Modeling DAX-Tracking Portfolios with adaptive Beta-Estimators

Rodriguez, Michael [*]
Härdle, Wolfgang

September 28, 2000

Keywords: tracking portfolio, adaptive estimation, local homogeneity

This paper offers a new approach for building index tracking portfolios with a restricted number of stocks. We construct a two steps algorithm that first estimates the correlations between stocks and index in an adaptive way and secondly calculates the optimal weights that minimize the tracking error. Finally the procedure is applied to the DAX-Index and a comparison with a naive model that considers the real weights of each stock in the index is provided. It appears that the procedure succeeds in reducing the tracking error. Introduction of adaptive correlation estimators doesn't improve significantly the tracking error, nevertheless it procures an other distribution of weights that is exploitable on the option market in order to construct volatility arbitrages.

(*) Student involved in the double diploma cursus between the Humboldt Universität zu Berlin and the Ecole Nationale de la Statistique et de l'Administration Economique (ENSAE - Paris). This article is part of the Diplomarbeit which also contains a software application from which all graphs of this article are extracted.

Table of contents

1 Introduction

Since the 90's the market of equity derivatives is a market in big expansion. A lot of products have been emitted with a return indexed on a benchmark like a stock index. On the one hand it raises the problem of hedging such products and on the other hand it creates new topics for the research to arbitrage these products.

So mastering a performing method to reproduce index on a financial market still remains a substantial advantage. Among all the applications we can retain three of them that are important in view of a trading activity.

- Hedge of an index certificate : sure the fastest way is to hedge the fund directly by buying or selling future contracts. But tax conditions and tax goodwill on dividends make it sometimes profitable to intervene on the cash market.

- Base arbitrage between cash and future : if at a given time the empirical base (difference between cash and future) differs from the theoretical base (computed with respect of the interest rates structure) there is a risk-free arbitrage opportunity by buying or selling the future and doing the inverse operation on the cash market.

- Volatility and correlation arbitrage on the option market : derivative products give us an implied volatility structure of the basket and of each stock which constitutes it. Studying it and correlations between stocks and index can result in trading opportunities.

We will not develop more these examples because it is not the aim of this paper. We just want to point out that it exists a great number of applications based on index-tracking portfolios. So developing a performing method is of the most interest in a market in big expansion. Nevertheless the literature is quite poor on this subject. Sure many banks and financial institutions have developed their own methodology but only a few papers have been published. Some banks possess an effective technology and make great performances in basket trading. This observations reveal how the research in this domain is strategic and how many opportunities it remain.

The intuitive way to reproduce an index consists in buying all the stocks which constitute it. This is of course the easiest way. It gives the most faithful reproduction with the smallest tracking error. Nevertheless it is not perfect and presents some inconvenience. First of all the procedure is not executable quickly. The prices may move significantly during the execution time so that a base arbitrage become risky if you just have bought part of the basket. Moreover some stocks are not liquid enough to be bought or sold in a adapted quantity without influencing the price. Finally in order to construct volatility arbitrage the considered stock options must be liquid too.

In this paper we propose, test and compare several other methods to build tracking portfolios. In particular we implement a new algorithm based on an adaptive procedure that identifies breaks in the correlations'structure between stocks and index. Intuitively we optimize the weights of our basket taking by calculing the historical correlations between each stock and the index. But as soon as we do it the question appears of the time period to take into account to calculate the correlations.

A fixed number of business days, say 250, reveals that the historical correlations are not stabil over the time. So we develop an algorithm that detects correlation break.

We limit us in building a tracking portfolio with a restricted number of predetermined stocks, six for the DAX 30. Our methodology refers for the first part to the article of N.F.Wagner (1996) and for the second part relative to the adaptive estimator to a paper of W.Härdle and V.Spokoiny (2000).

The challenge I fixed to myself for the Diplomarbeit is double : develop an efficient algorithm to build tracking portfolios and program a flexible software easily adaptable to market changes and that may be used as trading tool.

The reminder paper is organized as follow. The next section introduces the required modelisation for tracking portfolio, gives some fondamental hypothesis and describes the sampling method used to select stocks. Then we propose a modell with adaptive estimation of the correlations under local time-homogeneity and check it via Monte-Carlo simulation. Finally the procedure is applied to the german stock index (DAX) and the results are presented. The last part of this paper is a user's guide for the software application which is part of the Diplomarbeit.

2 Minimizing tracking error

Let define our goal : build a portfolio that minimize the tracking-error. Some assumptions have to be made to expose our process. Under tracking-error we understand the difference R_E between the return R_P of the build portfolio and the return R_B of the benchmark, that is to say the index we want to track.

$$R_E = R_P - R_B$$

Our aim is to minimize the standard error σ_E of the tracking error R_E. Let now present the fundamental hypothesis and the optimisation program.

2.1 Hypothesis and Beta-Estimation

For each stock *i* we denote by α_i and β_i the coefficients of the linear regression of its returns R_i on R_B and by ε_i the residual normal distributed term.

$$\forall i \in \{1,...,d\} \;\; R_i = \alpha_i + \beta_i R_B + \varepsilon_i$$

Analogically we suppose the tracking portfolio is build and denote by α_P and β_P the linear coefficients.

$$R_P = \alpha_P + \beta_P R_P + \varepsilon_P$$

Now we can write

$$\sigma_E^2 = Var[\alpha_P + (\beta_P - 1)R_B + \varepsilon_P]$$

Furthermore we make the following assumptions :

$$\begin{cases} \forall i \in \{1,...,d\} \;\; E[\varepsilon_i] = 0 \\ \forall i \in \{1,...,d\} \;\; \sigma_i = cst \\ \forall (i,j) \in \{1,...,d\}^2 \quad i \neq j \quad \sigma_{ij} = 0 \\ \forall i \in \{1,...,d\} \;\; \sigma_{iB} = 0 \end{cases}$$

We suppose so that the regression residual terms between two stocks and between the benchmark and each stock are independent. This is an important consideration that should be taken in account in the sampling method we will discuss later. As far as possible we should build the tracking portfolio with stocks that are not correlated. It can appears impossible in a market with trend. Nevertheless we will avoid to put two stocks of the same ecomomical sector in the portfolio.

Finally under the assumption of no correlation, we can decompose the tracking error as follow :

$$\sigma_E^2 = (\beta_P - 1)^2 \sigma_B^2 + \sigma_P^2$$

This equation points out the decomposition of the variance in a systematic part (first term) and a non-systematic part (second term). Non-systematic error results from bad diversification in the tracking portfolio. According to the definition of the tracking portfolio the systematic error should be null with β_P equal to one. As β_P differs from one, the systematic error increases with the benchmark variance. As a matter of fact the tracking error depends on the stability of the betas. This aspect is studied in details in the next parts of this paper. Now we consider the betas have been computed in regressions over 120 and 250 business days.

Once the betas have been calculated we have to determine the optimal weights x_i that minimize the non-systematic error. This error can be written

$$\sigma_P^2 = \sum_{i=1}^{d} x_i^2 \sigma_i^2$$

because of the independence of the stocks residual terms. σ_P is the tracking error we want to minimize.

