.PA	6.81
SASY.PA	5.54
SCHN.PA	4.22
SGEF.PA	2.10
SGOB.PA	3.34
SOGN.PA	5.27
STM.PA	1.05
TCFP.PA	3.97
TFFP.PA	7.39
TOTF.PA	8.44

Table 2.4: Weekday effect One-Way ANOVA test on CAC40 stocks, the experiments are daily volumes (outliers removed with $bw = 3, \alpha = 1\%$). Under a 5% p-value ($F_{4,\infty}^{5\%} = 2.37$) the null hypothesis is rejected except for the following stocks: AGFP.PA, BOUY.PA, DANO.PA, SGEF.PA and STM.PA.

2.3.1 Day-of-week effect

We performed a one-way ANOVA test with the null hypothesis that all daily means are equal. Table 2.4 shows the results of the test: the null hypothesis is rejected over the whole week except for five stocks with a p-value of 5%. The result seems correct since it looks common to have different trading activities for different days. Hence, we have to model each day of the week separately to model raw volume.

2.3.2 Stock effect

Concerning the stock effect, we are investigating here if the intraday volume is stock dependent. This question will be studied in the forthcoming Chapter 5. Loosely speaking, we will try to extract few

underlying forces that might drive a whole sector. In order to achieve this goal, component analysis techniques will be employed, and in particular correlation measures via principal component analysis (see Figure 5.2).

Chapter 3

Time-of-day Function

Contents

The purpose of this chapter is to estimate the seasonality of the intraday time series, called the time-of-day (TOD) function. Then, residuals will be computed from the TOD function and will be forecast in a stochastic modelling framework (see Chapter 4.2).

The TOD function is denoted by φ. The goal is to estimate $\hat{\varphi}$ from the data pairs $\{t_i, V_{t_i}\}_{i=1}^{N}$ with the following statistical model:

$$V_t = \varphi(t) + \epsilon_t, \tag{3.1}$$

where the random errors ϵ_t are zero mean iid. Thus, the problem can be viewed from the perspective of function approximation.

3.1 Splines

The first method for estimating the TOD function is to fit the widely used spline function class.

Definition 1 (Order-M spline). *An order-M spline with knots $\{\tau_j\}_{j=1}^{K}$ is a piecewise-polynomial of order M that has continuous derivatives up to order $M-2$. A cubic spline has $M=4$.*

It is claimed that cubic splines are the lowest-order spline for which the knot discontinuity is not visible to the human eye (Hastie, Tibshirani, and Friedman 2001). Splines of any higher order are seldom used, unless one is interested in smooth derivatives. However, since there is an explosion of the pointwise variance in a cubic spline model near the boundaries, a natural cubic spline is used since it adds additional constraints, namely that the function is linear beyond the boundary knots.

The number of knots and their placement is determined in an *ad hoc* way by placing one knot each hour and extra knots at the following active periods:

- opening of the domestic market,
- opening of the US market,
- ending of the domestic market.

As far as it concerns the data, since multiple realisations of the volume time series are available, we generate an artificial time series over a single day by averaging with respect to each time window over all days available; this is defined as the cross-sectional average volume $\bar{V}$:

$$\bar{V} := \frac{1}{d}\sum_{k=1}^{d} \boldsymbol{V}^{(k)}, \tag{3.2}$$

where d is the total number of days. Nevertheless, if the fitting is performed directly, the TOD estimate will obviously be very bumpy due to the nature of the financial data, and since the main interest here is a smooth seasonality estimate, *ad hoc* pre-smoothing is applied. This is achieved using a two-sided moving average at each knot, thus the following value is assigned to each knot τ_i:

$$\bar{V}_{\tau_i} := \frac{1}{2T+1}\sum_{k=-T}^{T} \bar{V}_{\tau_i+k}, \tag{3.3}$$

where T is the moving average window size. This parameter directly impacts the bias-variance trade-off of the model and consequently its generalisation[1]. This approach was conducted by (Azzouzi 2003b).

Since, overall, the cross-validation[2] (CV) curve (see Appendix C.1) is approximately unbiased as an estimate of the prediction error curve, $\mathrm{E}\left[(V_t - \hat{\varphi}_T(t))^2\right]$, 5-fold CV will be used for tuning each empirical parameter. Here the moving average window size T has an additional constraint which is to be a multiple of the time series tick frequency (5 minutes in general).

Table 3.1 presents the window sizes obtained. We can see that the time covered by a window, $2T+1$, is between 0.5 to 1 lag between two consecutive knots. This shows that the characteristic time for the data is half an hour.

[1] The ability to infer the correct structure from examples.

[2] A method of evaluating parameters by dividing the training set into several parts, and in turn using one part to test the procedure fitted to the remaining parts.

Figure 3.1 depicts the fitting of the estimated TOD function by this method for each day of the week. It is clear that the estimate follows the well-known U-shape with a bump at 15.00, which is the knot just after the opening of the US market. However, this estimate does not capture the peak, especially for Thursdays and Fridays, at 13.30, where it seems to reflect the characteristic of the end of the week just before the opening of the US market.

One major drawback of this method is that, actually, the smoothing has been performed twice, once by moving average technique and then by cubic spline interpolation. Also, the moving average window can only be a multiple of the tick-time frequency. This is a limitation since the search of the optimal smoothing parameter is reduced to few discrete values and not a whole interval.

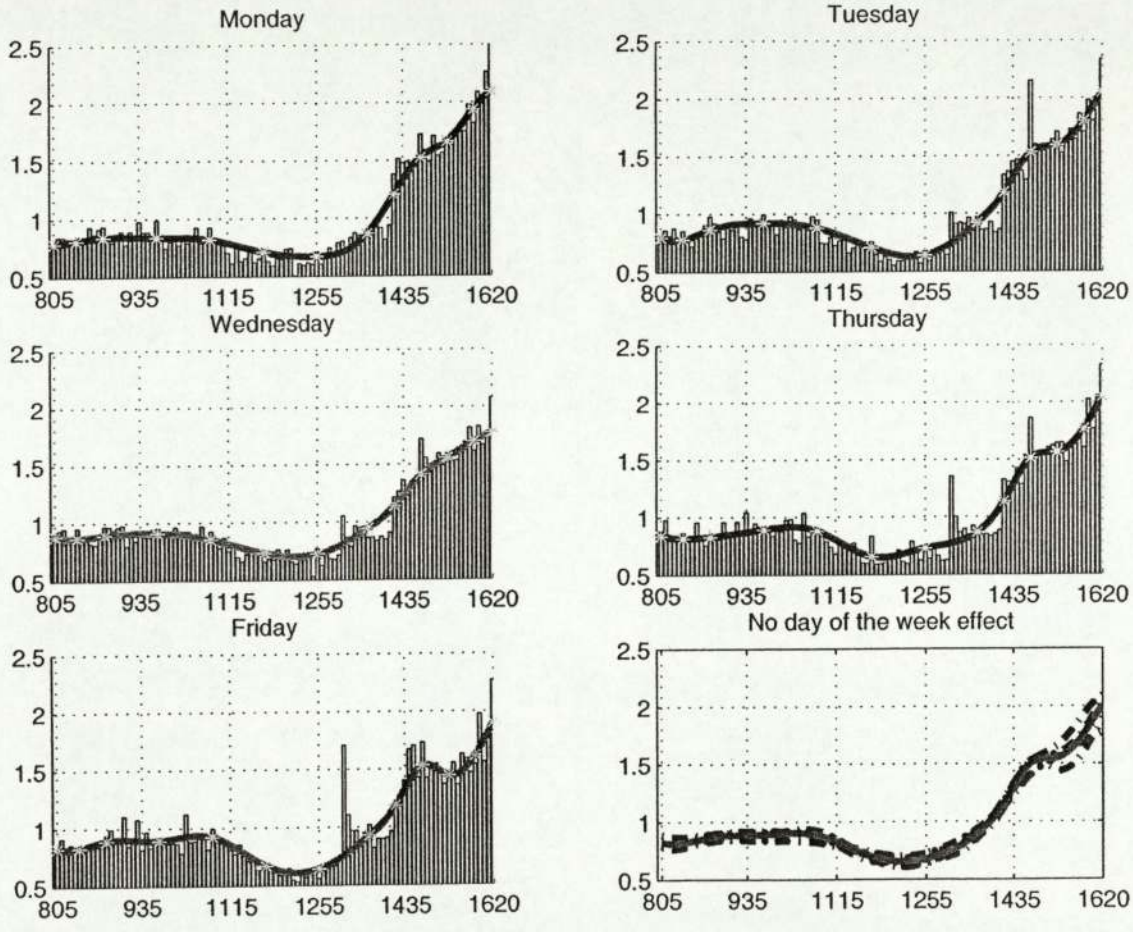


Figure 3.1: BNP-Paribas time of day estimate using moving average and cubic splines. The histograms represent the whole relative volume dataset.The stars correspond to the knots.

3.2 Smoothing splines

Seeking for a more general approach rather than the *ad hoc* method presented in section 3.1, the smoothing spline model arises. This spline based method avoids the knot selection problem completely by using a maximal set of knots, where all t_i's are knots. The complexity of the fit is controlled by

regularisation[3].

Definition 2. *The smoothing (cubic) spline function is defined as the solution of the following variational problem: among all functions, f, in the Hilbert space, W_2, of functions with continuous first derivatives and squared integrable second derivatives, find one that minimises the penalised residual sum of squares,*

$$\sum_{i=1}^{N}(y_i - f(x_i))^2 + \lambda \int f''(t)^2 \, \mathrm{d}t, \tag{3.4}$$

where $\lambda > 0$ is a fixed smoothing parameter.

The first term measures closeness of fit to the data, while the second term penalises curvature in the function. λ establishes a tradeoff between the two. Two special cases are:

$\lambda = 0$: f can be any function that interpolates the data,

$\lambda = \infty$: the simple least square fit, since no second derivative can be tolerated.

The smoothing parameter λ plays the same role as the moving average window of size T in section 3.1, and it is also typically chosen by minimising a CV criterion such as the 5-fold one. The numerical minimisation procedure was achieved using an iterated simple dichotomy (on the interval $[0, 1]$). This naive dissection technique should be improved using the golden section method.

Computational considerations for splines are given in Appendix A.

Table 3.1 presents the regularisation constants we obtain. Obviously there are completely different behaviours depending on the day of the week. Monday is smooth with a huge corresponding λ ($\approx 10^3$) and the rest of the week is generally very rough ($\lambda \approx 10^{-2}$). This shows that, mostly, the volume moves sharply against time. Simulation results on three other liquid stocks results can be found in Appendix D, in general we found a very small value of λ for all weekdays. Only BNP-Paribas exhibits a 'smooth Monday'.

Day	T	λ
All	15	6.7×10^{-2}
Monday	25	1.8×10^{3}
Tuesday	15	2.4×10^{-1}
Wednesday	30	22.7
Thursday	15	7.9×10^{-2}
Friday	15	3.5×10^{-2}
Witch	30	1.6×10^{-1}

Table 3.1: Regularisation constants obtained by 5-fold cross-validation (BNP-Paribas relative volume).

Figure 3.2 depicts the fitting of the TOD function by this method for each day of the week. The curves are mainly quite rough and, compared to the fitting by the 'moving average' method, this fit seems to capture the dynamics of the volume better. For example, each peak in the data is well captured.

[3] A class of methods of avoiding over-fitting to the training set by penalising the fit by a measure of 'smoothness' of the fitted function.

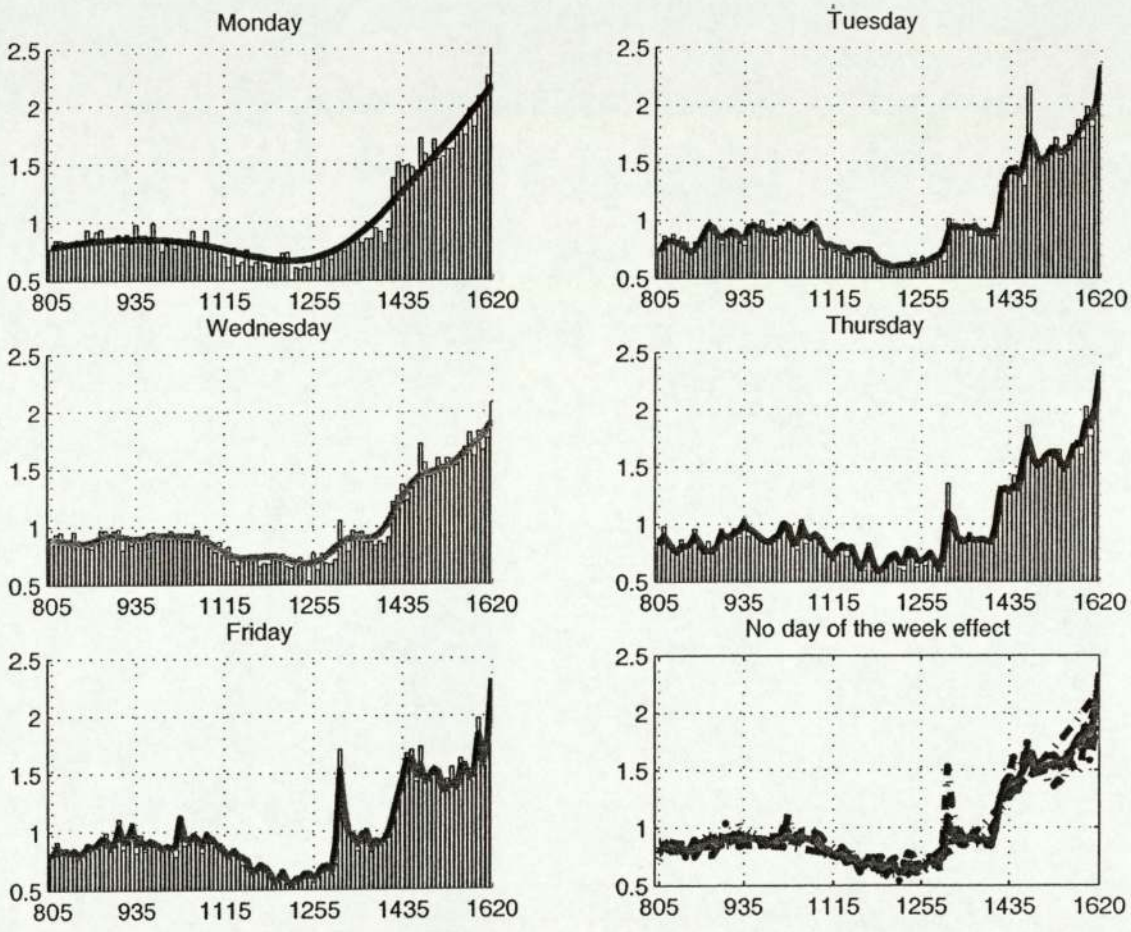


Figure 3.2: BNP-Paribas time of day estimate using cubic smoothing splines. The histograms represent the whole relative volume dataset.

By comparing the normalised Root Mean Squared (RMS) error, the smoothing spline performs slightly better than the moving average cubic spline (Table 3.2). Indeed, it is clear that spline-based methods outperform a naive mean prediction[4].

Thus, we can say that after all, the moving average cubic spline is a good method, but as a benchmark and a more principled way to handle the problem the smoothing spline is a better approach.

3.3 Bayesian regularisation

Knowing that cross-validation (CV) method is noisy, it is worth investigating the use of a Bayesian method which allows us to optimise regularisation constants on-line, where the estimates of regularisation constants are adjusted after each partial optimisation of the model parameters.

A general class of regularisation problem has the form

$$\arg\min_{f\in\mathcal{H}} \left\{ \sum_{i=1}^{N} L(y_i, f(x_i)) + \lambda\Omega[f] \right\}, \tag{3.5}$$

where L is a loss function, Ω is a penalty functional, and $\mathcal{H}$ is a space of function on which $\Omega[f]$ is defined.

This estimation method can be interpreted as a Bayesian method by identifying the prior for the function f as

$$p(f|\alpha) \propto \exp\{-\alpha\Omega[f]\} \tag{3.6}$$

and the conditional density of the data measurement $\mathcal{D}$, assuming independent (canonical) noise values $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$ as

$$p(\mathcal{D}|f_{\boldsymbol{w}}, \beta) \propto \exp\left\{ -\beta \sum_{i=1}^{N} L(y_i, f_{\boldsymbol{w}}(x_i)) \right\}, \tag{3.7}$$

where $\beta \equiv 1/\sigma^2$. Viewed as a function of the parameters, $\boldsymbol{w}$, and not of the observations, $\mathcal{D}$, it follows $\mathcal{L}(\boldsymbol{w}) \equiv p(\mathcal{D}|f_{\boldsymbol{w}}, \beta)$ which corresponds to the likelihood function.

Given this interpretation, the functional we wish to minimise in (3.5) is equal to the negative logarithm of the posterior probability, $p(f_{\boldsymbol{w}}|\mathcal{D}, \alpha, \beta)$, with an additive constant, and the estimation procedure can be interpreted as a Bayesian maximum a posteriori (MAP) estimate.

This Bayesian optimisation of model control parameters has four important advantages (MacKay 2003):

1. no test set or validation set is involved, so all available training data can be devoted to both model fitting and model comparison;

2. regularisation constants can be optimised simultaneously with the optimisation of ordinary model parameters;

3. the Bayesian objective function is not noisy, in contrast to a CV measure;

[4]For a mean prediction, the normalised RMS error is obviously 1.

4. the gradient of the evidence with respect to the control parameters can be evaluated, making it possible to simultaneously optimise a large number of control parameters.

This perspective also gives us an automatic method for inferring the hyperparameters , α and β. Assuming we have only a weak prior knowledge about the noise level and the smoothness of the interpolant, the evidence framework (see Section 4.2.3) optimises the constants α and β by finding the maximum of the evidence for α and β, $p(\mathcal{D}|\alpha, \beta)$.

3.4 Gaussian processes

Gaussian processes (GP) are based on the 'prior' assumption that adjacent observations should convey information about each other, that is to say the idea of GP modelling is to place a prior $p(f)$ directly on the space of functions, without parameterising f.

Just as a Gaussian distribution is fully specified by its mean and a covariance matrix, a GP is specified by a mean and a covariance function.

Definition 3 (Gaussian processes). *Denote by $t(x)$ a stochastic process parameterised by $x \in \chi$ (χ is an arbitrary index set). Then $t(x)$ is a Gaussian process if for any $m \in \mathbb{N}$ and $\{x_1, \ldots, x_m\} \subset \chi$, the random variables $(t(x_1), \ldots, t(x_m))$ are normally distributed.*

3.4.1 RKHS regularisation

Here, the duality between reproducing kernel Hilbert (RKHS) spaces and GPs is explained. For more details see (Hastie, Tibshirani, and Friedman 2001).

Considering a penalty functional, Ω, there exists a (positive definite) kernel K such that

$$\Omega[f] = \|f\|^2_{\mathcal{H}_K}, \tag{3.8}$$

where $\mathcal{H}_K$ is the RKHS with reproducing kernel K. Then, by Mercer's theorem (see appendix B), K has an eigen-expansion

$$K(x, x') = \sum_{\nu \geq 1} \gamma_\nu \phi_\nu(x) \phi_\nu(x'), \tag{3.9}$$

with $\gamma_\nu \geq 0$ and $\sum_{\nu \geq 1} \gamma_\nu^2 < \infty$. Thus, by definition the norm induced by K is

$$\|f\|^2_{\mathcal{H}_K} \equiv \sum_{\nu \geq 1} \frac{f_\nu^2}{\gamma_\nu} < \infty, \tag{3.10}$$

with $f_\nu = \langle f, \phi_\nu \rangle$ the Fourier coefficients of f.

Hence, (3.5) can be rewritten:

$$\arg \min_{f \in \mathcal{H}_K} \left[\sum_{i=1}^{N} L(y_i, f(x_i)) + \lambda \|f\|^2_{\mathcal{H}_K} \right]. \tag{3.11}$$

It can be shown that the solution of this minimisation has the form

$$f(x) = \sum_{i=1}^{N} \alpha_i K(x, x_i), \tag{3.12}$$

and hence, using the reproducing property of $\mathcal{H}_K$,

$$\Omega[f] = \sum_{i=1}^{N}\sum_{j=1}^{N} K(x_i, x_j)\alpha_i\alpha_j. \tag{3.13}$$

Finally, it follows that the regularisation prior is interpreted as a realisation of a zero-mean stationary GP with prior covariance $\boldsymbol{K}$,

$$p(f|\lambda) \propto \exp\{-\lambda\alpha^T \boldsymbol{K}\alpha\} \tag{3.14}$$

In particular, spline priors are GPs. To have a more unified theoretical way to conduct modelling, for the remainder of the thesis, kernel methods will be used, especially GPs.

3.4.2 Curve-fitting with Gaussian processes

Since our model is a Gaussian process, the conditional distribution of a prediction, t_{N+1}, at a new input given the training data, $\boldsymbol{t}_N$, $p(t_{N+1}|\boldsymbol{t}_N)$, is also Gaussian and completely specified by its mean and variance. We distinguish between different sizes of covariance matrix $\boldsymbol{K}$ with a subscript, such that $\boldsymbol{K}_{N+1}$ is the $(N+1)\times(N+1)$ covariance matrix for the vector $\boldsymbol{t}_{N+1} \equiv (t_1, \ldots, t_{N+1})^\top$ (following the notations in (MacKay 2003)). Then the matrix $\boldsymbol{K}_{N+1}$ can be partitioned as follows:

$$\boldsymbol{K}_{N+1} \equiv \begin{bmatrix} \boldsymbol{K}_N & \boldsymbol{k} \\ \boldsymbol{k}^\top & \kappa \end{bmatrix}. \tag{3.15}$$

The predictive mean and covariance at the new point are given by

$$\hat{t}_{N+1} = \boldsymbol{k}^\top \boldsymbol{K}_N^{-1}\boldsymbol{t}_N, \tag{3.16}$$

$$\sigma^2_{\hat{t}_{N+1}} = \kappa - \boldsymbol{k}^\top \boldsymbol{K}_N^{-1}\boldsymbol{k}. \tag{3.17}$$

GP fitting in Figure 3.3 captures the bumps of the data more smoothly, especially on Thursdays and Fridays.

Considering the RMS error in Table 3.2, the GP fit gives similar results as the smoothing spline fitting, which can be explained by the equivalence of the two models in a Bayesian framework as shown before.

Also, the GP fit presents small error bars, which means that the prediction of the GP is confident near the data. To remind the reader, the data is the cross-sectional average volume, that is to say just one artificial day.

3.4.3 Kernel design

The predictions produced by Gaussian processes depend entirely on the covariance matrix, $\boldsymbol{K}$. We implicitly assumed that the covariance function was known. We now discuss the types of covariance functions one might choose to define $\boldsymbol{K}$, and how we can automate the selection of the covariance function parameters in response to the data.

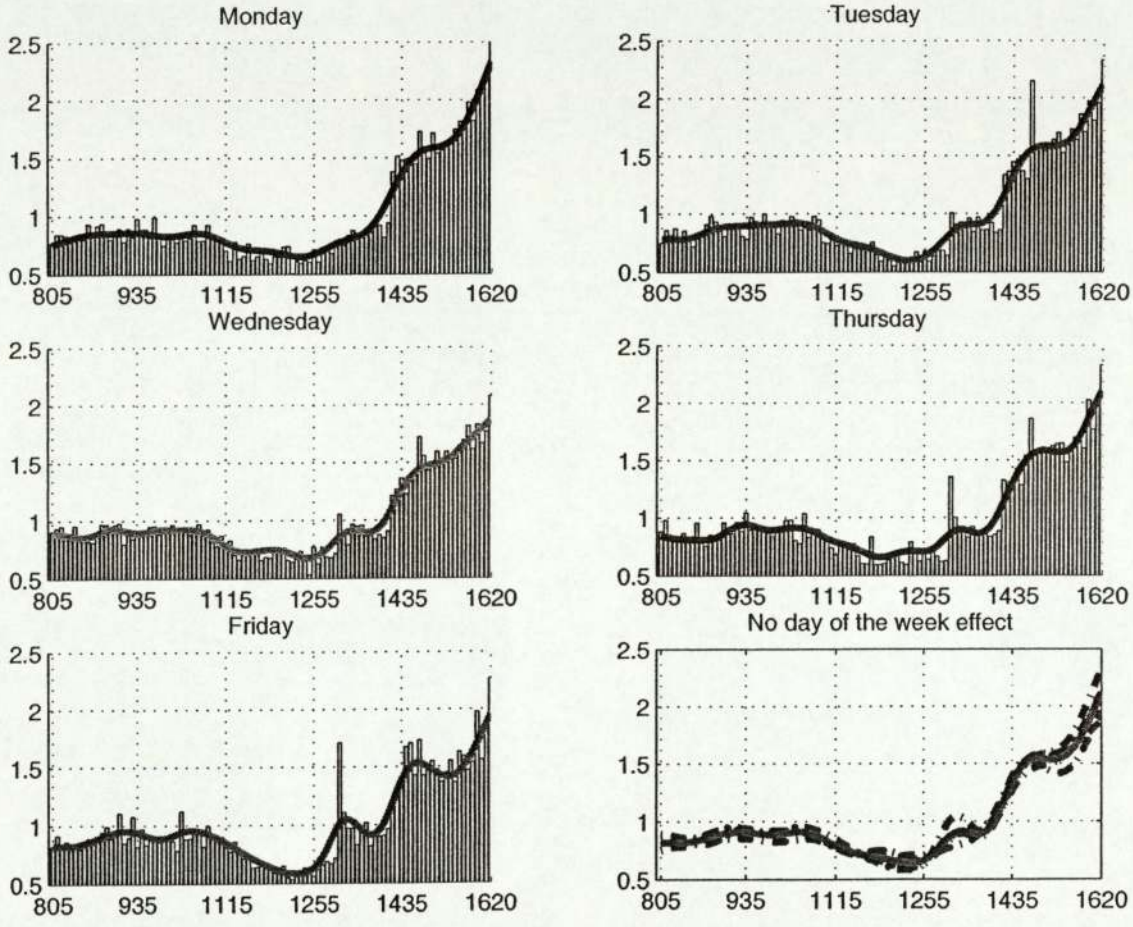


Figure 3.3: BNP-Paribas time of day estimate using Gaussian processes. The histograms represent the whole relative volume dataset.

Choice of the covariance function

The rational quadratic covariance function is given by

$$C(\boldsymbol{x}, \boldsymbol{x}') = \left(1 + (\boldsymbol{x} - \boldsymbol{x}')^\top \boldsymbol{W} (\boldsymbol{x} - \boldsymbol{x}')\right)^{-\nu}, \tag{3.18}$$

where $\boldsymbol{W}$ is positive definite. Typically, $\boldsymbol{W}$ is a diagonal matrix with length scale parameter w_j for each dimension.

The prior knowledge, a smooth TOD function estimate, is encoded in this covariance function. The motivation of this choice is that this function has an important feature, which is that the smoothness of the process can be regulated directly via ν, by controlling the rate of decay of the covariance function. Larger values of ν give a faster decay and hence a rougher process.

Furthermore, in practice a variance parameter, $v_0 > 0$, for vertical scaling and an offset parameter, $v_b > 0$, which is the uncertainty of a bias, are used. Therefore instead of C one uses $v_0 C + v_b$.

Learning the kernel

Let us assume that a form of covariance function has been chosen, but that it depends on undetermined hyperparameters $\boldsymbol{\theta}$. We would like to 'learn' these hyperparameters from the data. It is, once again, a complexity control problem, one that is solved by the Bayesian Occam's razor[5] (MacKay 2003).

Ideally we would like to define a prior distribution on the hyperparameters and integrate over them in order to make our predictions. But this integral is usually intractable. We can either approximate the integral using the most probable values of hyperparameters or perform the integration over $\boldsymbol{\theta}$ numerically using Monte-Carlo methods. In both cases, to implement these approaches efficiently, the gradient of the log-likelihood should be evaluated:

$$\frac{\partial \ell}{\partial \theta_i} = -\frac{1}{2}\mathrm{Tr}\left(\boldsymbol{K}_N^{-1}\frac{\partial \boldsymbol{K}_N}{\partial \theta_i}\right) + \frac{1}{2}\boldsymbol{t}_N^\top \boldsymbol{K}_N^{-1}\frac{\partial \boldsymbol{K}_N}{\partial \theta_i}\boldsymbol{K}_N^{-1}\boldsymbol{t}_N. \tag{3.19}$$

Approximate inference and learning One of the most important issues concerning GPs is the computational cost. In fact, prediction and evaluation of the gradient of the log-likelihood requires the evaluation of $\boldsymbol{K}^{-1}$, where $\boldsymbol{K} := [C(x_i, x_j)]_{i,j=1}^N$ is the covariance matrix of the training data. Any exact inversion method (such as Cholesky decomposition, LU decomposition or Gauss-Jordan elimination) has an associated cost of order $\mathcal{O}(N^3)$ and thus the cost of these direct methods becomes prohibitive when the number of data points, N, is greater than around 1000.

Sparse approximations to GP inference called sparse on-line Gaussian processes were developed in (Csató and Opper 2002). While the original application was online learning, they can be understood as a 'sparsification' of a special case of the expected propagation (EP) algorithm for Gaussian fields. The sparse inference approximations reduce this time scaling to $\mathcal{O}(Nd^2)$ with adjustable $d \ll N$. Another sparse scheme is the relevance vector machine (RVM) (Tipping 2001) which will be seen later in Section 3.5.

[5] If several explanations are compatible with a set of observation, Occam's razor advises us to buy the simplest.

3.5 Sparse Bayesian learning

Here, the model to be focused on is the relevance vector machine (RVM) (Tipping 2001; Bishop and Tipping 2003), which is a Bayesian framework for regression with sparsity properties. We adopt a fully probabilistic framework and introduce a prior over the model weights governed by a set of hyperparameters, one associated with each weight, whose most probable values are iteratively estimated from the data. Sparsity is achieved because the posterior distributions of many of the weights are sharply (indeed infinitely) peaked around zero. We term those training vectors associated with the remaining non-zero weights 'relevance' vectors, in deference to the principle of automatic relevance determination which motivates this approach (MacKay 1995).

The most compelling feature of the RVM for the TOD function estimation problem is to have a view of the sparsity/relevance of the time-of-day against the traded volume.

3.5.1 Model specification

Given a data set of input-target pairs $\{x_i, t_i\}_{i=1}^N$, it is assumed that the targets are samples from a model with an additive noise process given by a zero-mean Gaussian with variance σ^2, leading to

$$p(\boldsymbol{t}|\boldsymbol{w}, \sigma^2) = (2\pi\sigma^2)^{-N/2} \exp\left\{-\frac{\|\boldsymbol{t} - \boldsymbol{\Phi}\boldsymbol{w}\|^2}{2\sigma^2}\right\}, \tag{3.20}$$

with $y(x) = \sum_{i=1}^M w_i \phi_i(x)$.

We encode a preference for smoother functions by using a Gaussian distribution over $\boldsymbol{w}$ with a separate hyperparameter (adjustable variance) for each parameter in the model

$$p(\boldsymbol{w}|\boldsymbol{\alpha}) = \prod_{i=1}^M \mathcal{N}(w_i|0, \alpha_i^{-1}). \tag{3.21}$$

To complete the specification of this hierarchical prior, we must define hyperpriors over the scaling parameters $\boldsymbol{\alpha}$ and σ^2. Suitable parameters for these are given by Gamma distributions:

$$p(\boldsymbol{\alpha}) = \prod_{i=1}^M \mathcal{G}(\alpha_i|a, b), \tag{3.22}$$

$$p(\beta) = \mathcal{G}(\beta|a, b), \tag{3.23}$$

with $\beta \equiv 1/\sigma^2$ and where $\mathcal{G}(x|a, b) = \frac{1}{\Gamma(a)b^a} x^{a-1} e^{-x/b}$.

3.5.2 Inference

Given $\boldsymbol{\alpha}$, the posterior parameter distribution is Gaussian and given via Bayes' rule as $p(\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\alpha}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with

$$\boldsymbol{\Sigma} = (\boldsymbol{A} + \sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{\Phi})^{-1}, \tag{3.24}$$

$$\boldsymbol{\mu} = \sigma^{-2}\boldsymbol{\Sigma}\boldsymbol{\Phi}^\top\boldsymbol{t}, \tag{3.25}$$

and $\boldsymbol{A}$ defined as $\text{diag}(\alpha_1, \ldots, \alpha_M)$. Sparse Bayesian learning can then be reformulated as a type-II maximum likelihood procedure, in that the objective is to maximise the marginal likelihood, or equivalently, its logarithm with respect to the hyperparameters $\boldsymbol{\alpha}$:

$$\log p(\boldsymbol{t}|\boldsymbol{\alpha}, \sigma^2) = -\frac{1}{2}\left(N \log 2\pi + \log |\boldsymbol{C}| + \boldsymbol{t}^\top \boldsymbol{C}^{-1} \boldsymbol{t}\right), \tag{3.26}$$

with $\boldsymbol{C} = \sigma^2 \boldsymbol{I} + \boldsymbol{\Phi} \boldsymbol{A}^{-1} \boldsymbol{\Phi}^\top$ (Tipping 2001).

3.5.3 Relevance vector regression in action

To apply the RVM model, the specification of the design matrix, $\boldsymbol{\Phi}$, is required. One way to use the preceding results is to employ the kernel K 'learnt' in Section 3.4.3 as basis functions ϕ_i such that $\phi_i := K(\cdot, x_i)$.

Figure 3.4 depicts the fit of the RVM. This fit is not much better than the GP considering the RMS error (Table 3.2), and the improvement is insignificant.

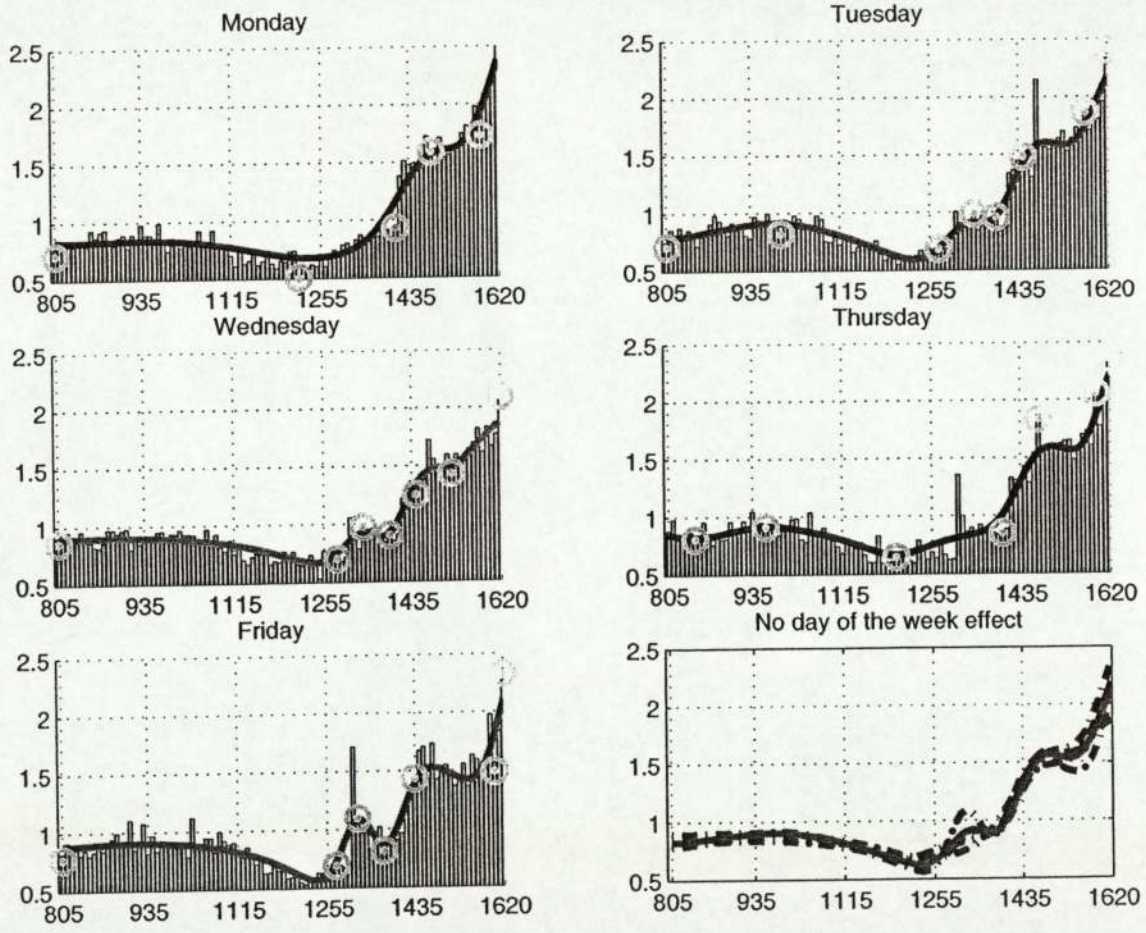


Figure 3.4: BNP-Paribas time of day estimate using the relevance vector machine. The histograms represent the whole relative volume dataset.The circled points correspond to the relevance vectors.

Concerning sparsity, the plot shows the relevant time points occurring over the day: one or two points in the morning (8:05 and sometimes 10:15) and around six points every half an hour from 13:00.