2.2 Optimal weights calculation procedure

Once the betas have been computed we have so to resolve the following problem

$$Min \quad \sum_{i=1}^{d} x_i^2 \sigma_i^2$$

under the restrictions

$$\begin{cases} \sum_{i=1}^{d} \beta_i x_i = \beta_P \\ \sum_{i=1}^{d} x_i = 1 \end{cases}$$

The restriction $\forall \{1,...,d\} \quad x_i \geq 0$ can be added to impose no short sales. But as we will observe negative weights rarely appear so that we first can ignore this constraint. And as we ignore inequality constraints the problem can be solved with the Lagrange operator. Finally we obtain the solution

$$\forall j \in \{1,...,d\} \quad x_j = \frac{(\alpha_2 - \alpha_1 \beta_P) + (\alpha_0 \beta_P - \alpha_1)\beta_j}{2\sigma_j^2 (\alpha_0 \alpha_2 - \alpha_1^2)}$$

with

$$\begin{cases} \alpha_0 = \sum_{i=1}^{d} \frac{1}{2\sigma_i^2} \\ \alpha_1 = \sum_{i=1}^{d} \frac{\beta_i}{2\sigma_i^2} \\ \alpha_2 = \sum_{i=1}^{d} \frac{\beta_i^2}{2\sigma_i^2} \end{cases}$$

In a first time, the β -coefficients have been estimated as regression coefficients over a fixed period (120 or 250 business days). They so can be calculated as

$$\forall i \in \{1,...,d\} \quad \beta_i = \frac{Cov(R_i, R_B)}{Var(R_B)}$$

where *Cov* and *Var* denote the historical covariance and variance over 120 or 250 business days.

We have defined and resolved the optimisation problem as soon as the correlations have been calculated. Let now describe the sampling method befor getting back to the correlation calculation.

2.3 Sampling Method

Befor computing the weights we have to make a selection between the stocks of the index and to choose a set of them to put in our portfolio. In order to apply the methodology to the german stock index we used the sampling method related in the article of Walter and retain three criteria to select some stocks among the 30 stocks that compose the DAX-index. The selection process is not quantitative but qualitative.

- First of all, because of the hypothesis made in section 2.1, the stocks have to be as far as possible independent. So we will choose stocks from different sectors in order to represent all the activities. By this way we attempt to reduce the correlations.

- Secondly, we try to select the stocks that have the biggest weights in the Index composition. This should intuitively assure a good correlation between the tracking portfolio and the index.

- Finally we must retain stocks whose options are liquid enough to buid arbitrage on the Eurex. This is an important condition in view of further development.

According to these criteria, we retain six stocks for our tracking portfolio. Figure 1 resumes our selection and gives an estimation of the average beta. Let observe that the beta doesn’t necessary get nearer from one as the stock’s weight in the index increases. For example BASF is one of the more correlated stock to the DAX despite of its weight under 3 %. On the contrary Deutsche Telekom has a weight over 20 % and is less correlated.

		Gewichte	Beta250		Beta120	
		23.02.00	Average	STD	Average	STD
MUV	Versicherer	5,28%	1,175040	0,042759	1,153954	0,132566
ALV	Versicherer	8,87%	1,178192	0,045334	1,185679	0,102042
VEB	Energie	2,40%	0,811844	0,077701	0,802504	0,114596
SIE	Technologie	9,80%	0,902450	0,085060	0,903614	0,109051
SCH	Pharma	0,92%	0,745094	0,120946	0,741985	0,141877
LHA	Luftfahrt	0,82%	1,059532	0,121591	1,019483	0,218225
CBK	Bank	2,05%	0,927897	0,123549	0,921196	0,153012
DCX	Auto	6,61%	1,106635	0,130878	1,084941	0,179129
BAY	Chemie	3,20%	0,986073	0,139819	0,984041	0,162802
DTE	Telekom	21,91%	0,943929	0,152176	0,931202	0,185967
DBK	Bank	5,29%	0,946657	0,162349	0,925291	0,201795
DRS	Bank	2,86%	0,976234	0,174848	0,973465	0,196221
BAS	Chemie	2,99%	1,025059	0,184363	1,030510	0,228272
VOW	Auto	1,92%	1,154701	0,186228	1,143979	0,228916
THY	Stahl	1,37%	0,943990	0,207117	0,946342	0,225444
SAP	Software	9,26%	1,284778	0,299991	1,355519	0,463981

Figure 1 – Sectors, weights and average betas for the most liquide stocks in the DAX-Index. In grey, the selected stocks for the tracking portfolio.

If we have a look on the correlations we observe that insurance companies Allianz and Münchener Rückversicherung are the most constant correlated stocks to the DAX with a standard error from 0.45 for the Allianz'beta over 250 days and 0.10 over 120 days. Münchener Rückversicherung has a beta standard error from 0.17 over 250 days and 0,13 over 120 days. Let note that these stocks do not have the biggest weights in the DAX-index.

Finally, according to the criteria developed above, we retain several stocks. In the bank sector we retain Deutsche Bank, in the insurance Allianz, in the technologie Siemens and Deutsche Telekom, in the automobil Volkswagen and in chemical BASF. This total of six stocks represents the 23 th of february 2000 55% of the DAX-index.

We also made many tests with other portfolio selections. Switching some stocks of the portfolio discribed above presents the benefit that we have more historical data to test it. In particular replacing Deutsche Telekom by an other stock permits to dispose about data ut to 1991 since Deutsche Telekom just entered the DAX-index the 19th of September 1996. On the other side, the selected one presents the advantage that it contains the stock Deutsche Telekom which is important the last years in the DAX.

Anyway we let run the algorithms for a lot of stock combinaisons. A lot of time has been invested in developing an application so that the algorithms still remain usable and can be run by everybody in the future. It is also simple to had or change data. If a new stock enter the index you can easily change the time series (see the last part of this paper as user's guide).

For a volatility trading activity, the stock selection should be done with a look on the option market's opportunity. So having a flexible software is indispensable.

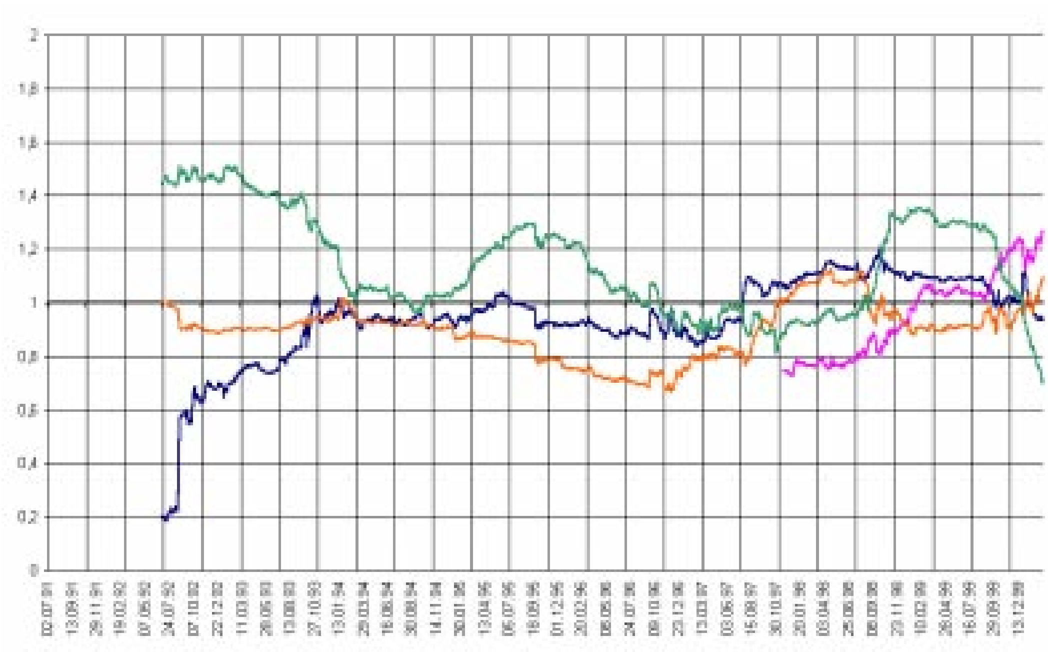
The weights of the stocks is important because we construct another benchmark portfolio to which we systematically compare our tracking portfolio. This benchmark portfolio is made of the same stocks but with different weights. Intuitively we normalize the real weights over 100%. Let ω_i be the real weight of each stock in the index. To each stock *i* of our benchmark portfolio, we attribute the weight φ_i defined as

$$\forall i \in \{1,\ldots,d\} \quad \rho_i = \frac{\omega_i}{\sum_{k=1}^{d} \omega_k}$$

Our aim is now to minimize the tracking error and to obtain an error smaller as these of the benchmark portfolio.