It has also to be pointed that for some certain stocks (Tables D.4 and D.8), almost all Fridays' time points are found out as relevant. This behaviour cannot be simply explained, although Fridays have a much more complex behaviour than the other weekdays.

This result shows the importance of the opening of the US market at 14:30 to the end of the day.

Day	MA	Smooth	GP	RVM	RBF
Monday	0.8275	0.8282	0.8237	0.8247	0.8280
Tuesday	0.8553	0.8495	0.8537	0.8531	0.8629
Wednesday	0.8718	0.8695	0.8697	0.8690	0.8718
Thursday	0.8608	0.8532	0.8565	0.8580	0.8576
Friday	0.8862	0.8709	0.8764	0.8761	1.2892

Table 3.2: RMSE on the time-of-day function on the testing set (BNP-Paribas relative volume).

3.6 Training with multiple days

The motivation here is to investigate whether the use of the cross-sectional average volume instead of the multiple realisations of the daily volume time series leads to a loss of information.

Due to the problem of scaling size in dealing with multiple days, kernel methods are intractable since the computational cost will be multiplied by the cube[6] of the number of days in the training set. An alternative is to recall, that the inputs of the training set are always the same, that is to say the time points over the day at a 5-minute tick frequency. Therefore, since the covariance matrix in GPs only involves input training data, with some block design and linear algebra, we should be able to devise a less costly model.

An alternative and simpler approach is to use radial basis functions (RBF). Training is very fast with this model, using an error function quadratic in the weights. More details about the use of the RBF network will be given in Section 4.2.3. It has to be pointed out that RBF is also a solution of the regularisation problem (3.5) as for example the smoothing spline. (Girosi, Jones, and Poggio 1995) describe a quite general penalty of the form[7]

$$\Omega[f] = \int_{\mathbb{R}^d} \frac{|\tilde{f}(s)|^2}{\tilde{G}(s)} \, \mathrm{d}s, \tag{3.27}$$

for some positive symmetric function $\tilde{G}$ that falls off to zero as $\|s\| \to \infty$. Here the tilde denotes Fourier transformation, and it turns out that $\tilde{G}$ is the Fourier transform of the basis function $g(\|\boldsymbol{x} - \boldsymbol{\mu}_i\|)$.

The close link with GP is that using a covariance function C such that $C(\boldsymbol{x}, \boldsymbol{x}') = g(d(\boldsymbol{x}, \boldsymbol{x}'))$, where d is a metric on the input space, leads to a mapping of the following form

$$y(\boldsymbol{x}) = \sum_{j=0}^{M} w_j \phi_j(\boldsymbol{x}), \tag{3.28}$$

with ϕ defined by the empirical kernel map with respect to the training patterns.

[6] This is due to a matrix inversion.

[7] The idea is that $1/\tilde{G}$ increases the penalty for high-frequency components of f.

Then, we choose $g(r) = r^2 \log(r)$, the thin plate spline function, an unbounded and non positive definite basis function that has a good theoretical motivation from the domain of functional interpolation (Lowe 1995).

In addition, Bayesian techniques are used to compute the predictive distribution

$$p(y|x^\star, \mathcal{D}) = \int p(y|x^\star, \boldsymbol{w}) p(\boldsymbol{w}|\mathcal{D}) \, \mathrm{d}\boldsymbol{w}, \tag{3.29}$$

of a new input, $x^\star$, in order to take into account uncertainty about parameters estimated from the data. Thus error bars of our prediction can be provided using the variance of the distribution. This approach is carried out using an approximation of the integral via the evidence procedure (see Section 4.2.3).

Figure 3.5 shows that the prediction using all data is, sensibly, the same as considering a single day with cross-sectional average volume. However, the error bars (one standard deviation) are enormous, which means that there is significant uncertainty in the fitted parameters. Furthermore, this figure depicts error bars being negative, which reminds us that the Gaussian noise model is not the right one, since volume is a positive quantity.

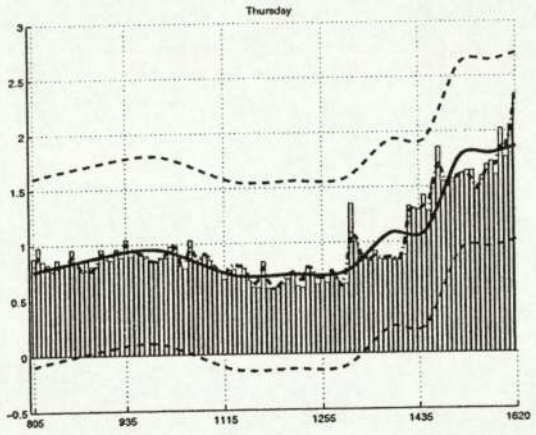


Figure 3.5: BNP-Paribas time of day estimate on Thursdays using an RBF network (9 hidden units). The histograms represent the whole relative volume dataset. The dashed lines represent one standard deviation error bars. The dash-dotted line represents the corresponding smoothing spline fit.

3.7 Issues

We have seen that the smoothing spline estimate was a good estimate of the TOD function and we have introduced it in a Bayesian framework which gives us more insight in the modelling, although it does not improve the fitting significantly. Recalling that the posterior, from Bayes' theorem, describes our knowledge of the parameters once we have combined the observations with our prior belief of the parameters. Therefore, we could explain the fact that the Bayesian perspective did not achieve significantly better fit by saying that the posterior 'update' was not successful and 'remained' as the

likelihood. In other words, the maximum a posteriori (MAP) estimator remained as a maximum likelihood estimate.

Since we obtained with huge error bars concerning multiple days prediction, one important issue to work further on is robust regression with a non-Gaussian noise model, for example a Student-t noise (Tipping and Lawrence 2003). This kind of model is expected to have a better generalisation, and of course, smaller error bars.

Chapter 4

Stochastic Modelling

Contents

The purpose of this chapter is to convey the stochastic modelling of the residuals computed by removing the time-of-day seasonality of the time series. First, the residual distribution is studied, then prediction is tackled.

4.1 Residual analysis

Here, we investigate the residuals, r_t, of the estimated time-of-day function, $\hat{\varphi}$, either multiplicative, $r_t^* = \frac{V_t}{\hat{\varphi}_t}$ (ratios), or additive, $r_t^+ = V_t - \hat{\varphi}_t$ (differences).

More precisely, we will focus on, the empirical probability density function of the residuals and the autocorrelation structure of the residual time series.

Since our aim is to predict the residuals, this is a preliminary study, for example in order to investigate which noise model to employ.

4.1.1 Properties

First, by construction, the distribution presents a high peak at zero and one for differences and ratios respectively. Furthermore, the distributions exhibit fat tails and fast decay in both cases. All of these features are encountered regardless of which day of week is considered.

Table 4.1 presents summary statistics on the residuals. The high kurtosis and the skewness confirm the features already seen and discussed in Section 2.1.1.

Thus, the distributions are clearly non-normal and a possible way to observe departures from normality is to use a quantile-quantile (QQ) plot[1]. A QQ plot is a scatter plot of the empirical quantiles against theoretical quantiles. Obviously, here, the theoretical quantiles used are those from a normal distribution.

Figure 4.1 shows that the departure from the normal distribution is very significant. However, this behaviour is not present to the same degree for small residual values, that is to say, less than zero and one respectively for differences and ratios

In summary, the residuals have an asymmetric heavy-tailed distribution. An interpretation of these empirical results could be that the deterministic models we used for modelling the intraday volume were fairly good in general, except for rare events, which seem hard to capture in a deterministic way. The aim of the stochastic models we are going to build is to detect these kind of events in a more powerful manner.

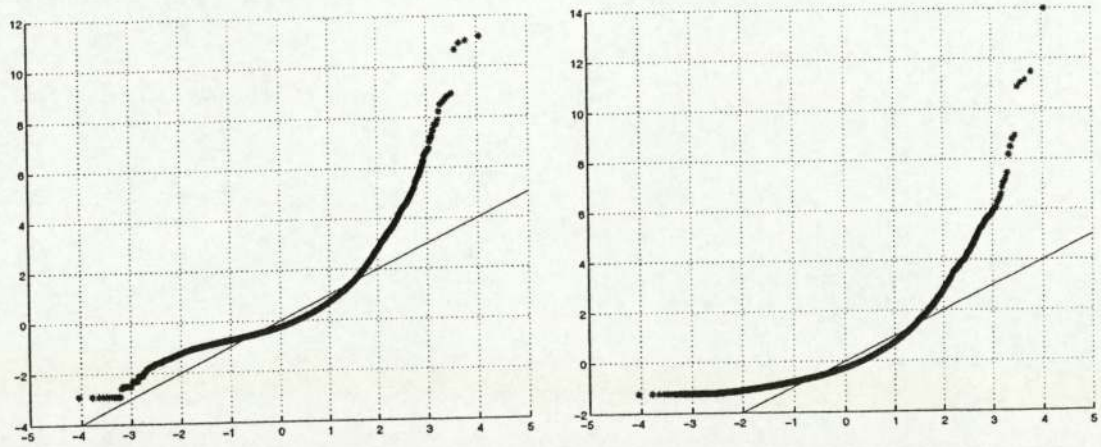
4.1.2 Density estimation

For practical reasons, we will use residuals computed from smoothing spline fitting on relative volumes, since residuals from raw and relative volume have the same patterns. Our aim is to infer the parameters of the densities used for fitting.

[1] An α-quantile of a probability is a value x such that $\Pr(X < x) = \alpha$.

day	μ	σ	min	max	λ_3	λ_4
			Alcatel			
Monday	0.000	0.772	-1.921	8.580	2.309	10.681
Tuesday	-0.000	0.784	-1.937	9.054	2.285	10.195
Wednesday	0.000	0.759	-1.820	10.968	2.386	12.196
Thursday	-0.000	0.786	-1.981	15.423	2.956	24.846
Friday	0.000	0.799	-1.937	12.891	2.803	17.776
			BNP-Paribas			
Monday	-0.000	0.813	-2.184	9.061	2.367	11.172
Tuesday	-0.000	0.811	-2.338	17.355	2.809	21.430
Wednesday	-0.000	0.787	-1.917	9.518	2.247	9.370
Thursday	0.000	0.804	-2.343	8.988	2.454	11.817
Friday	0.000	0.818	-2.316	13.407	2.863	19.575
			France Télécom			
Monday	-0.000	0.842	-2.300	12.134	3.131	22.889
Tuesday	-0.000	0.814	-2.128	12.479	2.945	20.018
Wednesday	0.000	0.821	-2.255	13.922	2.987	20.702
Thursday	0.000	0.855	-2.510	23.758	3.792	46.347
Friday	0.000	0.866	-2.116	16.325	3.614	31.383
			Vivendi Universal			
Monday	-0.000	0.811	-3.281	10.781	2.707	17.228
Tuesday	0.000	0.831	-2.490	11.921	2.874	16.415
Wednesday	-0.000	0.822	-2.259	13.651	3.106	21.769
Thursday	-0.000	0.818	-2.677	10.497	2.794	16.009
Friday	-0.000	0.839	-3.016	15.242	3.026	22.369

Table 4.1: Statistics on Alcatel, BNP-Paribas, France-Télécom, Vivendi Universal additive residuals.



(a) Additive residuals

(b) Multiplicative residuals

Figure 4.1: QQ plots of BNP-Paribas residuals on Thursdays. x-axis corresponds to theoretical quantiles, y-axis to empirical ones.

Multiplicative residuals

Using multiplicative residuals, it is natural to use the Gamma density:

$$\mathcal{G}(x|a,b) = \Gamma(a)^{-1} b^{-a} x^{a-1} e^{-bx}, \tag{4.1}$$

and the Weibull density:

$$\mathcal{W}(x|a,b) = abx^{a-1} e^{-ax^b}, \tag{4.2}$$

to fit the empirical distribution.

Inference on the Gamma distribution parameters is achieved using a moment-matching technique, and maximum likelihood estimation is used for the Weibull distribution parameters.

Figure 4.2(a) shows that the Gamma distribution misses the fit completely. The Weibull fit is better, although it is still very poor between residual values of 0 and 1.

Additive residuals

Additive residuals are not restricted to be positive. Hence it is sensible to use semi-parametric models such as a Gaussian mixture model (GMM):

$$p(x) = \sum_{i=1}^{n} P(i)\mathcal{N}(x|\mu_i, \sigma_i^2), \tag{4.3}$$

with the following constraints on the mixing coefficients $P(i)$: $\sum_{i=1}^{n} P(i) = 1$ and $0 \leq P(i) \leq 1$.

GMMs are universal approximators, in that they can model any density function arbitrarily closely provided that they contain enough components.

Gaussian components are fitted by maximum likelihood using a specialised method, known as the expectation-maximisation, or EM, algorithm (Bishop 1995).

Figure 4.2(b) depicts that GMM with two centres fits reasonably well and with three centres it 'sticks' to the data very well.

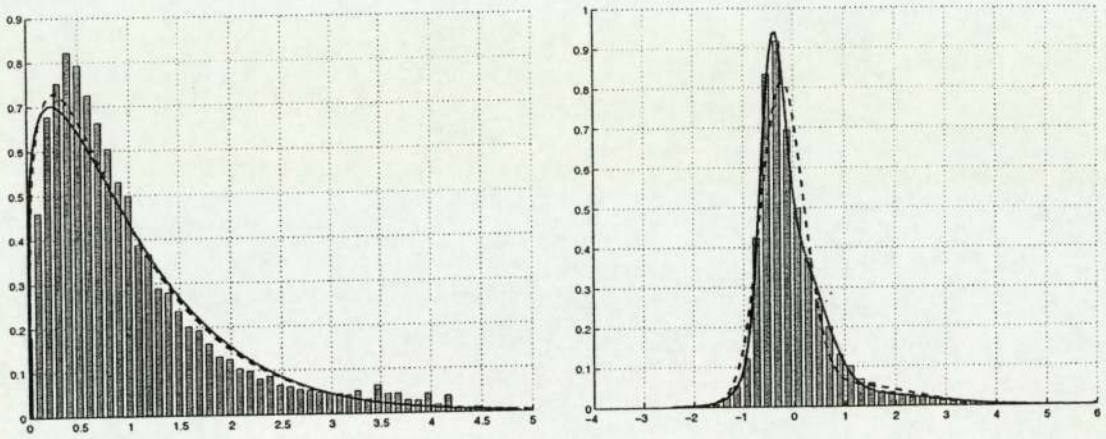
4.1.3 Autocorrelation structure

To investigate the autocorrelation of the residual time series, we evaluate the partial autocorrelation function (PACF) rather than the full autocorrelation function in order to have a more robust measure since the PACF removes the effect of shorter lag autocorrelation from the correlation estimate at longer lags.

Definition 4 (Partial autocorrelation function). *The partial autocorrelation function is defined as* $\pi_1 = \mathrm{Corr}(X_1, X_2)$ *and*

$$\pi_k = \mathrm{Corr}(R_{1|2,\ldots,k}, R_{k+1|2,\ldots,k}) \quad \text{for } k \geq 2, \tag{4.4}$$

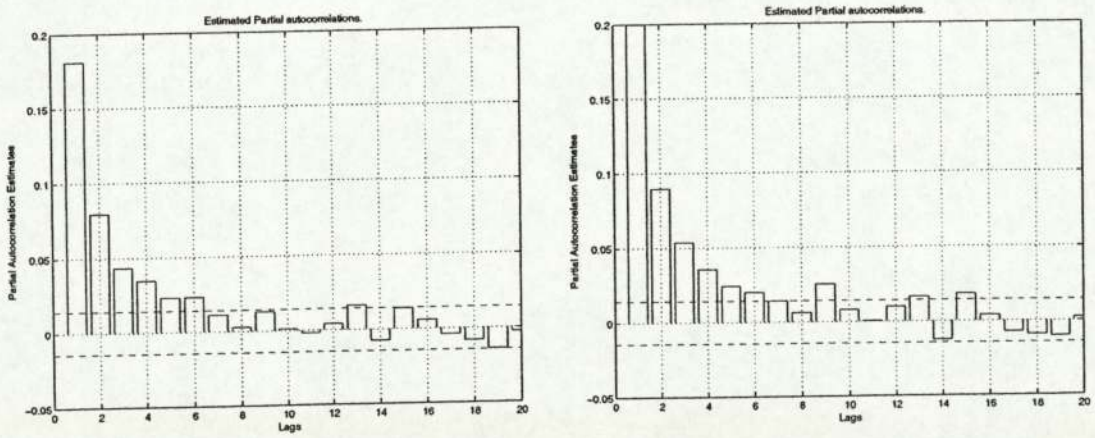
where $R_{j|2,\ldots,k}$ *is the residual from the linear regression of* X_j *on* $(X_2, \ldots, X_k)$.



(a) Gamma (dashed line) and Weibull (solid line). $\ell(\mathcal{G}) = 0.9910$, $\ell(\mathcal{W}) = 0.9914$.

(b) GMM with 2 centres (dashed line) and 3 centres (solid line) with spherical covariance. $\ell(\text{GMM}_2) = 0.96$, $\ell(\text{GMM}_3) = 0.93$.

Figure 4.2: Density fits on multiplicative and additive residuals on BNP-Paribas on Thursdays (raw volumes). ℓ denotes the negative log likelihood of the model per data point.



(a) Additive residuals.

(b) Multiplicative residuals.

Figure 4.3: Partial autocorrelation plot of Thursdays residuals of BNP-Paribas. The dashed lines give the pointwise acceptance region for testing the null hypothesis, $\pi_k = 0$, at the 5% significance level.

Figure 4.3 displays the PACF estimates. Tthe plot cuts around the fifth lag. Hence, this plot suggests the use of an autoregressive model as we shall see in the next chapter.

Also, the same PACF plot is obtained for all day of the week so it is legitimate to assume that the linear structure of the residuals is day-of-week independent.

4.2 Prediction

The goal of this section is to achieve the one-step ahead prediction of the residuals. This prediction task is important for VWAP trading since, usually, traders receive orders that must be filled over a trading horizon, hence knowing the volume dynamics is crucial.

4.2.1 Delay embedding technique

Definition 5 (Embedding). *An embedding of a manifold, M, is a smooth[2] diffeomorphism, $\Psi : M \mapsto \Psi(M) \subset U$, such that $\Psi(M)$ is a sub-manifold of U.*

Definition 6 (Delay map). *A delay map $\mathcal{F}_{F,s,\tau} : M \mapsto \mathbb{R}^m$ of dimension m is defined by:*

$$\mathcal{F}_{F,s,\tau}(\boldsymbol{x}_n) = [s(\boldsymbol{x}_n), s(F^{\tau}(\boldsymbol{x}_n)), \ldots, s(F^{m\tau}(\boldsymbol{x}_n))], \tag{4.5}$$

where $F : M \mapsto M$ is a flow on the manifold M, $s : M \mapsto \mathbb{R}$ is smooth measurement function and τ is a positive number called the delay.

(Takens 1981) proved that an embedding could be performed using a specific class of maps called delay coordinate maps:

Theorem 1 (Delay embedding). *It is a generic property that a delay map of dimension $2D+1$ is an embedding of a compact manifold with dimension D if the measurement function, $s : M \mapsto \mathbb{R}$, is C^2 and if either the dynamics or the measurement function is generic in the sense that it couples all degrees of freedom.*

The theorem can be visualised in the following commutative diagram:

$$\begin{array}{ccc}
\boldsymbol{x}_n \in M & \xrightarrow{F} & \boldsymbol{x}_{n+1} \in M \\
\downarrow s & & \downarrow s \\
s_n \in \mathbb{R} & & s_{n+1} \in \mathbb{R} \\
\downarrow e & & \downarrow e \\
\boldsymbol{s}_n \in \mathbb{R}^{2D+1} & \xrightarrow{G} & \boldsymbol{s}_{n+1} \in \mathbb{R}^{2D+1}
\end{array}$$

By applying this result to our case, we try to 'learn' the following mapping:

$$x_t = f(x_{t-1}, x_{t-2}, \ldots, x_{t-d}), \tag{4.6}$$

that is to say we forecast the next value with the knowledge of the d previous values, as shown in Figure 4.4.

[2]We use 'smooth' for at least C^2.

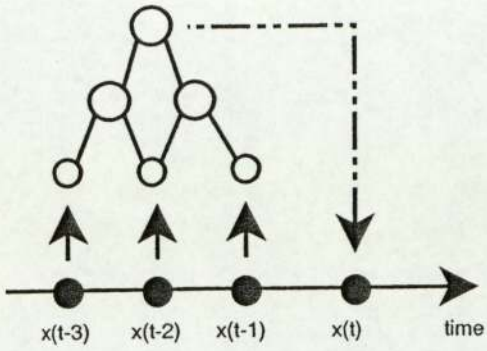


Figure 4.4: One-step ahead prediction with a neural network.

4.2.2 Benchmarks

Autoregressive model

Definition 7 (Autoregressive (AR) model). *An autoregressive model of order $p \geq 1$ AR(p) is defined as*

$$X_t = \sum_{i=1}^{p} b_i X_{t-i} + \epsilon_t, \tag{4.7}$$

where $\{\epsilon_t\}$ is a white noise process.

It is straightforward to see that the AR(p) model can be viewed as a single layer network with inputs as past values $\{x_{t-i}\}_{i=1}^{p}$ and weights as the parameters $\{b_i\}_{i=1}^{p}$. Single layer networks implement the well known statistical techniques of linear regression and generalised linear models (GLM). It is always useful to apply a GLM to a dataset to provide a benchmark for more sophisticated methods. Furthermore, because of their simplicity, they rarely overfit the training data, and they also have the advantage of being extremely fast to train (Nabney 2002).

Random walk

Another widely used benchmark in the financial industry is the following:

Definition 8 (Random walk model). *A random walk model defined by:*

$$X_t = X_{t-1} + \epsilon_t, \tag{4.8}$$

where $\{\epsilon_t\}$ is a white noise process.

We can interpret the assumption made by this model as 'the best forecast for tomorrow is today's value'.

4.2.3 Neural networks

Artificial neural networks (NN) are nonlinear, identical and highly connected elements capable of learning, in the sense of modification of model parameters and/or the model itself on the basis of

training examples. In other words, a NN is a nonlinear model whose parameters can be estimated from the data.

Hence a NN can be viewed as a machine that can 'learn' and perform 'pattern recognition'. Furthermore, a NN is a universal approximator in the sense that it can approximate to arbitrary accuracy any continuous function from a compact region of input space provided the number of hidden units is sufficiently large and provided the weights and biases are chosen appropriately. This means, in practice, that a NN can model any smooth function provided there is enough data to estimate the network parameters. Unfortunately, there is no guarantee that we quickly find such an ϵ-error solution. The optimisation problem is theoretically hard because of a potentially large number of local minima.

We will consider neural network modelling through multi-layer perceptrons and radial basis function networks both of which are two-layered feedforward NN consisting of:

d **input units:** the embedding dimension of the time series,

m **hidden units:** the complexity of the network,

and one output unit. The topology of the networks will be chosen in the usual method using a training and validation set, since a large amount of data is available.

Multi-layer perceptron

The multi-layer perceptron (MLP) is probably the most widely used architecture for practical applications of neural networks (Bishop 1995). It consists of two layers of adaptive weights with full connectivity between inputs and hidden units, and between hidden units and outputs.

The relationship between inputs, weights and output is defined through this mapping:

$$y(\boldsymbol{x}, \boldsymbol{w}) = \sum_{j=1}^{m} w_j^{(2)} \tanh\left(\sum_{i=1}^{d} w_{ji}^{(1)} x_i + b_j^{(1)}\right) + b^{(2)}, \tag{4.9}$$

with d and m defined as before. The network is trained using the back-propagation algorithm[3]. The optimisation routine is performed via the scaled conjugate gradient algorithm.

Radial basis function network

The radial basis function (RBF) network is the main practical alternative to the MLP for nonlinear modelling. Instead of units that compute a nonlinear function of the scalar product of the input vector and a weight vector, the activation of the hidden units in an RBF network is given by a nonlinear function of the distance between the input vector and a weight vector (Nabney 2002).

We shall write the RBF network mapping in the following form:

$$y(\boldsymbol{x}; \boldsymbol{w}) = \sum_{j=1}^{m} w_j \phi_j(\boldsymbol{x}) + w_0, \tag{4.10}$$

[3]Method used to calculate the gradient vector of a fitting criterion for a feed-forward neural network with respect to the weights.

where the ϕ_j are the basis functions, and the w_j the output layer weights.

One advantage of the RBF model is that it brings together different approaches such as function approximation, kernel methods, regularisation theory *etc.* Another attraction of RBF networks is that there is a two-stage training procedure which is considerably faster than the methods used to train MLPs. In the first stage, the parameters governing the basis functions (corresponding to the hidden units) are determined using relatively fast, unsupervised methods[4]. The second stage of training then involves the determination of the output-layer weights, which requires the solution of a linear problem, and is therefore fast (Bishop 1995).

Bayesian techniques

Using Bayesian probability theory, relying on coherent inference based on clearly defined axioms, one can automatically infer how flexible a model is warranted by the data; the Bayesian Occam's razor automatically suppresses the tendency to discover spurious structure in data. In the context of NNs, Bayesian inference techniques offer a number of important benefits including the following (Bishop 1995):

1. regularisation can be given a natural interpretation in the Bayesian framework,
2. error bars can be assigned to the predictions generated by a network,
3. Bayesian methods allow the values of regularisation to be selected using only the training data,
4. the Bayesian approach allows different models to be compared using only the training data,
5. the relative importance of different inputs can be determined using automatic relevance determination (ARD).

Evidence procedure Motivated by the benefits that Bayesian methods are able to bring to NNs, we move into a Bayesian framework to learn the weights in a NN on the basis of a set of training data.

First, we assume a Gaussian prior for the weights:

$$p(\boldsymbol{w}) = \frac{1}{Z_{\boldsymbol{w}}(\alpha)} \exp(-\alpha E_{\boldsymbol{w}}), \tag{4.11}$$

and a likelihood function of the form

$$p(\mathcal{D}|\boldsymbol{w}) = \frac{1}{Z_{\mathcal{D}}(\beta)} \exp(-\beta E_{\mathcal{D}}). \tag{4.12}$$

The correct Bayesian treatment for parameters such as α and β, whose values are unknown, is to integrate them out of any predictions. An alternative approach is to determine the values of the hyperparameters α and β, using a technique called the evidence approximation (MacKay 1995).

The approximation that is made in the evidence procedure is that the posterior density of the hyperparameters, $p(\alpha, \beta|\mathcal{D})$, is sharply peaked around α_{MP}, β_{MP}, the most probable values of the

[4]Methods which use only the input data and not the target data.

parameters. So, we should find the hyperparameter values that optimise the posterior probability of the weights, and then perform the remaining calculations with the hyperparameters set to these values.

To achieve this, we use Bayes' theorem and a non-informative hyperprior, $p(\alpha, \beta)$, therefore we shall just seek to maximise $p(\mathcal{D}|\alpha, \beta)$, called the evidence for α and β, which gives:

$$2\alpha E_{\boldsymbol{w}}^{\mathrm{MP}} = W - \gamma, \tag{4.13}$$

$$2\beta E_{\mathcal{D}}^{\mathrm{MP}} = N - \gamma, \tag{4.14}$$

where $\gamma = W - \alpha \mathrm{Tr}(\boldsymbol{A}^{-1})$ is the number of 'well-determined' parameters, with $\boldsymbol{A}$, the Hessian matrix of the total error, $\beta E_{\mathcal{D}} + \alpha E_{\boldsymbol{w}}$, at $\boldsymbol{w}_{\mathrm{MP}}$.

Automatic relevance determination Another difficulty in NN models is the number of input variables used in modelling the distribution of the targets. We must limit the number of input variables we use, based on our assessment of which attributes are most likely to be relevant.

Associating a separate hyperparameter with each input variable, with careful choice of prior, we can use the posterior distribution to evaluate the importance of each input variable (and to weed out unwanted inputs) for the model predictions. The hyperparameters, which corresponds to decay rates, for irrelevant inputs will automatically be inferred to be large, preventing those inputs from causing significant overfitting (MacKay 1995).

4.2.4 Results and comments

After running simulations, we clearly see that the residuals from raw volumes rather than percentage volumes yield better results (RMS error). Table 4.3 (and Tables D.2, D.3, D.4, D.5) shows that the GLM gives a very good benchmark result of around 0.8 for the testing RMS error. One can also notice that the number of inputs is 5 almost every time, which is justified by the PACF plot in Figure 4.3. Table 4.2 displays RMS error back in the initial space. We clearly see that the GLM (RMS error around 0.77) outperforms the random walk model (RMS error around 0.90). NNs improve on the former RMS error by a small amount. The RBF and the MLP produce similar results. It is remarkable that the complexity of the MLP is very low (1-3 hidden units) and that of RBF is high (8-10). Hence, we advocate the RBF, with many hidden units[5], over the MLP, since firstly, results are similar, secondly training is faster and thirdly evidence approximation is exact in this case.

Hinton diagrams are used in Figure 4.5 as a useful method for visualising the weights in a NN. We clearly see that the biases are much greater than their weight counterparts. However, weights fanning in the third hidden unit are very small. Then, to see the relevance of the different inputs, we look at the magnitude of the hyperparameters, we conclude that the second input is more relevant than the others (where the α's are high), α's for the second layer weights and biases have the same low magnitude.

[5] Overfitting is avoided using Bayesian techniques.

Day	RW	GLM
Monday	0.8292	0.6953
Tuesday	0.9047	0.7585
Wednesday	0.9659	0.7868
Thursday	0.9392	0.7717
Friday	0.9504	0.7743

Table 4.2: RMSE on the initial space (with smoothing splines to estimate the TOD) on the testing set (BNP-Paribas raw volumes). RW is the random walk model, GLM is a single-layer network with 5 input units.

Model	d	m	RMSE	ℓ_{Gauss}	ℓ_{GARCH}
Mondays					
GLM	5	–	0.8225	1.1751	1.0808
MLP	5	3	0.8069	1.1730	1.0805
RBF	5	2	0.8218	1.1908	1.0975
MDN	4	8	0.9561	0.8546	–
Tuesdays					
GLM	5	–	0.8592	1.2084	1.1642
MLP	5	1	0.8590	1.2095	1.1651
RBF	5	8	0.8517	1.2068	1.1632
MDN	5	10	1.0393	0.9105	–
Wednesdays					
GLM	5	–	0.8561	1.1738	1.1410
MLP	5	1	0.8561	1.1739	1.1411
RBF	5	8	0.8505	1.1720	1.1406
MDN	5	6	1.0796	0.9247	–
Thursdays					
GLM	5	–	0.8652	1.1935	1.1470
MLP	5	1	0.8653	1.1933	1.1468
RBF	4	10	0.8590	1.1885	1.1467
MDN	5	9	1.0439	0.9021	–
Fridays					
GLM	5	–	0.8689	1.2244	1.1871
MLP	4	4	0.8571	1.2242	1.1880
RBF	5	6	0.8637	1.2262	1.1889
MDN	5	9	0.9654	0.9228	–

Table 4.3: Prediction on BNP-Paribas (raw additive residuals). The MDN model used contains 3 centres.

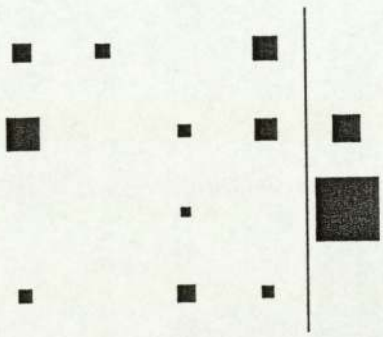
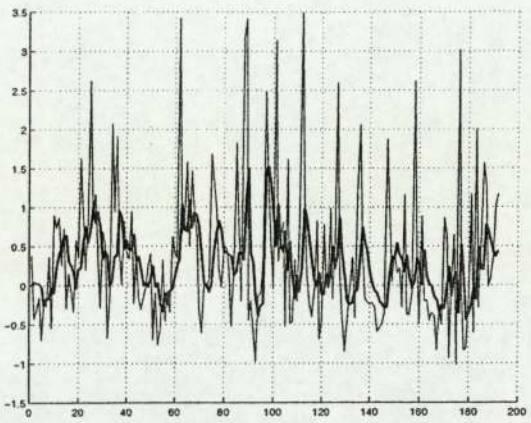
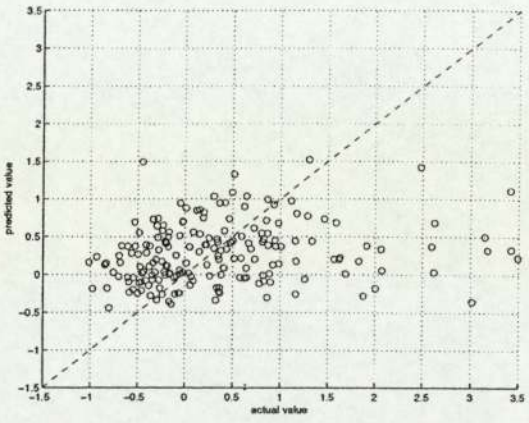


Figure 4.5: Hinton diagram of input layer weights.



(a) Actual values represented by the thin line, predictions by the thick one.



(b) Scatter plot of predictions against actual values. $R^2 = -0.0082$.

Figure 4.6: Prediction of BNP-Paribas normalised additive residuals on Thursdays (2 consecutive days in the testing set) using an RBF network.

Figure 4.6(a) depicts the prediction achieved. We can see that the main dynamics are captured but the prediction is not able to follow high peaks, which is very clear looking at the scatter plot on Figure 4.6(b). We might explain this behaviour by a high α hyperparameter which would 'smooth' the fit too much.

4.2.5 Are the residuals predictable?

Due to the poor results of the prediction, one question arises: are the residuals predictable? More precisely, it should be investigated whether the data has any dynamics by, for example, using statistical tests. To achieve this goal, we use the method of surrogate data (Theiler, Eubank, Longtin, Galdrikian, and Farmer 1992) which is an application of the 'bootstrap' method[6] of modern statistics. Here, we describe the main steps of the surrogate method:

1. generate many surrogate time series, for example by shuffling the time-order of the original time series to test the null hypothesis of iid noise with arbitrary amplitude distribution;

2. compute a discriminant statistic, Q_{H_i}, for each surrogate, such as the forecasting error (on a test set);

3. measure the 'significance', $\mathcal{S} = \frac{|Q_D - \mu_H|}{\sigma_H}$, by the difference between the original and the mean surrogate value of the mean, μ_H, divided by the standard deviation of the surrogate values σ_H.

Using twenty consecutive days for the original time series (raw volumes), a single layer network (with number of inputs determined by CV) as a regressor, and having generated two hundred surro-

[6] An idea for statistical inference, using training sets created by re-sampling with replacement from the original training set, so examples may occur more than once.

gates, we found significance ranging from 0.9 to 3 depending on the day considered. These results lead us to the conclusion that there is evidence for no dynamics and confirm the fact that the residuals are unpredictable.

4.2.6 Conditional variance

Because we are in a probabilistic framework, we are able to compute the likelihood of our model parameters. Equivalently, this can be viewed as the fitting of the noise, $\epsilon_t = x_t - y_{\mathcal{M}}(\boldsymbol{x}^{(t)}; \boldsymbol{w}_{\text{MP}})$, from our model, $\mathcal{M}$.

The usual assumption is that the noise process, ϵ, is Gaussian, $\mathcal{N}(0, \sigma^2)$, where σ^2 is a fixed variance. We can enhance this model by assuming a time-varying variance σ_t as:

$$\sigma_t^2 = \omega + \alpha\sigma_{t-1}^2 + \beta\epsilon_{t-1}^2. \tag{4.15}$$

With a conditional variance dictated by (4.15), the process $\{\epsilon_t\}$ is called a Generalised Autoregressive Condition Heteroskedasticity (GARCH) process[7] (Bollerslev 1986). Moreover, since the GARCH(1,1) model may provide parsimonious representation for some complex autodependence structure that can only be produced by an ARCH(p) model with large p, we will not use higher lags than that in the GARCH model. In fact, the GARCH(1,1) model has been very successful in empirical work and is regarded as the benchmark model by many econometricians.

The results concerning the likelihood of the different models shows that the GARCH(1,1) model fits the noise better than the usual Gaussian noise (Table 4.3).

4.2.7 Modelling conditional distribution

Instead of predicting the residuals, since it is a very difficult task, we focus on increasing the likelihood of the previous models. According to Section 4.1.2, the GMM was quite powerful in estimating the unconditional distribution, $p(x_t)$, therefore we model the conditional distribution of the residuals using GMM.

Hence, if we let the mixture model parameters $\boldsymbol{\theta}$ be functions of the input vector $\boldsymbol{x}^{(t)} := (x_{t-1}, \ldots, x_{t-d})$ then we can model the distribution conditional on $\boldsymbol{x}^{(t)}$. Since, in general, the mapping, $\boldsymbol{x}^{(t)} \mapsto \boldsymbol{\theta}(\boldsymbol{x}^{(t)})$, will be complicated, we use an MLP to model it. Thus the model has the form:

$$p(x_t|\boldsymbol{x}^{(t)}) = \sum_{j=1}^{M} \alpha_j(\boldsymbol{x}^{(t)})\phi_j(x_t|\boldsymbol{x}^{(t)}), \tag{4.16}$$

where M is the number of components in the mixture. This combination of a NN and a mixture model is known as a mixture density network (MDN) (Bishop 1995; Nabney 2002).