3 Adaptive estimation of the correlations under local time-homogeneity

Up to now we have considered that the betas remains constant overall the fixed regression period, say 250 or 120 business days. But as an evidence we can see that it is not the fact. Figure 2 and Figure 3 shows how the betas can fluctuate over the



time.

Figure 2 - Evolution of the betas computed over the 250 past business days for the stocks Deutsche Bank (blue), Deutsche Telekom (rose) Siemens (orange) and Volkswagen (green) from July 1992 to February 2000.

Clearly the betas are not constant. Reducing the regression period from 250 business days to 120 business days we can observe that the betas fluctuate more.

Figure 3 - Evolution of the betas computed over the 120 past business days for the stocks Deutsche Bank (blue), Deutsche Telekom (rose) Siemens (orange) and Volkswagen (green) from July 1992 to February 2000.

We now propose an other approach focusing on a very simple model but with a possibility for model parameters to depends on time. This means that the model is regularly checked and adapted to the data. No assumption is made about the parametric structure of the regression process, we only suppose that it can be locally approximated by constants, that is, for every time point τ ther exist a past interval $[\tau - m, \tau]$ where the regression parameter β do not vary much. This interval is referred to as interval of time homogeneity. An algotithm is proposed for data-driven estimation of the interval of time homogeneity, after which the estimate of β can be obtained as regression coefficient.

3.1 Description and theoretical properties

We denote by $\tilde{\beta}_I$ the estimated parameter β on a properly selected time interval of the form $[\tau - m, \tau]$ to minimize the corresponding regression estimation error ε_i. We discuss one approach which goes back to the idea of pointwise adaptive estimation, see Lepski (1990), Lepski & Spokoiny (1997) and Spokoiny (1998). The idea of the method can be explained as follows.

Suppose *I* is an interval-candidate, that is, we expect time-homogeneity in *I* and hence, in every subinterval of *I*. This particularity implies that the value Δ_I is negligible and similarly for all Δ_J, $J \subset I$ qnd that the mean value of the β over *I* and over *J* coincide. Our adaptive procedure roughly means a familiy of tests to check whether $\tilde{\beta}_I$ does not differ significantly from $\tilde{\beta}_J$ for any subinterval *J* of *I*.

Now we propose a formal description. Suppose a family $\mathcal{I}$ of interval-candidates *I* is fixed. Each of them is of the form $I = [\tau - m, \tau]$, so that the set $\mathcal{I}$ is ordered due to

m. With every such interval we associate the estimate $\tilde{\beta}_I$ of the parameter β_τ and the corresponding estimate $\tilde{\varepsilon}_I$ of the conditional standard deviations ε_I.

Next, for every interval I from $\mathcal{I}$, we suppose to be given a set $\Im(I)$ of testing subintervals J. For every $J \in \Im(I)$, we contruct the corresponding estimate $\tilde{\beta}_J$ from the observations R_t for $t \in J$ and compute $\tilde{\varepsilon}_J$.

Now, with two constant λ and μ, we define the adaptive choice of the interval of homogeneity by the following iterative procedure :

Initialization : Select the smallest interval in $\mathcal{I}$.

Iteration : Select the next interval I in $\mathcal{I}$ and calculate the corresponding estimate $\tilde{\beta}_I$ and the estimated conditional standard deviation $\tilde{\varepsilon}_I$.

Testing homogeneity : Reject I, if there exists one $J \in \Im(I)$ such that

$$\left|\tilde{\beta}_I - \tilde{\beta}_J\right| > \lambda \cdot \tilde{\varepsilon}_J + \mu \cdot \tilde{\varepsilon}_I$$

Loop : If I is not rejected, then continue with the iteration step by choosing a larger interval. Otherwise, set $\hat{I}$ as the latest non rejected I.

The adaptive estimate $\hat{\theta}_\tau$ of $\tilde{\theta}_{\hat{I}}$ is defined by applying this selected interval $\hat{I}$:

$$\hat{\theta}_\tau = \tilde{\theta}_{\hat{I}}$$

It is supposed that the procedure is independently carried out at each time point τ. A possibility to reduce the computational effort of the selection rule is to make an adaptive choice of the intewrval of homogeneity only for some specific time points t_k and to keep the left end-point of the latest selected interval for all τ between two neighbor points t_k and t_{k+1}.

The presented algorithm involves the sets $\mathcal{I}$ and $\Im(I)$ of considered intervals and two numeric parameters λ and μ. We now discuss how these parameters can be selected starting from the set of intervals $\mathcal{I}$. The simplest proposal is to introduce a regular grid $G = \{t_k\}$ with $t_k = m_0 \cdot k$ for some natural number m_0 and to consider the intervals $I_k = [t_k, \tau[$ for all $t_k < \tau$. It is also reasonable to carry out the adaptive procedure only for points τ from the same grid G. The value m_0 can be selected, e.g., between 5 and 30.

If $\tau = t_{k^*}$ for some $k^* \geq 1$, then clearly every interval $I = [t_k, \tau[$ contains exactly k^*-k smaller intervals $I' = [t_{k'}, \tau[$ for all $k < k < k^*$. Next, for every such interval $I = [t_k, \tau[$, we define the set $\Im(I)$ of testing intervals J by taking all smaller intervals $I' = [t_{k'}, \tau[$ with the right end-point τ and similarly all smaller intervals $[t_k, t_{k'}[$ with the left end-point t_k, $k < k' < k^*$:

$$\Im(I_k) = \{J = [t_{k'}, \tau[\; or \; J = [t_k, t_{k'}[: k < k' < k^*\}$$

Let N_I stand for the number of subintervals J in $\Im(I)$. Clearly, for $I = [t_k, t_{k^*}[$, the set $\Im(I)$ contains at most $2(k^* - k)$ elements, that is, $N_I \leq 2(k^* - k)$.

3.2 Adjusting λ and μ

The behaviour of the procedure critically depends on the parameters λ and μ. The simulation results in the next section incate that there is no universal optimal choice. So we calculate the optimal λ and μ by iterating the optimisation process described above many times an retain for λ and μ the value that minimize the tracking error.

Nevertheless inspired from the Chow test, we propose a second algorithm that makes abstraction from λ and μ. For every interval $I_k = [t_k, \tau[$ candidate to homogeneity, we apply the Chow test for every subdivision of I_k in two parts :

$$\wp(I_k) = \{\{[t_k, t_{k'}[, [t_{k'}, \tau[\} : k < k' < k^*\}$$

We denote by SSE_0 the sum of squared errors for the global regression over the interval candidate I_k,, by SSE_1 the sum of squared errors for the regression over the subinterval $[t_k, t_{k'}[$ and by SSE_2 the sum of squared errors for the regression over the subinterval $[t_{k'}, \tau[$. According to the Chow test, we have asymptotically

$$F = \frac{\dfrac{SSE_0 - SSE_1 - SSE_2}{K_0}}{\dfrac{SSE_1 + SSE_2}{K_1 + K_2 - 2K_0}} \xrightarrow{d} F(K_0, K_1 + K_2 - 2K_0)$$

Finally the candidate I_k is accepted if the Chows test hypothesis of no break point is not rejected for every partition of $\wp(I_k)$. On the contrary to the adaptive algorithm, we begin here with a big interval and reduce it until the Chow test confirms homogeneity.

This algorithm presents the advantage to have a statistical test. Neverthless the more the interval becomes small, the less it can be applied because of the asymptotic property. On the other hand, the adaptive algorihm can be optimize over all the data as explained above.

Finally we systematically compare four tracking portfolio :

- The portfolio with the real weights of the stocks extrapolated over 100 %.
- The portfolio obtained by computing the betas over a fixed number of days.
- The portfolio where local time homogeneity is tested via the Chow test.

- The portfolio where local time homogeneity is detected via adaptive estimator.

Before generating tracking portfolios, let evaluate the quality of the beta-estimations between the different methodoligies and discuss it with a Monte-Carlo simulation.

3.3 Monte-Carlo simulation

The aim of this section is to illustrate the performance of the proposed procedure on some simulated examples and to give some hints concerning the choice of the parameters λ, μ and m_0.

Given a random generated time-serie from 1991 to 2000 we generate a correlated serie with two chosen β and two chosen standard deviation σ for the residuals. The first couple (β_1, σ_1) is used for the first half period 1991-1995 and the second couple (β_2, σ_2) for the second period 1996-2000. We then apply the different algorithms to calculate β and compare the estimations.

As an evidence Figure 4 shows that the adaptive procedure faster detects the structure break in the data.

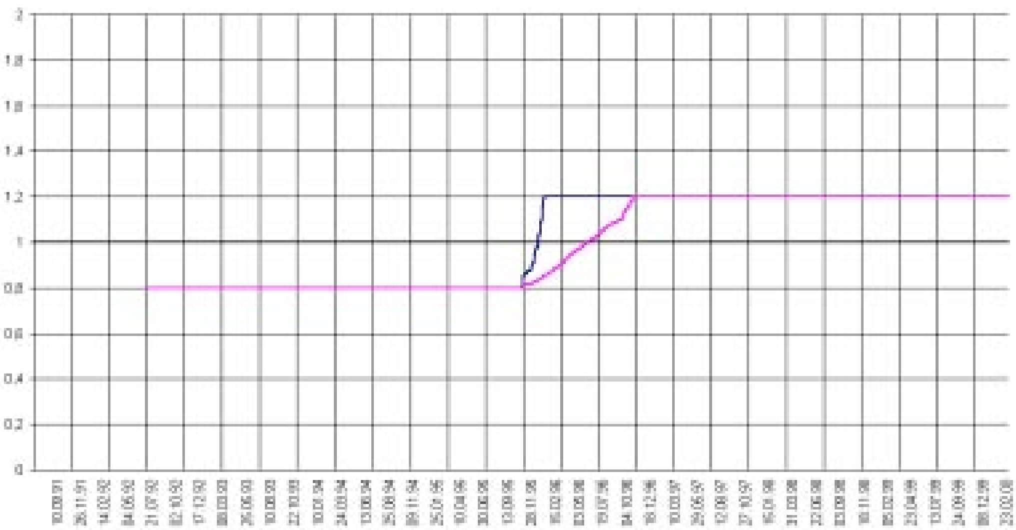


Figure 4 – Monte Carlo simulation : adaptive beta (blue) and normal beta (rose) for two correlated time series with a break at a date point.

The graphic of the Figure 4 is obtained with $m_0 = 50$. Further experiences show that the smaller m_0 , the quicker the break is detected. Nevertheless if m_0 becomes too small comparatively to the volatility parameters σ_1 and σ_2 , the procedure commits mistakes and detects wrong breaks. There is so an equilibrium to be found between m_0 and the volatility parameters. Empirically volatility parameters are intrinsic to the data so that m_0 has to be adapted in order not to fall in a trap.

Figure 5 shows how the historical period used for the regression evoluate over the time. Typically we can see that when the correlation is constant the biggest period M

is used. When a break is detected, it sudden becomes small and as the break is confirmed by the passage of time the period becomes bigger again until it reaches the maximal authorised size *M*.

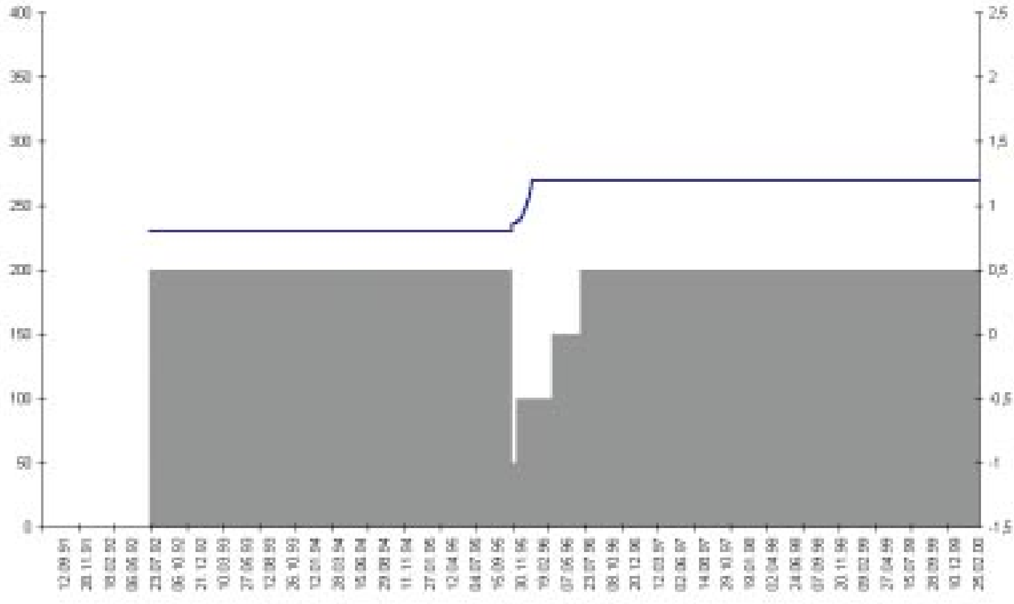


Figure 5 – Monte-Carlo simulation : adaptive beta calculation (blue line) and adaptive depth used to compute beta (grey histogram).

The two other parameters to be precised are λ and μ. Through many experiences we can see that they impact considerably the results estimation. If the parameters are to high, the procedure never detects breaks. If the parameters are to small, the parameters detects a lot of insignificant breaks. On the contrary to m_0 , these parameters do not depend on the volatility of the time-series because these ones are automatically taken into account at each step of the algorithm. Simulations also show that the results seem to be better as λ and μ are proportional.

In order to optimize the procedure we minimize the tracking error by iterating the algorithm for several values of λ and μ. The tracking error is view as a function of λ and μ and minimized.

$$\left(\hat{\lambda}, \hat{\mu}\right) = Arg \min \ \hat{\sigma}_E(\lambda, \mu)$$

Monte-Carlo simulations reveal that the minimal tracking error is obtained for λ and μ next to 2 and 3. Nevertheless if we apply this optimization process for λ and μ to historical data, it appears that the optimal value depends a lot on the stock. This subject is approached in the next section.