The MDN model assumes a non-Gaussian noise without dynamics in time whereas the GARCH model assume a Gaussian noise with dynamical time dependence.

As expected, the predictions are worse than the GLM, however the likelihood of the MDN outperforms the GARCH likelihood (see results in Table 4.3).

[7] To be precise, it is the GARCH(1,1) model.

4.2.8 Conclusions

Concerning forecasting, simple models such as the GLM perform quite well. More sophisticated techniques such as Bayesian learning for multi-layer neural networks did not improve our results significantly. This can be explained by the low time dependence found in the time series studied. Moreover, the modelling of the conditional variance was achieved by a mixture density network which yielded good results and clearly outperformed benchmarks as the GARCH model.

Chapter 5

Feature Extraction: the UK Bank Sector

Forecasting with a single stock was proved to be difficult. However applying component analysis techniques considering a whole sector should give us insight into underlying forces driving the sector, and also these latent variables may be easier to predict. We hope that the source signals we will extract will have simple patterns such that their prediction will be an easy task, allowing us to go back to the initial space to forecast the actual stocks. In fact, multivariate data are often viewed as multiple indirect measurements arising from an underlying source, which typically cannot be measured directly.

Therefore, in this chapter, we will consider multivariate volume time series with the following matrix $[\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(i)}, \ldots, \boldsymbol{x}^{(N)}]$, where i denotes the stock considered and $\boldsymbol{x}^{(i)}$ its residual time series.

5.1 Principal component analysis

Principal component analysis (PCA) is a powerful technique for extracting structure from high-dimensional data sets. PCA consists of an orthogonal transformation of the coordinate system in which we describe our data. The new coordinate system is obtained by projection onto the so-called principal axes of the data. The important feature of PCA is that it retains maximal information, by retaining variance of projected data, amongst all linear projections.

Theorem 2 (Principal component analysis). *Given a set of observations $\boldsymbol{x}^{(i)} \in \mathbb{R}^d, i = 1, \ldots, N$, which are centered, $\sum_{i=1}^{N} \boldsymbol{x}^{(i)} = 0$, PCA finds the principal axes by diagonalising the covariance matrix,*

$$C = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{x}^{(i)} \boldsymbol{x}^{(i)\top}. \tag{5.1}$$

Unfortunately, there is no general technique to decide how many principal components should be used to represent the data adequately. Commonly, the number of principal components are chosen by

looking at the eigen spectrum of the data shown in Figure 5.1.

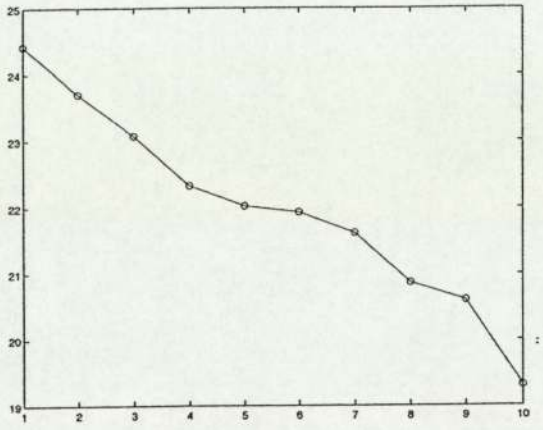


Figure 5.1: Log eigen spectrum of the UK bank sector. x-axis corresponds to eigen-values, y-axis to amplitude.

5.1.1 Epps effect

(Epps 1979) reported empirical evidence of a dramatic drop in correlations among stocks when decreasing the sampling time horizon. In other words, it shows that the higher the frequency in the intraday data, the smaller the correlation. This phenomenon has been observed across different markets. We investigate this pattern in the residuals of the UK bank sector by increasing the tick-time from five minutes to sixty minutes.

Figure 5.2 depicts the average value of the off-diagonal elements of the correlation matrix and the percentage of information (variance) represented by the first principal component[1].

We still see very low values for the average correlation matrix value although they are increasing, which confirm again the difficulty of modelling the residuals.

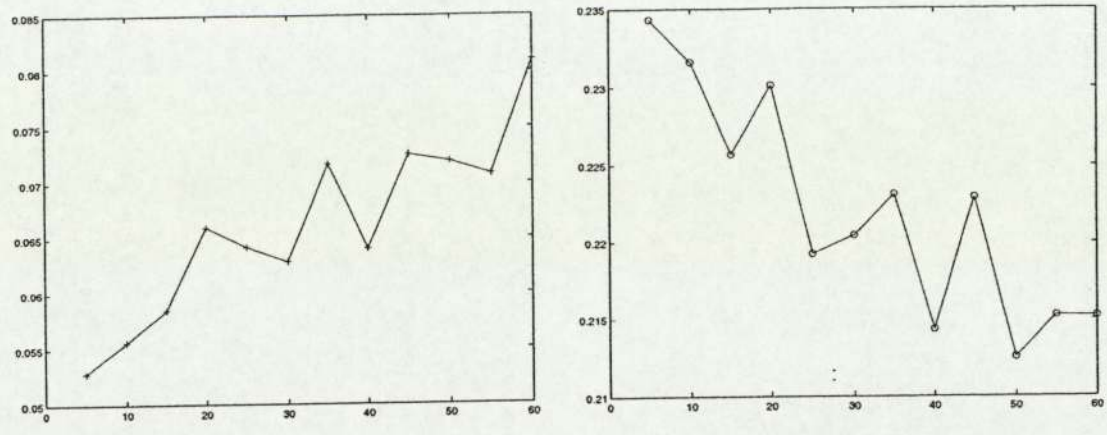
5.1.2 Automatic choice of dimensionality

(Tipping and Bishop 1999) showed how PCA can be reformulated as the maximum likelihood solution of a specific latent variable model, called probabilistic PCA (PPCA), as follows. We first introduce a k-dimensional latent variable $\boldsymbol{z}$ whose prior distribution is a zero mean Gaussian $p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}|0, \boldsymbol{I}_k)$. The observed variable $\boldsymbol{x}$ is then defined as a linear transformation of $\boldsymbol{z}$ with additive Gaussian noise, $\boldsymbol{x} = \boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}$, where $\boldsymbol{W}$ is a $d \times k$ matrix and $\boldsymbol{\epsilon}$ is a zero mean normally distributed vector with covariance $\sigma^2 \boldsymbol{I}_d$. Thus $p(\boldsymbol{x}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu}, \sigma^2 \boldsymbol{I}_d)$. The marginal distribution of the observed variable is then given by

$$p(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{C}), \tag{5.2}$$

with the covariance matrix $\boldsymbol{C} = \boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I}_d$.

[1] $\frac{|\lambda_{max}|}{\sum_i \lambda_i}$ where λ's are the eigenvalues of the covariance matrix.



(a) Average value of non-diagonal elements

(b) Percentage variance explained by the first principal component

Figure 5.2: Properties of the covariance matrix of the UK Bank sector against tick-time frequency.

Armed with the probabilistic reformulation of PCA, a Bayesian treatment is obtained by first introducing a prior distribution, $p(\boldsymbol{\mu}, \boldsymbol{W}, \sigma^2)$, over the parameters of the model. Then the evidence of intrinsic dimensionality k is given by

$$p(\mathcal{D}|k) = \iiint p(\mathcal{D}|\boldsymbol{W}, \boldsymbol{\mu}, \sigma^2) p(\boldsymbol{\mu}, \boldsymbol{W}, \sigma^2) \, \mathrm{d}\boldsymbol{\mu} \, \mathrm{d}\boldsymbol{W} \, \mathrm{d}\sigma^2. \tag{5.3}$$

(Minka 2001) uses Laplace's method (see Appendix C.2) to compute the integral (5.3). Applying this technique to our dataset, we still have a high dimensionality which is 7 or 8 out of 10 (Table 5.1).

day	k
Monday	7
Tuesday	7
Wednesday	7
Thursday	7
Friday	8

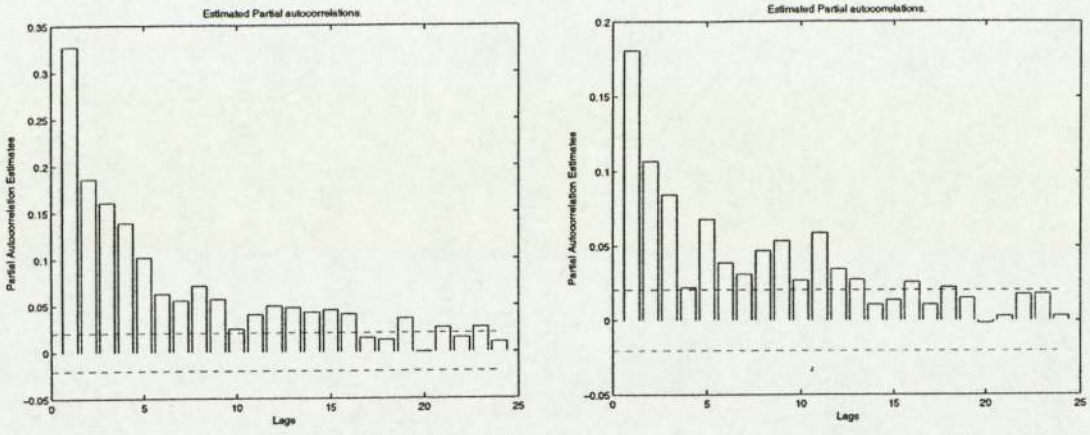
Table 5.1: Intrinsic dimensionality of the UK bank sector (raw volumes).

5.2 Independent component analysis

The PCA methods always produces orthogonal vectors but the assumption of orthogonality is obviously invalid for a general mixing process and should be removed. Instead, we shall assume that the sources are statistically independent: this is the independent component analysis (ICA) model.

Definition 9 (Noise-free (ICA) model). *Independent component analysis of a vector $\boldsymbol{x}$ consist of estimating the following generative model for the data:*

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{s}, \tag{5.4}$$



(a) IC number 1

(b) IC number 2

Figure 5.3: PACF of the ICs of the UK bank sector (Mondays' raw volumes).

where the latent variables, s_i, in the vector, $\boldsymbol{s} = [s_1, s_2, \ldots, s_n]^\top$, are assumed to be independent. The matrix $\boldsymbol{A}$ is a constant $m \times n$ 'mixing' matrix.

The identifiability of this model can be assured (Hyvärinen 1999) if:

1. all the independent components (IC) s_i, with the possible exception of one component, are non-Gaussian,
2. the number of observed linear mixtures, m, is at least as large as the number of ICs, n,
3. the matrix $\boldsymbol{A}$ is of full column rank.

What distinguishes ICA from PCA is that the non Gaussian structure of the data is taken into account in ICA. Intuitively, lack of correlation determines the second degree cross-moments (covariances) of a multivariate distribution, while, in general, statistical independence determines all of the cross-moments.

The ICA model is implemented with the the FastICA algorithm (Hyvärinen 1999).

Figure 5.3 depicts the partial autocorrelation function of the two first source signals. We can see that their structure is not simpler than the observed (original) signals. In fact, these ICs appear very similar to the original signal which is confirmed by the high intrinsic dimensionality found in Table 5.1.

Hence this method is not beneficial for this project.

Chapter 6

Conclusion

6.1 Summary of the work done

This thesis has given an account of the work produced by conducting research on the problem of forecasting in finance. Below are the main points that have been covered in this thesis.

After trying to model the seasonality of the time series using spline based methods, we found that an equivalent Bayesian approach, via kernel methods and Gaussian processes, was more general.

Having removed the estimated seasonality, we have studied the non Gaussian residual distribution and autocorrelation structure. Then, we tried to predict, via phase space methods, the residuals using simple and efficient technique, namely the generalised linear model, and also more sophisticated models such as multi-layer neural networks. Once again, we used Bayesian perspectives to add considerable insight to the fitting of neural networks. But, the generalised linear model performed quite well compared to neural networks. This may be explained by the low time dependence we found in the time series by running statistical tests on surrogate time series.

Moreover, we modelled the conditional distribution of the residuals using mixture density networks and also the conditional noise using a GARCH model. The mixture density network model outperformed the GARCH benchmark.

In addition, instead of focusing on a single stock, we tried to extract linear components of the UK bank sector (with PCA and ICA), but the sources signals extracted revealed a behaviour not simpler than that of the original individual signals.

6.2 Further work

Impact of news Based on the forecasting model we set up, it would be interesting to evaluate the impact of American and European macroeconomic news. (Andersen, Bollerslev, Diebold, and Vega 2003) provides an empirical examination of price discovery in the context of foreign exchange, in particular it s hows that announcement surprises produce conditional mean jumps. Moreover,

it finds that the adjustment pattern is characterised by a sign effect. (Omrane, Bauwens, and Giot 2003) deals with the impact of scheduled and unscheduled news announcements on the Euro/Dollar return volatility. It is shown, using high-frequency intraday data and within the framework of ARCH-type and realised volatility models, that volatility increases in the pre-announcements periods, particularly before scheduled events. Another approach would be to think of it as an impulse function (a simple modelling would be a Gaussian kernel) with limited duration. This former duration could be evaluated looking at the likelihood of the model.

Stationarity of the time-of-day function Another important work concerns the time-of-day function. First, the stationarity of the time-of-day estimate should be assessed, that is to say, the question, "how often (in time unit of, for example number of days) should the estimate be recomputed", should be answered.

Robust regression Since the residuals exhibit an asymmetric heavy-tailed distribution, lack of robustness to outliers is a crucial drawback. Hence, a robust noise model should be employed (Bishop and Svensén 2004; Tipping and Lawrence 2003; Roberts and Penny 2002).

Feature extraction The techniques we used, namely PCA and ICA, to extract features relied on linear combinations of existing features. A different approach could be to investigate algorithms which extract nonlinear structures in the data such as Kernel PCA (Müller, Mika, Rätsch, Tsuda, and Schölkopf 2001).

6.3 Afterword

This project has been interesting in many aspects, from the involvement of abstract machine learning theory to its practical financial application. It has given us the opportunity to use different areas of expertise such as applied mathematics, statistics, computer science and finance to achieve the results obtained. In conclusion, unfortunately, no one model is perfect but we hope that further research will be conducted to improve the machine learning approach concerning finance.

Bibliography

Andersen, T. G., T. Bollerslev, F. X. Diebold, and C. Vega (2003). Micro Effects of Macro Announcements: Real-Time Price Discovery in Foreign Exchange. *American Economic Review* **93**(1), 38–62.

Azzouzi, M. (2003a). A comparison of simple VWAP strategies. Technical report, Equity Linked Products, HVB Corporates & Markets, London.

Azzouzi, M. (2003b). An empirical analysis of intraday volumes. Technical report, Equity Linked Products, HVB Corporates & Markets, London.

Azzouzi, M. (2004). Personal communication with Mehdi Azzouzi. Conversations and e-mails.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. New York: Oxford University Press.

Bishop, C. M. and M. Svensén (2004). Robust Bayesian mixture modelling. In *European Symposium on Artificial Neural Networks*, Volume 12. Accepted for publication.

Bishop, C. M., M. Svensén, and C. K. Williams (1998). The Generative Topographic Mapping. *Neural Computation* **10**(1), 215–234.

Bishop, C. M. and M. E. Tipping (2003). Bayesian regression and classification. In J. Suykens, G. Horvath, S. Basu, C. Micchelli, and J. Vandewalle (Eds.), *Advances in Learning Theory: Methods, Models and Applications*, Volume 190 of *NATO Science Series III: Computer and Systems Sciences*. IOS Press.

Bollerslev, T. (1986). Generalized Autoregressive Conditional Heteroskedasticity. *Journal of Econometrics* **31**, 307–327.

Csató, L. and M. Opper (2002). Sparse On-Line Gaussian Processes. *Neural Computation* **14**(3), 641–668.

de Boor, C. (2001). *A Practical Guide to Splines* (Revised ed.). New York: Springer-Verlag.

Engle, R. F. (1982). Autoregressive Conditional Heteroskedasticity With Estimates of the Variance of UK Inflation. *Econometrica* **50**, 987–1008.

Engle, R. F. and J. R. Russell (1998). Autoregressive Conditional Duration : A New Model for Irregularly Spaced Transaction Data. *Econometrica* **66**(5), 1127–1162.

Engle, R. F. and J. R. Russell (2002, October). Analysis of High Frequency Data. Unpublished.

Epps, T. (1979). Comovements in Stock Prices in the Very Short Run. *Journal of the American Statistical Association* **74**, 291–298.

Girosi, F., M. Jones, and T. Poggio (1995). Regularization Theory and Neural Networks Architectures. *Neural Computation* **7**(2), 219–269.

Hastie, T., R. Tibshirani, and J. Friedman (2001). *The Elements of Statistical Learning.* Spinger Series in Statistics. New York: Springer-Verlag.

Hyvärinen, A. (1999). Survey on Independent Component Analysis. *Neural Computing Surveys* **2**, 94–128.

Kantz, H. and T. Schreiber (2004). *Nonlinear Time Series Analysis* (Second ed.). Cambridge: Cambridge University Press.

Lo, A. W. and J. Wang (2000). Trading volume: Definitions, Data Analysis and Implications of Portfolio Theory. *The Review of Financial Studies* **13**(2), 257–300.

Lowe, D. (1995). On the use of nonlocal and non positive definitive basis functions in radial basis function networks. In *International Conference on Artificial Neural Netwoks*, Volume 4.

MacKay, D. J. C. (1995). Probable networks and plausible predictions — a review of practical Bayesian methods for supervised neural networks. *Network: Computation in Neural Systems* **6**(3), 469–505.

MacKay, D. J. C. (2003). *Information Theory, Inference and Learning Algorithms.* Cambridge: Cambridge University Press.

Minka, T. P. (2001). Automatic choice of dimensionality for PCA. In T. K. Leen, T. G. Dietterich, and V. Tresp (Eds.), *Advances in Neural Information Processing Systems*, Volume 13, Cambridge, MA. The MIT Press.

Müller, K.-R., S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf (2001, March). An Introduction to Kernel-Based Learning Algorithms. *IEEE Transactions on Neural Networks* **12**(2), 181–202.

Nabney, I. T. (2002). *Netlab: Algorithms for Pattern Recognition.* Advances in Pattern Recognition. London: Springer-Verlag.

Omrane, W. B., L. Bauwens, and P. Giot (2003). News announcements, market activity and volatility in the Euro/Dollar foreign exchange market. Technical report, CORE, Université catholique de Louvain. DP 2003/11, forthcoming in Journal of International Money and Finance.

Ripley, B. D. (1996). *Pattern Recognition and Neural Networks.* Cambridge: Cambridge University Press.