4 Empirical Evidences

In this part we make an empirical experience the procedures we proposed above. We compute and test every algorithm proposed on the german stock market. We so consider the german stock index Dax and try to track it some selected stocks. For the selection of the stocks see section 2.3.

4.1 Data description and market overview

We dispose from 18 time-series, the DAX-index and 17 stocks that compose it. The data ara corrected from external events like split. Nevertheless they are not corrected from dividend payments. This is a source of biais because the DAX is an index where dividends are reinvested.

Time-series begin the 1st of July 1991 and end the 19th of May 2000. As explicated in part 5.1 of this paper the data are easily updated.

Nevertheless let note that some stock time-serie begin later because the stock entered later in the Dax-Index. This is the fall for Deutsche Telekom that entered the Dax-index the 19th of November 1996, of Münchener Rückverischerung whose time-serie begin the 23rd of September 1996 and of SAP (the 28th of september 1995).

The time-series are daily. Nevertheless it could be interesting to apply the procedures we have developped in this paper to tick data. This remains a topic for further research.

Actually we are working with returns time-series that are calculated as log-returns from the original data.

4.2 Forecasting the correlations

We now test empirically the procedures we have elaborated to estimate the correlations between the Dax-index and the main stocks that compose it. In order to do this we use the software applition we have programmed and we discribed in section 5 of this paper.

The performance of the different parameter sets are compared for two different criteria : averaged quadratic risk (MSE) and average absolute deviation risk (MAE), and also on their empirical counterparts based on forecast error (FE) : mean squared forecast error (MSFE) and mean absolute forecast error (MAFE) :

$$MSE = \frac{1}{T - t_0 - 1} \sum_{t=t_0}^{T} \left(\hat{\theta}_t - \theta_t\right)^2$$
$$MAE = \frac{1}{T - t_0 - 1} \sum_{t=t_0}^{T} \left|\hat{\theta}_t - \theta_t\right|$$
$$MSFE = \frac{1}{T - t_0 - 1} \sum_{t=t_0}^{T} \left(\hat{\theta}_t - Y_t\right)^2$$
$$MAFE = \frac{1}{T - t_0 - 1} \sum_{t=t_0}^{T} \left|\hat{\vartheta}_t - Y_t\right|$$

We can observe that the betas are not stabil over the time. The bigger beta is, the more the volatility of the stock is higher as this of the index. So when the blue line (adaptive beta) jumps over the rose (constant regression beta) one we can anticipate that the stock volatility is becoming higher. On the contrary, when the blue line goes under the rose one, the volatility of the stock is becoming smaller. To check it visually we have represented on the same graph (see Figure 6 and 8) in orange the difference of the returns between the stock and the index. Figure 6 shows the evolution of the adaptive beta for the stock Allianz and Figure 8 for the stock Siemens. On the graphics we can effectively see that the blue line jumps over the rose one when stock becomes relatively more volatile as the index.

Another important consideration is to see how the interval length used for the regressions evoluates over the time. In Figure 7 and Figure 9 we can see that it fluctuate a lot. It quickly becomes smaller as the algorithm detects a correlation break. Many experiences we made show us that interval length fluctuation is extremely correlated to the parameters lambda and mu.

The figures exposed in this paper are these obtained for lambda and mu optimized for each stock. Figure 10 shows the optimal lambda and mu computed for each stock. It also shows the interval average length and its standard deviation that results from the optimalisation. It is remarquable to notice how the optimal parameters differ between the stocks. Mu seems to be more stable as lambda. Lambda is actually greater for technology stocks. The interval length and its standard deviation also fluctuate a lot.

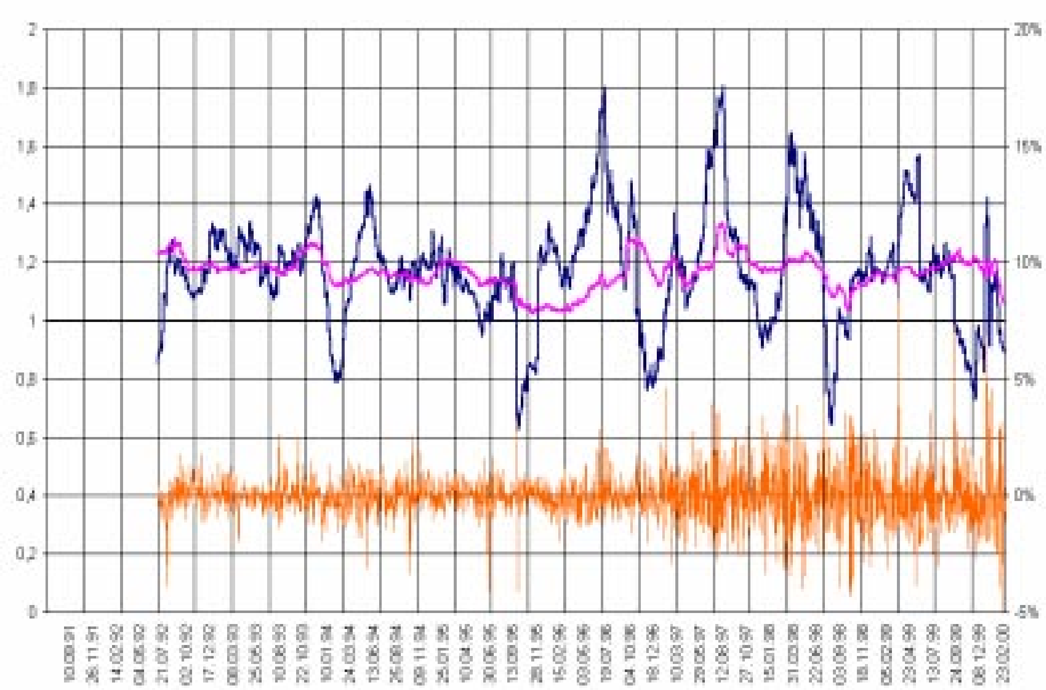


Figure 6 – Normal beta over 250 days (rose) and adaptive beta (blue) for the stock Allianz computed from 1992 to 2000. In orange the ralative variation of the stock in comparison with the index (difference of the two returns).

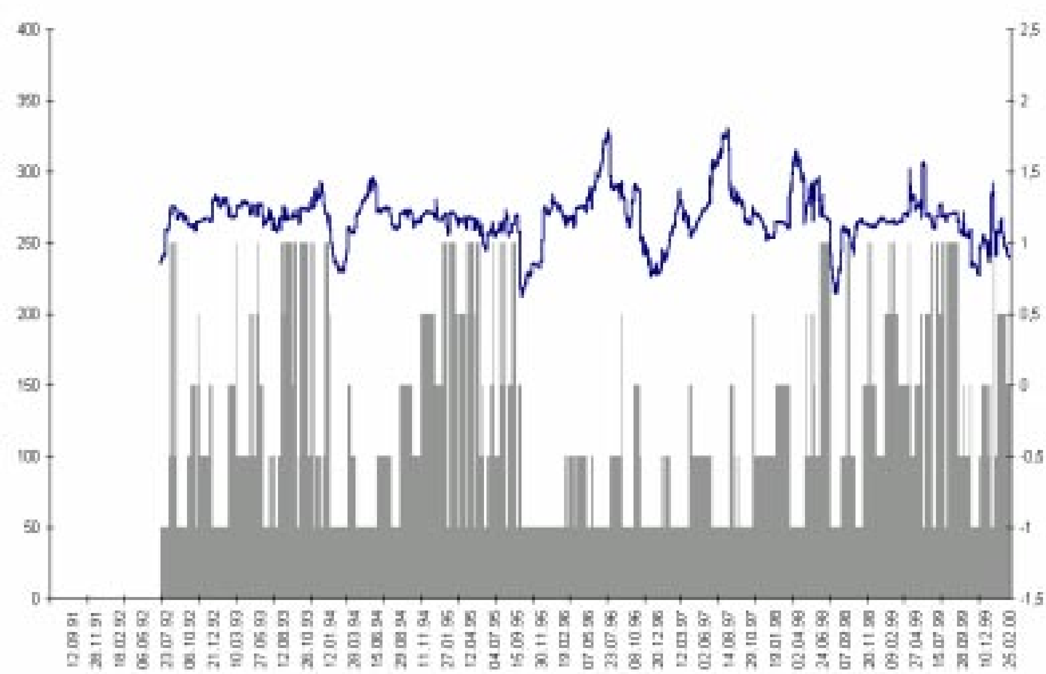


Figure 7 – Adaptive beta estimation (blue line) and adaptive deph for the regression calculation of the betas (grey histogram) for the stock Allianz.

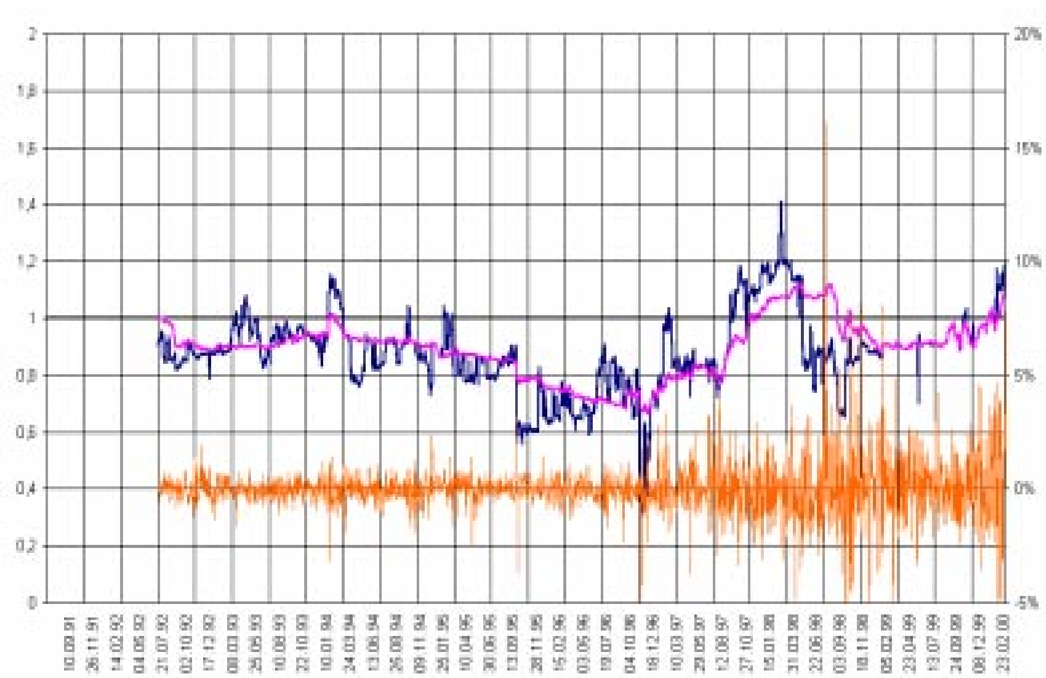


Figure 8 - Normal beta over 250 days (rose) and adaptive beta (blue) for the stock Siemens computed from 1992 to 2000. In orange the relative variation of the stock in comparison to the index (difference of the two returns) .

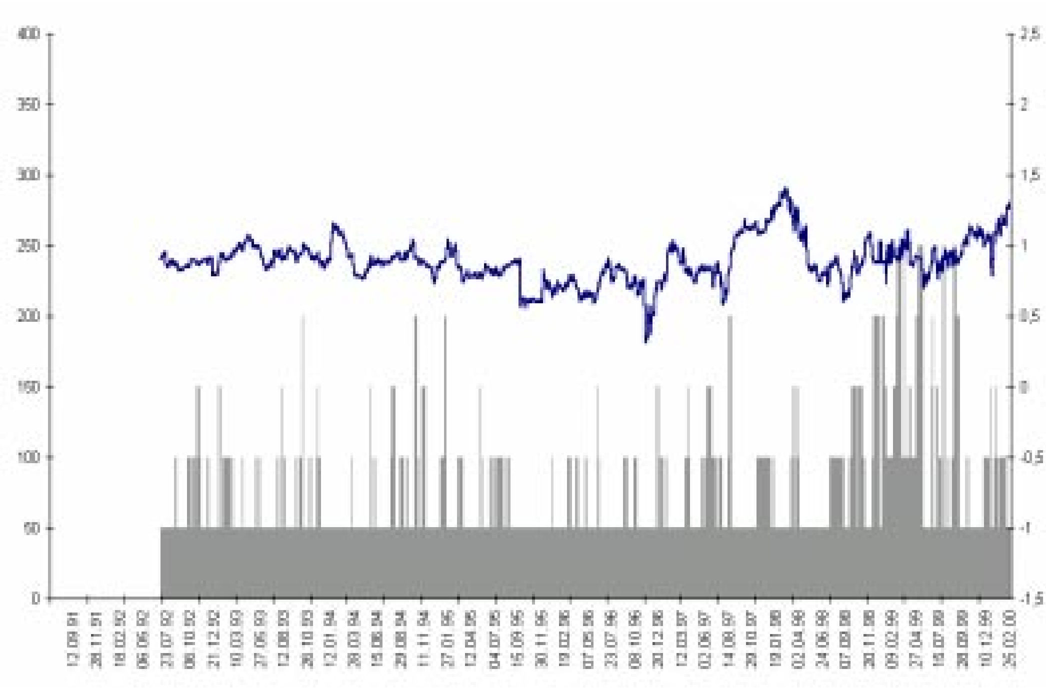


Figure 9 - Adaptive beta estimation (blue line) and adaptive deph for the regression calculation of the betas (grey histogram) for the stock Siemens.

	Lambda	Mu	Mean interval length	Interval length standard deviation
ALV	3,00	2,00	147	49
BAS	4,75	3,50	86	75
BAY	5,25	3,25	103	49
CBK	5,25	4,00	112	53
DCX	6,75	4,57	98	46
DBK	5,50	4,00	166	45
DRS	2,75	2,00	152	54
DTE	7,25	3,50	83	35
LHA	4,25	3,25	123	53
MUV	3,50	2,75	153	61
SAP	6,00	3,25	65	26
SCH	4,25	3,75	101	48
SIE	5,50	3,25	74	36
THY	4,25	3,50	131	58
VEB	3,50	3,00	110	62
VOW	2,75	2,50	175	41

Figure 10 – Optimalisation results for lambda and mu and overview on the mean and the standard deviation of the adaptive interval length

Befor studying tracking portfolio let imagine an intuitive application of adapative beta estimation on the option market. An intuitive and simple trading system would be : be long stock option - short index option when the adaptive beta goes above the normal beta, be short stock option - long index option when the adaptive beta goes under the normal beta. It should result in a positif P&L because of the gamma and vega valorisation.

We checked this simple dispersion strategy under constant volatility and it appears that it would have had an average return of 23 % per year from 1992 to 2000 for Allianz and 15 % per year for Siemens. Althought we made a lot of assumptions (Black & Scholes model with constant dividend yield and constant interest rates), this example points out a possible application of the beta adaptive estimator on the option market. Naturally the strikes of the options have to be optimize in consideration of the smile. This subject is not the aim of our paper. It was just to have a look on the further possible and intersting development. So let come back to the tracking portfolio.