Roberts, S. J. and W. D. Penny (2002, September). Variational Bayes for generalized autoregressive models. *IEEE Transactions on Signal Processing* **50**(9), 2245–2257.

Schölkopf, B. and A. J. Smola (2002). *Learning with Kernels.* Cambridge, MA: The MIT Press.

Seeger, M. (2004). Gaussian Processes for Machine Learning. *International Journal of Neural Systems* **14**(2), 69–106.

Takens, F. (1981). Detecting strange attractors in turbulence. In D. A. Rand and L. Young (Eds.), *Dynamical systems and turbulence - Warwick 1980*, Volume 898 of *Lectures Notes in Mathematics*, Berlin, pp. 366. Springer-Verlag.

Theiler, J., S. Eubank, A. Longtin, B. Galdrikian, and J. D. Farmer (1992). Testing for nonlinearity in time series: the method of surrogate data. *Physica D* **58**, 77–94.

Tipping, M. E. (2001). Sparse Bayesian Learning and the Relevance Vector Machine. *Journal of Machine Learning Research* **1**, 211–244.

Tipping, M. E. and C. M. Bishop (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society, Series B* **21**(3), 611–622.

Tipping, M. E. and N. D. Lawrence (2003). A variational approach to robust Bayesian interpolation. In C. Molina, T. Adali, J. Larsen, M. V. Hulle, S. Douglas, and J. Rouat (Eds.), *IEEE International Workshop on Neural Networks for Signal Processing*, Volume 13.

Appendix A

Computational Considerations for Splines

The idea is to find a smooth function that minimises the residual sum of squares (RSS). A popular measure of roughness of a function f is $\|f''\|_2^2$. By the Lagrange multiplier method, minimising the RSS subject to the roughness constraint is equivalent to the following penalised least-squares problem: minimising

$$\sum_{i=1}^{N}\{y_i - f(x_i)\}^2 + \lambda \int \{f''(t)\}^2 \, \mathrm{d}t, \tag{A.1}$$

with respect to f, where λ is a fixed smoothing parameter (Lagrange multiplier). The first term measures the closeness of fit to the data, while the second term penalises curvature in the function, and λ establish a tradeoff between the two. Two special cases are:

$\lambda = 0$: f can be any function that interpolates the data,

$\lambda = +\infty$: the simple least squares line fit, since the second derivative must be zero.

As λ ranges from zero to infinity, the estimate ranges from the most complex model (interpolation) to the simplest model (linear model). Thus, the model complexity of the smoothing spline approach is effectively controlled by the smoothing parameter λ. The estimator, $\hat{f}_\lambda$, is a spline function and is referred to as a smoothing spline estimator.

The criterion (A.1) is defined on an infinite-dimensional function space[1]. Remarkably, it can be shown that (A.1) has an explicit, finite-dimensional, unique minimiser which is a natural cubic spline with knots at the unique values of the x_i, $i = [\![1, N]\!]$. We write:

$$f(x) = \sum_{j=1}^{N+2} \gamma_j B_j(x), \tag{A.2}$$

where γ_j's are coefficients and B_j is the cubic B-spline basis function $N_{j,3}$.

[1] A Sobolev space of functions for which the second term is defined.

The criterion thus reduces to

$$\mathrm{RSS}(\gamma, \lambda) = (\boldsymbol{y} - \boldsymbol{B}\gamma)^{\mathsf{T}}(\boldsymbol{y} - \boldsymbol{B}\gamma) + \lambda\gamma^{\mathsf{T}}\boldsymbol{\Omega}_B\gamma, \tag{A.3}$$

where

$$B_{ij} = B_j(x_i) \quad \text{and} \quad \Omega_{Bij} = \int B_i''(t)B_j''(t)\,\mathrm{d}t. \tag{A.4}$$

Setting the derivative with respect to γ to zero gives the equation

$$(\boldsymbol{B}^{\mathsf{T}}\boldsymbol{B} + \lambda\boldsymbol{\Omega}_B)\hat{\gamma} = \boldsymbol{B}^{\mathsf{T}}\boldsymbol{y}. \tag{A.5}$$

Since the columns of $\boldsymbol{B}$ are the evaluated B-splines, in order from left to right and evaluated at the sorted values of X, and the cubic B-splines have local support, $\boldsymbol{B}$ is lower 4-banded. Consequently, the matrix $\boldsymbol{M} = \boldsymbol{B}^{\mathsf{T}}\boldsymbol{B} + \lambda\boldsymbol{\Omega}_B$ is 4-banded and hence its Cholesky decomposition, $\boldsymbol{M} = \boldsymbol{L}\boldsymbol{L}^{\mathsf{T}}$, can be computed easily. The equation $\boldsymbol{L}\boldsymbol{L}^{\mathsf{T}}\hat{\gamma} = \boldsymbol{B}^{\mathsf{T}}\boldsymbol{y}$ can then be solved by back-substitution to give $\hat{\gamma}$ and hence the solution $\hat{f}$ in $\mathcal{O}(N)$ operations.

Appendix B

Kernels

Definition 10 (Reproducing kernel Hilbert space). *Let χ be a nonempty set and by H a Hilbert space of functions $f : \chi \mapsto \mathbb{R}$. Then H is called a reproducing kernel Hilbert space endowed with the dot product $\langle \cdot, \cdot \rangle$ if there exists a function $k : \chi^2 \mapsto \mathbb{R}$ with the following properties:*

1. *k has the reproducing property $\langle f, k(x, \cdot) \rangle = f(x)$ for all $f \in H$,*
2. *k spans H.*

Theorem 3 (Mercer). *Suppose $k \in L_\infty(\chi^2)$ ((χ, μ) is a finite measure space) is a symmetric real-valued function that the integral operator $T_k : L_2(\chi) \mapsto L_2(\chi)$ defined by $(T_k f)(x) = \int_\chi k(x, x') f(x') \, \mathrm{d}\mu(x')$ is a positive definite.*

Let $\psi_j \in L_2(\chi)$ be the normalised orthogonal eigenfunctions of T_k associated with the eigenvalues $\lambda_j > 0$, sorted in decreasing order. Then

1. *$(\lambda_j)_j \in \ell_1$,*
2. *$k(x, x') = \sum_{j=1}^{N_H} \lambda_j \psi_j(x) \psi_j(x')$ holds for almost all (x, x'). Either $N_H \in \mathbb{N}$, or $N_H = \infty$; in the latter case, the series converges absolutely and uniformly for almost all (x, x').*

Appendix C

Miscellaneous theoretical details

C.1 K-fold cross-validation

Let $\kappa : \{1, \ldots, N\} \mapsto \{1, ..., K\}$ be an indexing function that indicates the partition to which observation i is allocated by the randomisation. Given a set of models, $f_\alpha(x)$, indexed by a tuning parameter, α, denote $\hat{f}_\alpha^{-k}(x)$ the αth model fit with the kth part of the data removed. Then for this set of models we define the cross-validation estimate of prediction error

$$\mathrm{CV}(\alpha) = \frac{1}{N} \sum_{i=1}^{N} L(y_i, \hat{f}_\alpha^{-\kappa(i)}(x_i)). \tag{C.1}$$

C.2 Laplace approximation

We seek to approximate $Z_P \equiv \int P(x)\,\mathrm{d}x$. We assume that P has a maximum at a point x_0. By a Taylor expansion at the point x_0, we have

$$\ln P(x) = \ln P(x_0) - \frac{c}{2}(x - x_0)^2 + \mathcal{O}(x^3), \tag{C.2}$$

where

$$c = -\frac{\partial^2}{\partial x^2} \ln P(x)\Big|_{x=x_0}. \tag{C.3}$$

We then approximate P by an unnormalised Gaussian Q:

$$Q(x) = P(x) \exp\left[-\frac{c}{2}(x - x_0)^2\right], \tag{C.4}$$

and it follows

$$Z_P \simeq P(x_0)\sqrt{\frac{2\pi}{c}}. \tag{C.5}$$

Similarly, on a K-dimensional space, we have:

$$Z_P \simeq P(x_0)\sqrt{\frac{(2\pi)^K}{\det \boldsymbol{A}}}, \tag{C.6}$$

with $\boldsymbol{A}$ the Hessian matrix of $-\ln P$ evaluated at the point x_0.

Appendix D

Simulation Results

Day	MA	Smooth	GP	RVM	RBF
Alcatel					
Monday	0.8446	0.8891	0.8390	0.8389	0.8393
Tuesday	0.8765	0.8661	0.8684	0.8694	0.9842
Wednesday	0.8792	0.8748	0.8753	0.8753	0.8866
Thursday	0.8921	0.8783	0.8850	0.8894	0.8886
Friday	0.9127	0.8913	0.8939	0.8922	0.9208
France Télécom					
Monday	0.8307	0.8299	0.8292	0.8301	1.0302
Tuesday	0.8758	0.8671	0.8702	0.8708	0.9230
Wednesday	0.8944	0.8938	0.8938	0.8940	0.8936
Thursday	0.8791	0.8699	0.8741	0.8750	0.8880
Friday	0.9018	0.8945	0.8934	0.8974	0.9441
LVMH					
Monday	0.8742	0.8699	0.8697	0.8707	0.8315
Tuesday	0.8706	0.8631	0.8651	0.8649	6.7208
Wednesday	0.8906	0.8881	0.8888	0.8891	0.8792
Thursday	0.8886	0.8764	0.8790	0.8803	0.8678
Friday	0.9188	0.9046	0.9063	0.9056	0.9038
Vivendi Universal					
Monday	0.8289	0.8255	0.8250	0.8249	0.8340
Tuesday	0.8663	0.8579	0.8623	0.8627	2.6845
Wednesday	0.8760	0.8733	0.8742	0.8746	0.9004
Thursday	0.8664	0.8595	0.8606	0.8641	0.8791
Friday	0.8994	0.8932	0.8947	0.8965	0.9337

Table D.1: RMSE on the time-of-day function on the testing set (Alcatel, France Télécom, LVMH, Vivendi Universal relative volumes).

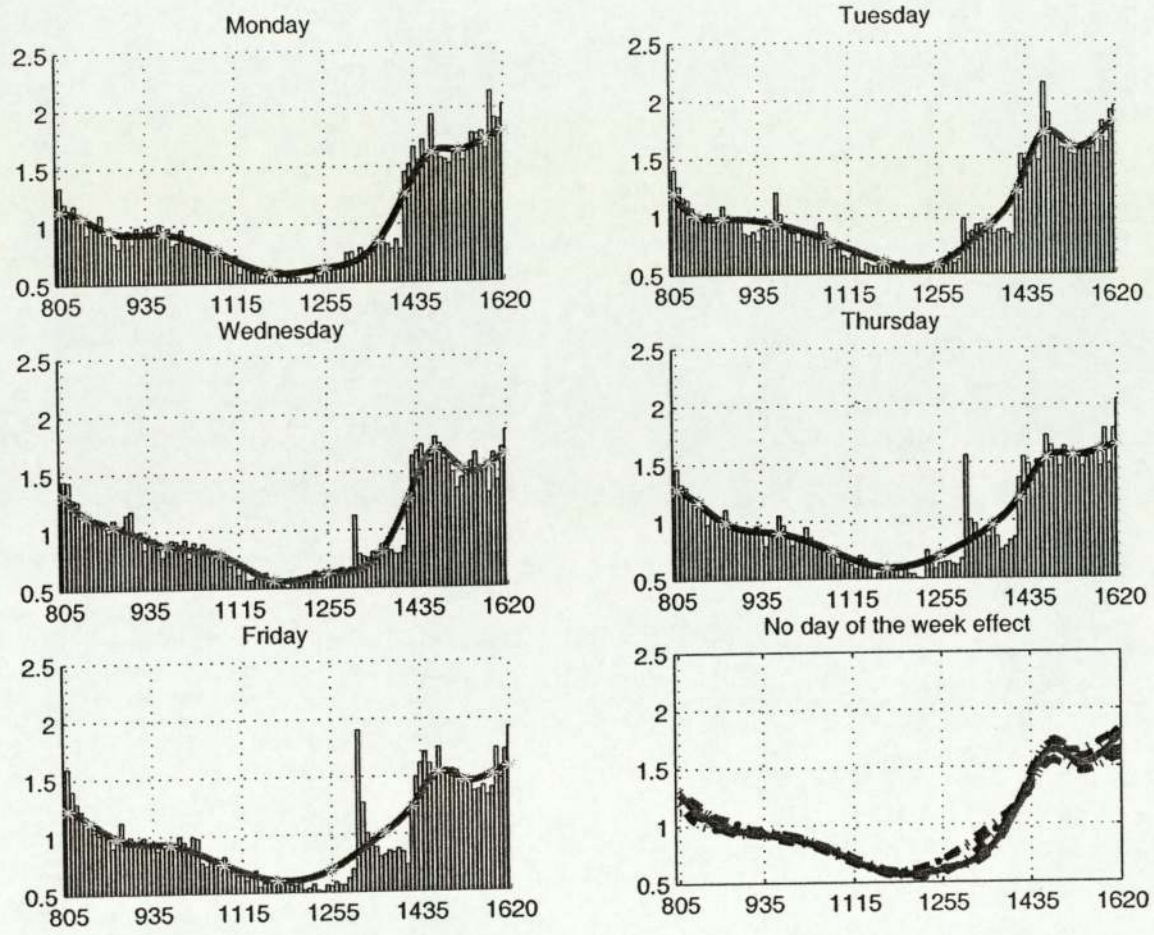


Figure D.1: Alcatel time-of-day estimate using moving average and cubic splines. The histograms represent the whole relative volume dataset.The stars correspond to the knots.

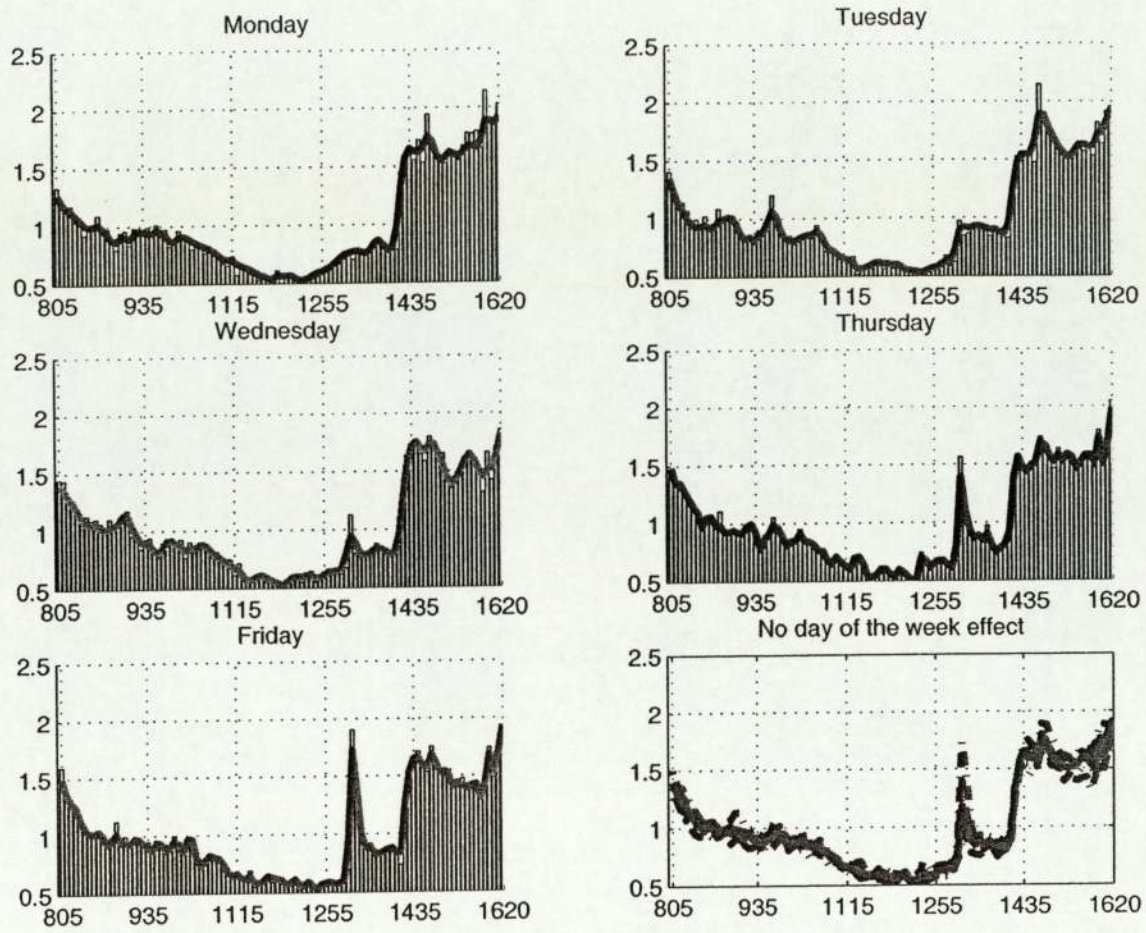


Figure D.2: Alcatel time-of-day estimate using cubic smoothing splines. The histograms represent the whole relative volume dataset.