4.3 Tracking portfolio results estimation

This is the final part of the study. We simulate and test tracking portfolios according to the methodologies we have developped above. We systematically compute and test four tracking portfolios :

- Bench : The portfolio with the real weights of the stocks extrapolated over 100%.
- Normal : The portfolio obtained by computing the betas over a fixed number of days.
- Chow : The portfolio where local time homogeneity is tested via the Chow test.
- Spok : The portfolio where local time homogeneity is detected via adaptive estimator.

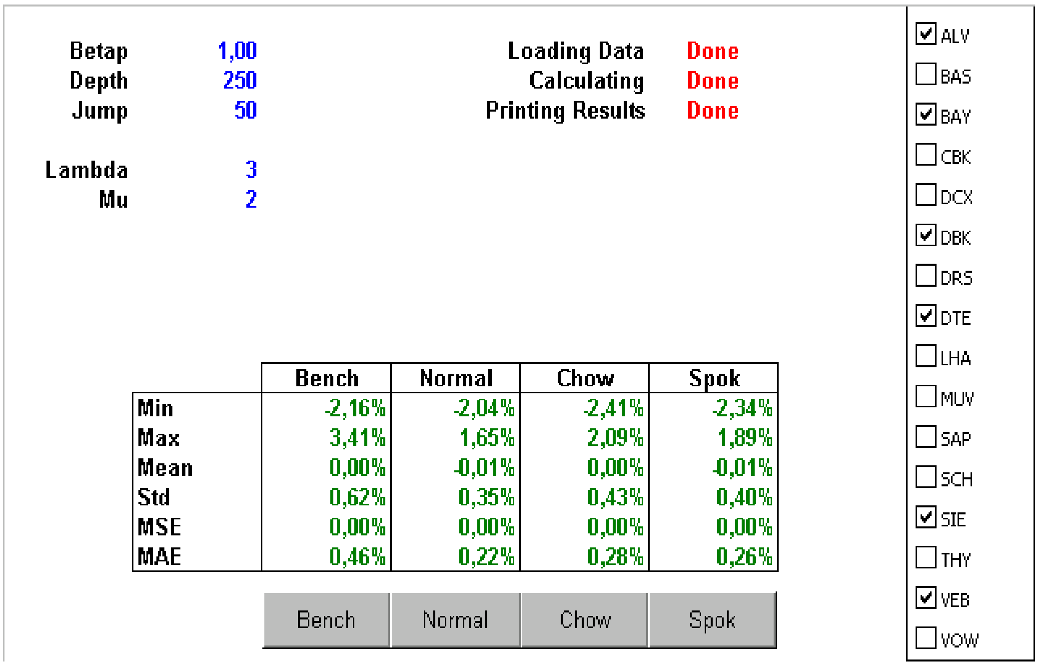
The empirical experiences show that the three last algorithms succeeds in reducing the tracking error comparatively to the benchmark portfolio. Whatever the selected stocks, we obtain a better result. In term of Mean Absolute Error the tracking error is often divided by two. The figures 12 to 15 show the best results obtained. One portfolio contains the six stocks Allianz, Bayer, Deutsche Bank, Deutsche Telekom, Siemens and Veba. In the other portfolio Deutsche Telekom is substituted by Volkswagen. Actually the aim in the second portfolio was to select stocks that have a longer time-serie as Deutsche Telekom so that we could check the model with more data.

Concretely we let run the algorithms with all the possible conbinaisons of six stocks and it appears that the portfolios in Figure 12 to 15 have the smallest average absolute tracking error.

The portfolio with Deutsche Telekon has an average absolute tracking error of 0,22 % with the normal method, 0,28 % with the Chow test and 0, 26% with the adaptive procedure. The error is always smaller 0,46 %, the error obtained with the benchmark procedure. If we swap Deutsche Telekom by Volkswagen the error are respectively 0,22 %, 0,26 %, 0,25% and 0,33%.

						Bench	Normal	Chow	Spok
ALV	BAY	DBK	DTE	SIE	VEB	0,48%	0,22%	0,28%	0,26%
ALV	BAY	DBK	SIE	VEB	VOW	0,33%	0,22%	0,26%	0,25%
ALV	BAY	DRB	DTE	SIE	VEB	0,47%	0,24%	0,30%	0,30%
ALV	BAS	DBK	DTE	SIE	VEB	0,48%	0,25%	0,30%	0,29%
ALV	BAS	CBK	DTE	MUV	VOW	0,35%	0,23%	0,29%	0,28%
BAS	DCX	DRS	DTE	SAP	VOW	0,40%	0,25%	0,28%	0,25%
BAS	DCX	DRS	LHA	SAP	VOW	0,42%	0,25%	0,28%	0,27%
BAY	DCX	DBK	LHA	SIE	SCH	0,45%	0,23%	0,27%	0,26%
BAY	DCX	CBK	LHA	MUV	THY	0,40%	0,30%	0,29%	0,29%
CBK	DBK	DTE	MUV	SCH	VOW	0,43%	0,29%	0,28%	0,27%

Figure 11 – Average absolute error for several good tracking portfolios



	Bench	Normal	Chow	Spok
Min	-2,16%	-2,04%	-2,41%	-2,34%
Max	3,41%	1,65%	2,09%	1,89%
Mean	0,00%	-0,01%	0,00%	-0,01%
Std	0,62%	0,35%	0,43%	0,40%
MSE	0,00%	0,00%	0,00%	0,00%
MAE	0,46%	0,22%	0,28%	0,26%

Figure 12 – Results estimation of the tracking portfolio build with the stocks Allianz, Bayer, Deutsche Bank, Deutsche Telekom, Siemens and Veba.

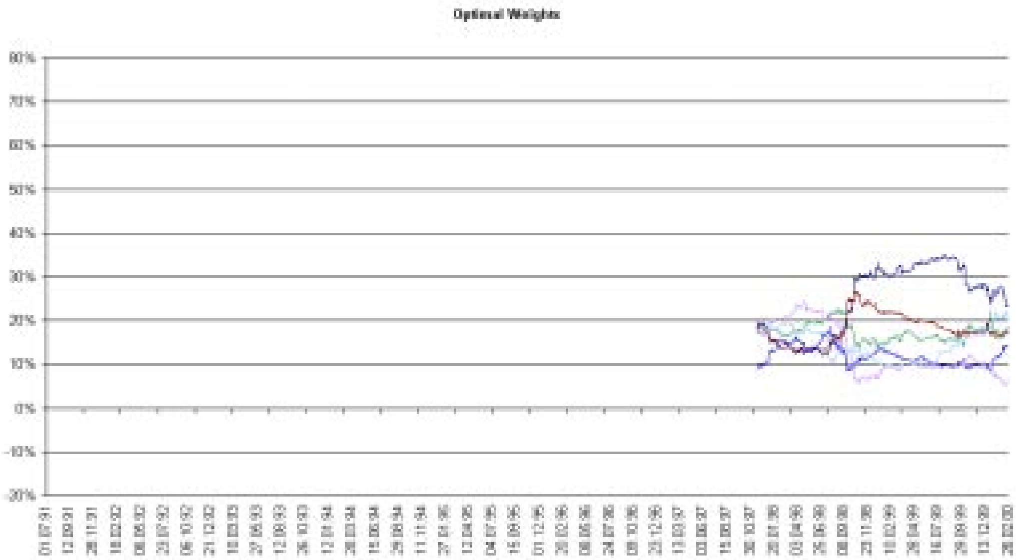


Figure 13 – Allianz (dark blue), Bayer (green), Deutsche Bank (brown), Deutsche Telekom (blue), Siemens (light blue) and Veba (violet)