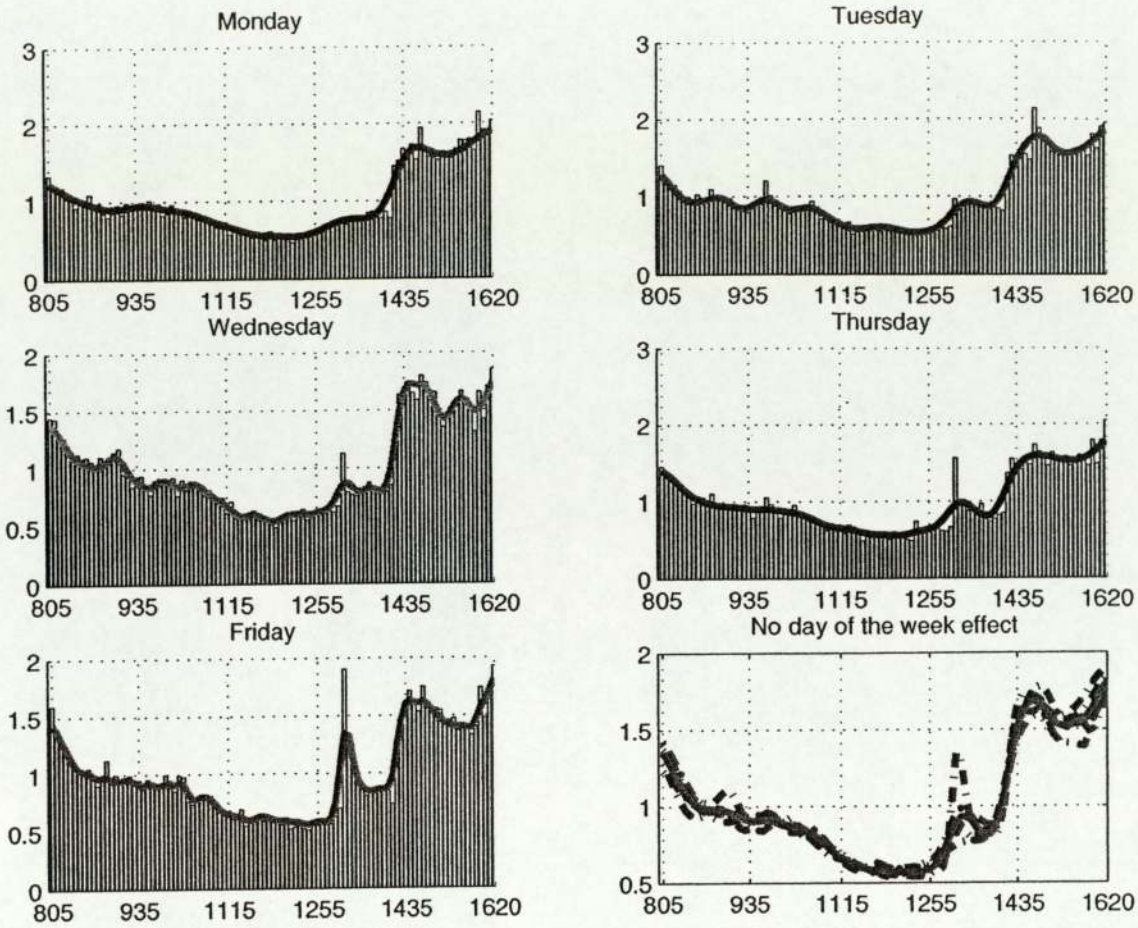


Figure D.3: Alcatel time-of-day estimate using Gaussian processes. The histograms represent the whole relative volume dataset.

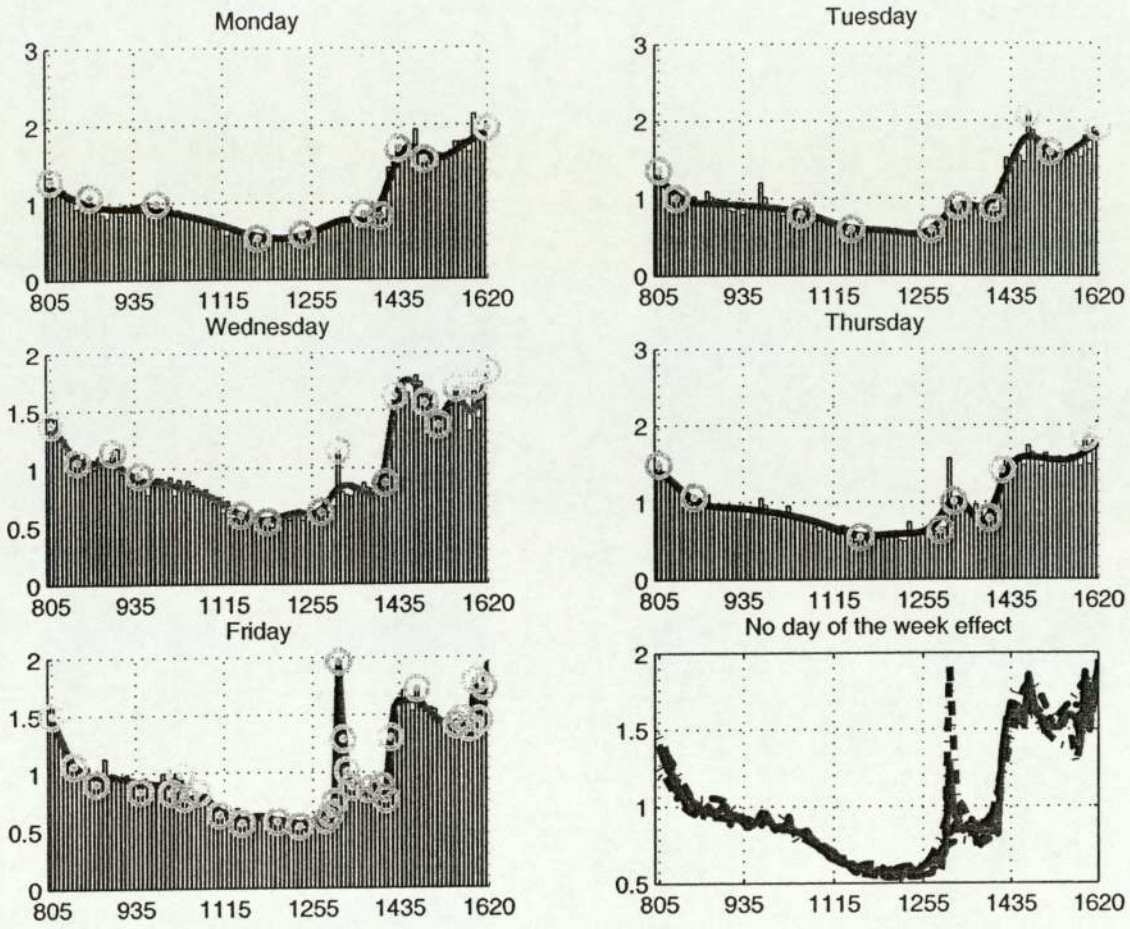


Figure D.4: Alcatel time-of-day estimate using relevance vector machine. The histograms represent the whole relative volume dataset. The circled points correspond to the relevance vectors.

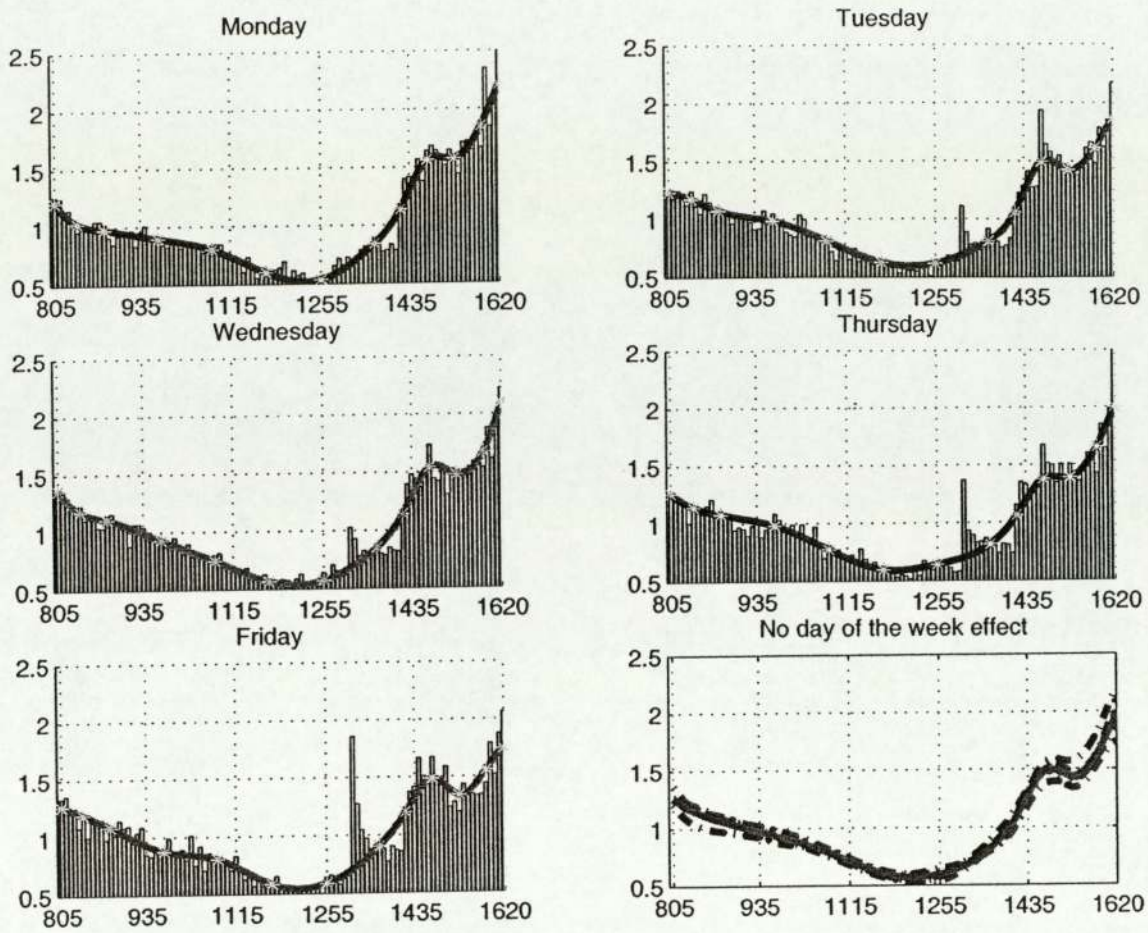


Figure D.5: France Télécom time-of-day estimate using moving average and cubic splines. The histograms represent the whole relative volume dataset.The stars correspond to the knots.

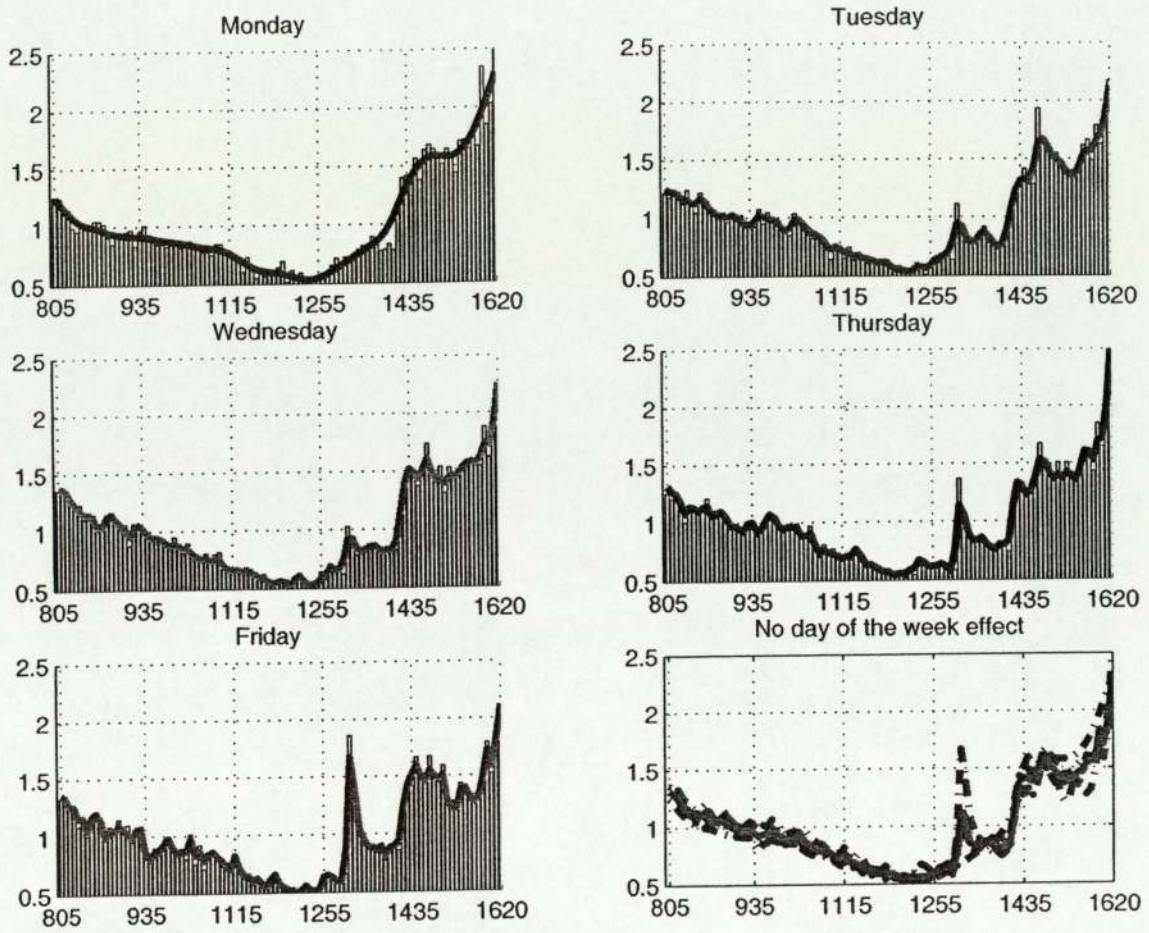


Figure D.6: France Télécom time-of-day estimate using cubic smoothing splines. The histograms represent the whole relative volume dataset.

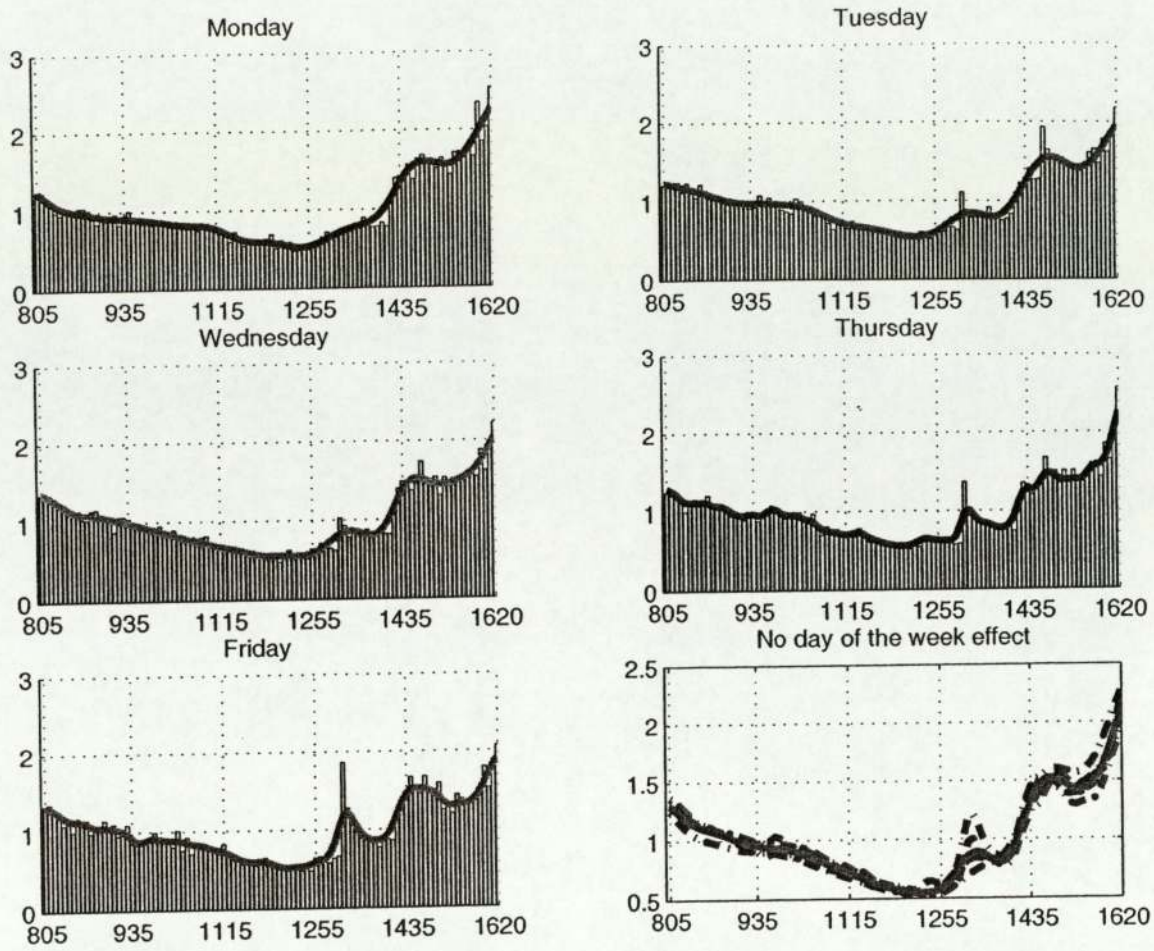


Figure D.7: France Télécom time-of-day estimate using Gaussian processes. The histograms represent the whole relative volume dataset.

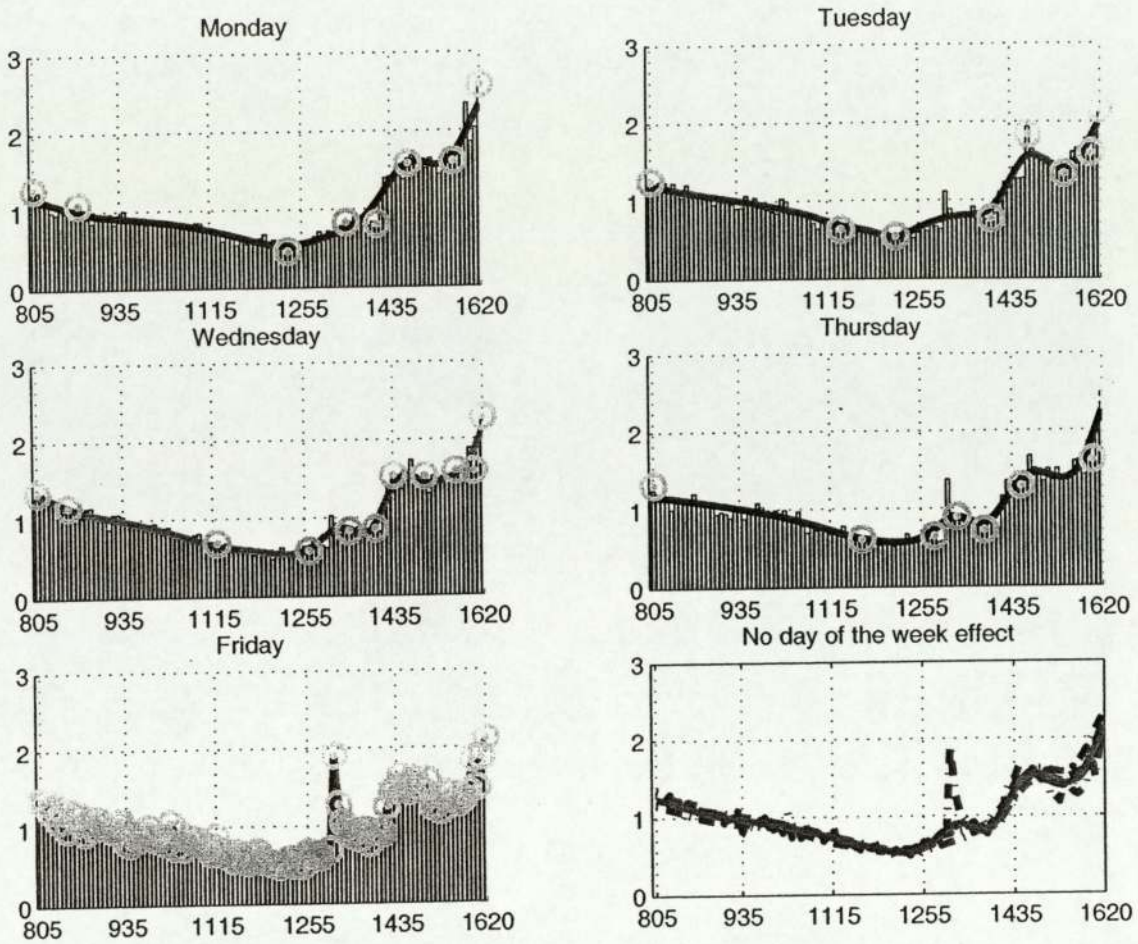


Figure D.8: France Télécom time-of-day estimate using relevance vector machine. The histograms represent the whole relative volume dataset. The circled points correspond to the relevance vectors.

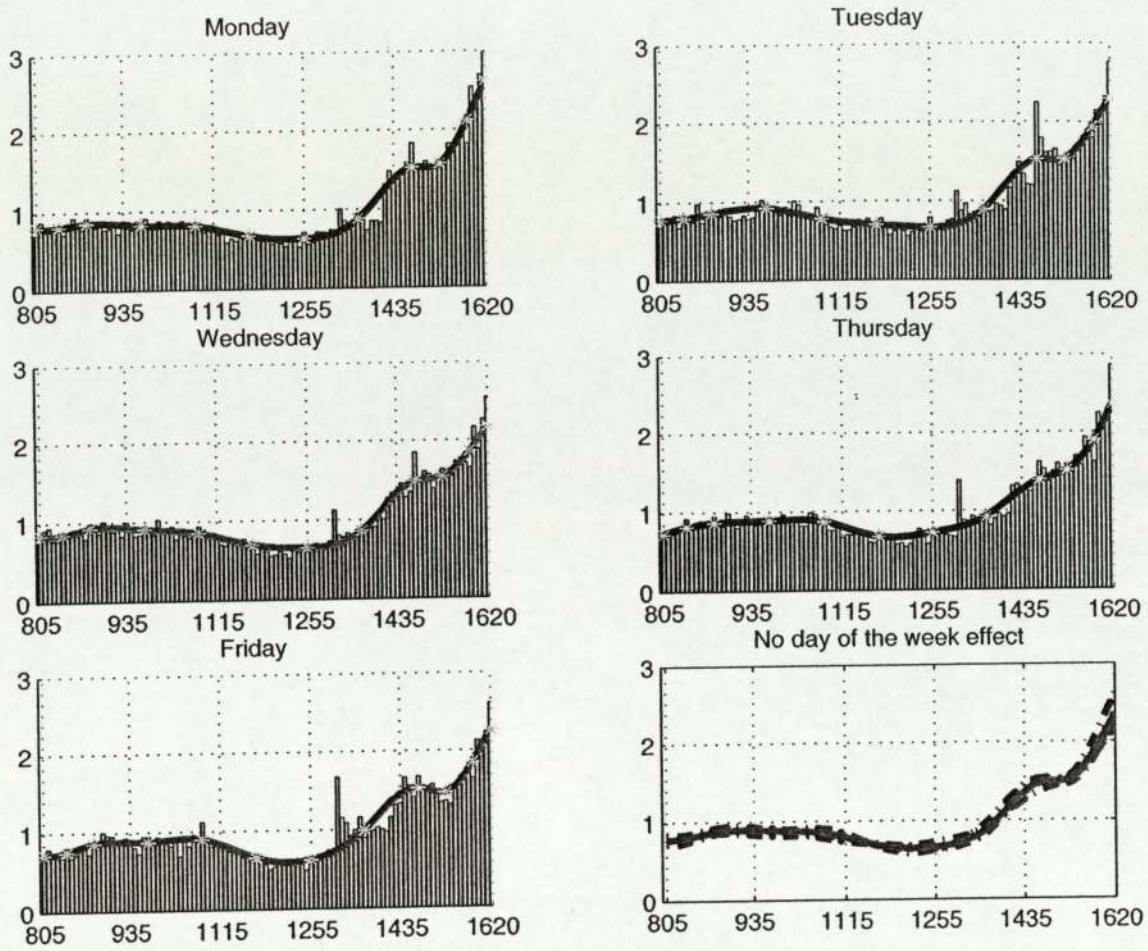


Figure D.9: LVMH time-of-day estimate using moving average and cubic splines. The histograms represent the whole relative volume dataset. The stars correspond to the knots.

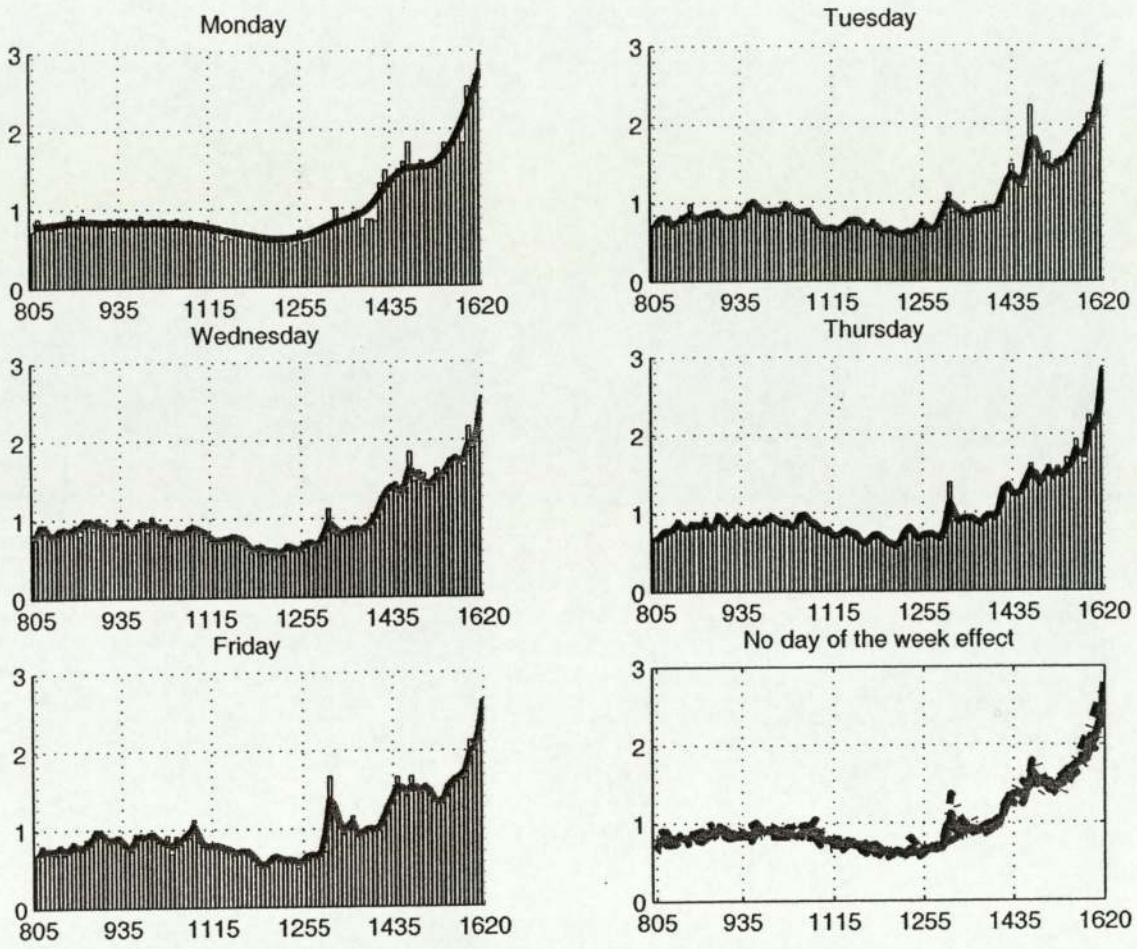


Figure D.10: LVMH time-of-day estimate using cubic smoothing splines. The histograms represent the whole relative volume dataset.

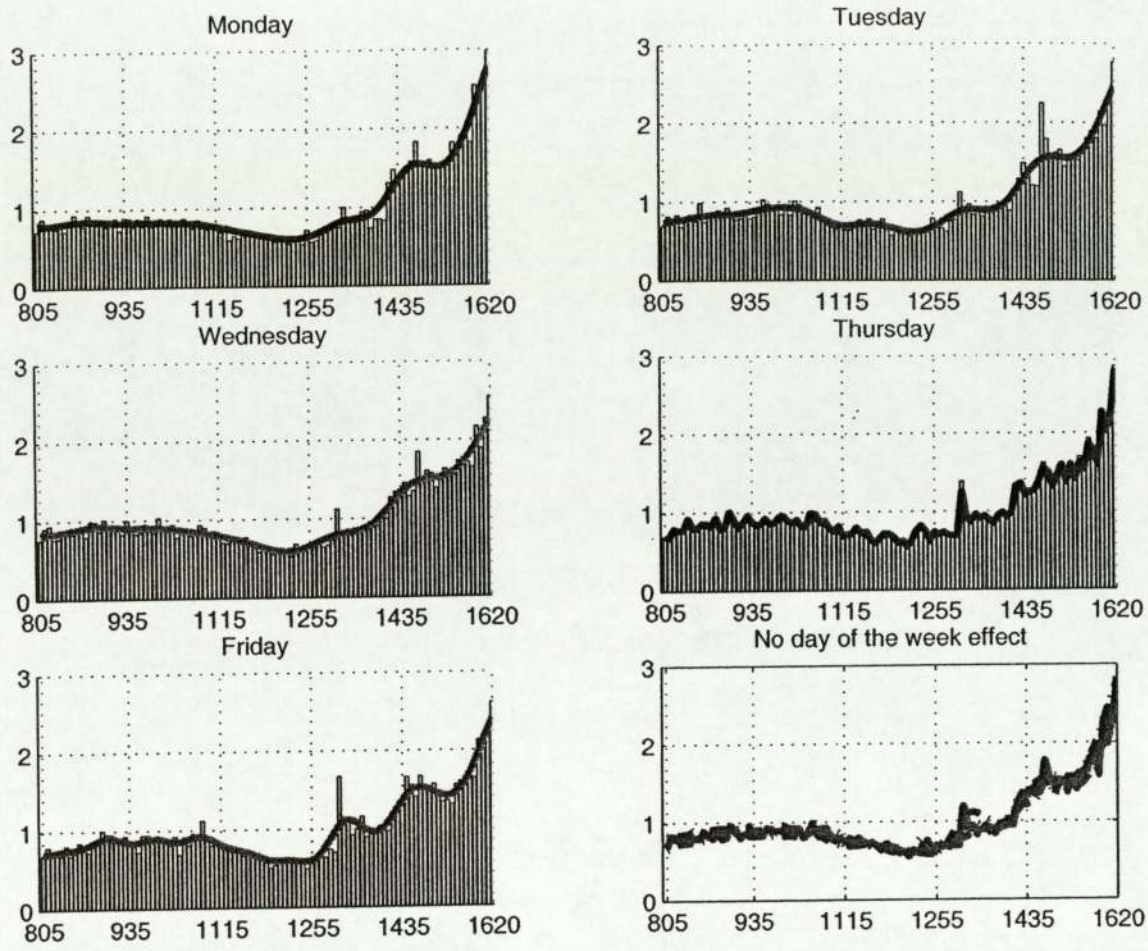


Figure D.11: LVMH time-of-day estimate using Gaussian processes. The histograms represent the whole relative volume dataset.

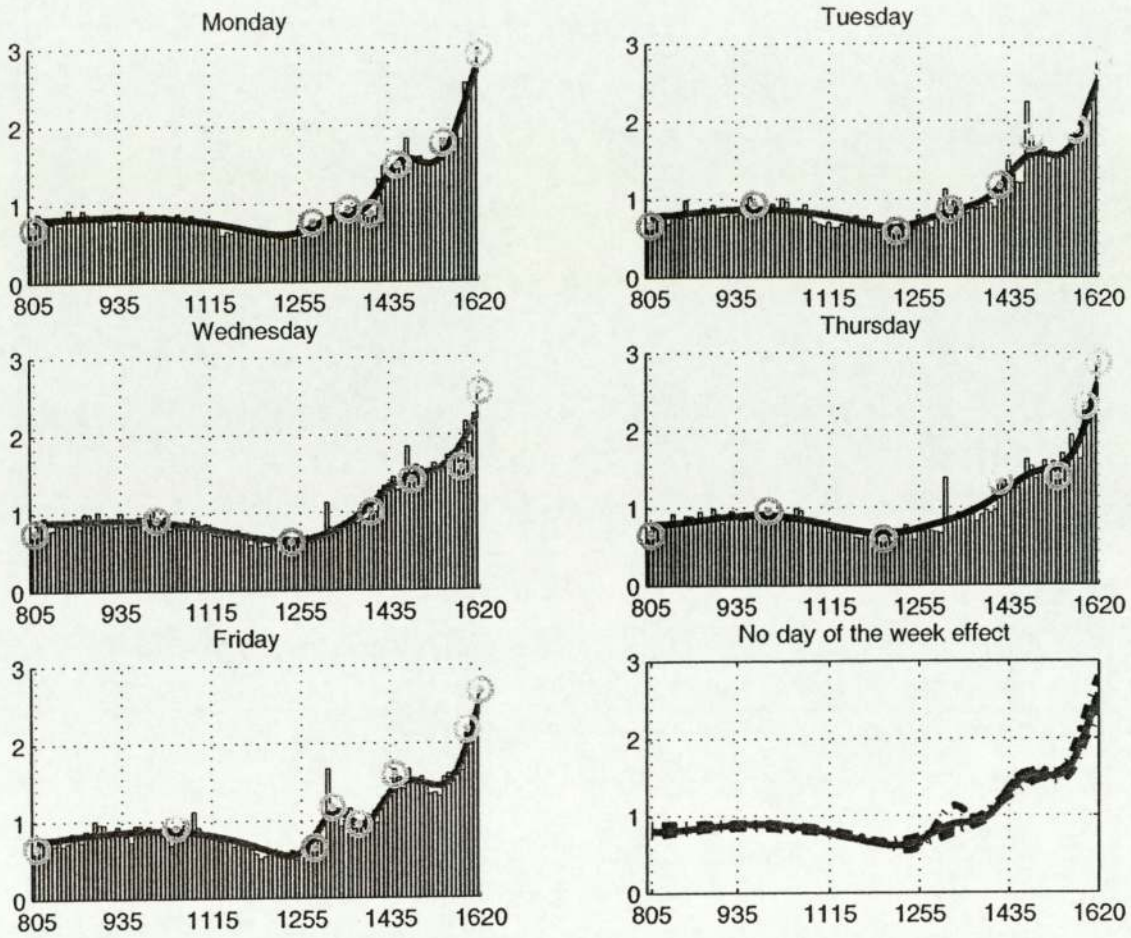


Figure D.12: LVMH time-of-day estimate using relevance vector machine. The histograms represent the whole relative volume dataset. The circled points correspond to the relevance vectors.

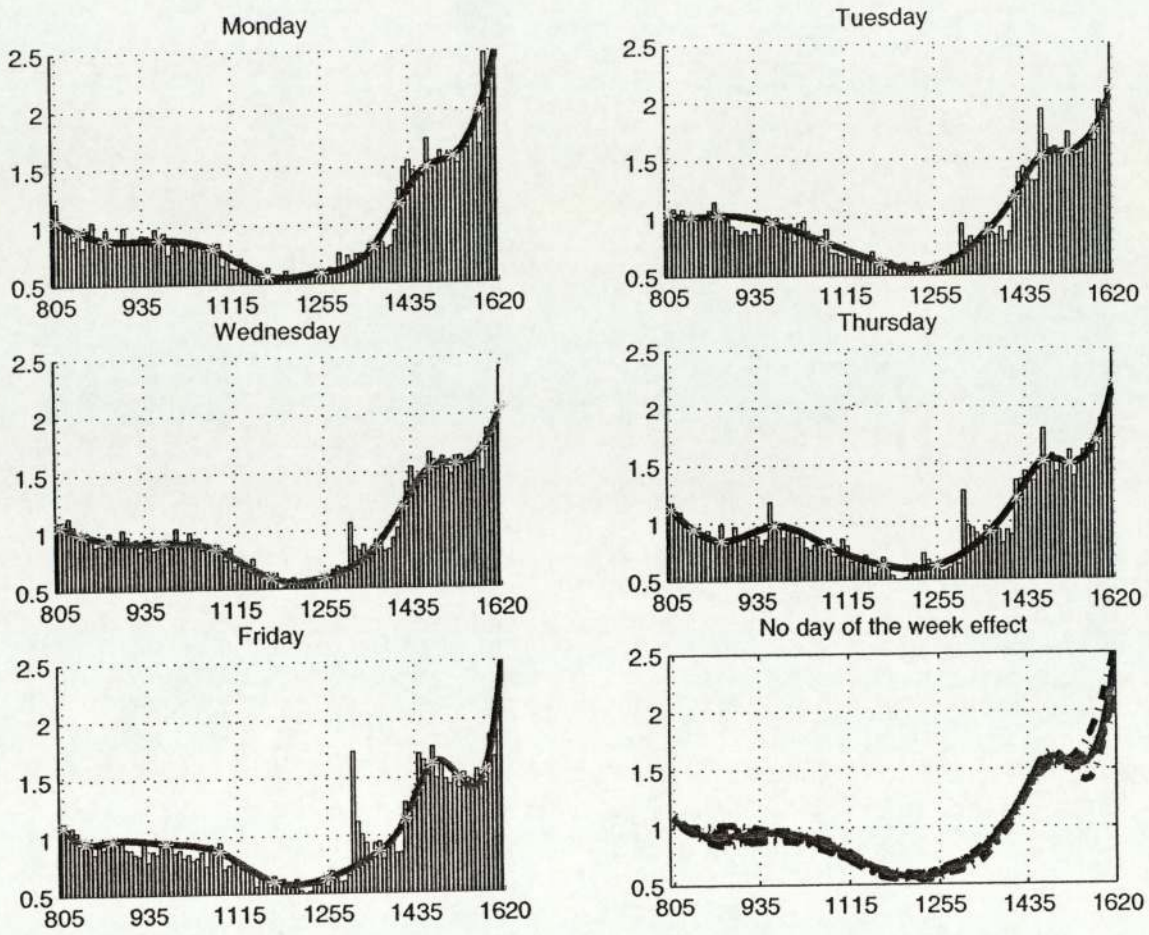


Figure D.13: Vivendi Universal time-of-day estimate using moving average and cubic splines. The histograms represent the whole relative volume dataset.The stars correspond to the knots.