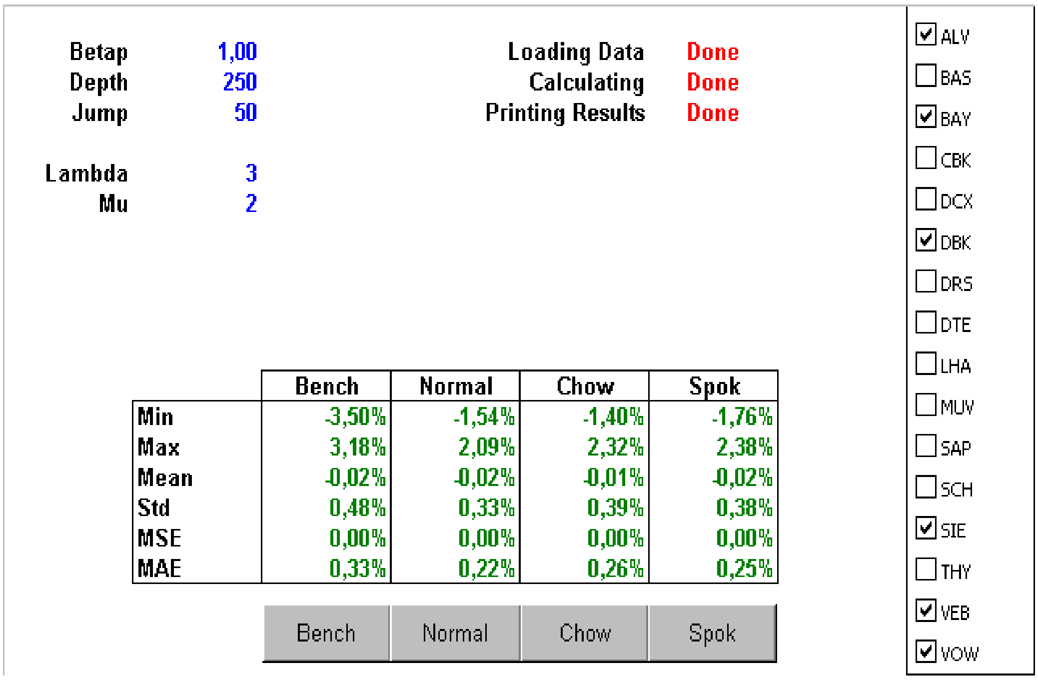


Figure 14 – Results estimation of the tracking portfolio build with stocks Allianz, Bayer, Deutsche Bank, Siemens, Veba, Volkswagen.

Figure 15 - Allianz (dark blue), Bayer (green), Deutsche Bank (brown), Siemens(light blue), Veba (violet) and Volkswagen (yellow)

As shown in Figure 11 the average tracking absolute error for the adaptive procedure seems always to be between 0,20 % and 0,30 % whereas for the benchmark it is between 0,30 % and 0,50%.

Let now compare the fluctuation of the weights for each procedure. In order to compare the evolution of the stocks' weights for every algorithm, Figure 16 gives the real weights normalized over 100 % and used for the benchmark portfolio.

Figure 16 – Real weights of the stocks Allianz (dark blue), Bayer (green), Deutsche Bank (brown), Siemens (light blue), Veba (violet) and Volkswagen(orange).

We can see the evolution of the weights for each stock. We observe that the weights fluctuate more in the adaptive procedures than in the benchmark. It should generate more transaction costs in the praxis. Calculating the transaction's costs it would be interesting in a further study to analyse if the money we can gain by reducing the tracking error is greater than the money we may lose due to the higher transaction costs.

As Deutsche Telekom just entered the DAX the 19th of November 1996, we build other portofolios without Deutsche Telekom to check the algorithms over more years. Replacing Deutsche Telekom by Volkswagen conduces to the best portfolio and is in agreement with the samplig method discribed in section 2.3.

Figure 14 and Figure 15 show the results. The performance still remains good but the stocks fluctuate more.

Neverthless as we can see in Figure 12 and 14, the introduction of adaptive algorithms does not improve more the results as the simple normal regression over a fixed period of days. Actually the tracking error is approximatively the same with a normal regression over a fixed number of days and with an adaptive process like these implemented. It could so first appear useless to compute adaptive algorithms. However the study succeeds in determining many distributions of weights that conduce to a better tracking error as the benchmark.

It is now important to understand why the adaptive procedure does not really improve the results. Actually it is the fact that it seems to be too sensible. Sure we can increase lambda and mu to reduce the sensibility of the model. As lambda and mu become bigger, the model converges to the normal model without adaptive parameters. At each time point the regression period is equal to the maximum length M and the final result is the same as this obtained with the normal model.

In fact we should start from the constatation that if we consider the estimation of beta separately as in part 4.2 the adaptive algorithm succeeds in obtaining a better result. The procedure is so efficient. But as soon as we introduce it in the optimization problem of tracking portfolios it doesn't inprove the global tracking result. So it looks like as if the minimization were not so dependent on the betas computed befor. The minimization program could find a good result whatever the vector of betas. The betas could fluctuate a lot it's finally the minimization program that adapts its arguments (the weights) to obtain a small tracking error.

So getting on the option market we can ask the following question : given a correlation structure between stocks and index, what is the optimal weights to build a tracking portfolio. And the optimization program we propose gives the answer. Now it remains the question of determining the correlation structure and the adaptive algorithm gives an answer conditionnally to some parameters.

Finally we are able to propose several weights' distributions of stocks to track an index. The tracking error has been divided approximatively by two on comparison with the model we take as benchmark. It would be to my mind interesting to apply this algorithm to tick date and to check if it is possible to build arbitrage between the cash and the future market. Such a tool that can track an index with only a few stocks should be performant.

5 Software application

This last part of the paper is a user's guide to the programs which belong to the Diplomarbeit. It is divided in three parts. The first one is an introduction to the principes of the programmation's methodology we apply. The two last parts explain how to use the two Excel files. These parts are usefull for a trader who wants to applicate the algorithms on an index. They explain how to import data and how to compute the algorithms. Excel files can be considered as trading tools.

5.1 Principes and conventions

As above evoked, we have elaborated a soft that makes it always possible to compute the algorithms in the future. We build a Dynamik Link Libraries (DLL) which contains two objects used by an Excel@ – Visual Basic@ application They can also be used with XploRE@ or an other program. The two objects programmed in C++ and whose code figures in annexe of the Diplomarbeit are named SimpleReg and TrackMachine. We also have programmed a Visual Basic interface that make it possible to use the two objects in Excel@.

- SimpleReg is the object used in the spreadsheet BetaStudie.xls to compute and test the betas. It contains various algorithms to calculate beta : the regression modell over a fixed period, the Chow test procedure and the adaptive procedure discribed above. Its functioning is explained in part 5.2.

- TrackMachine is the object used in the spreadsheet Finalsoft.xls to build and test tracking portfolios. It permits to buid portfolios using various algorithms : the benchmark modell, the modell with beta calculated over fixed periods, beta stability tested with the Chow test and adaptive procedure for calculating beta. Its functioning is explained in part 5.3.

These objects belong to the Diplomarbeit. They have been programmed in C++. Actually the code is divided in five objects. Although two objects ara useable in the DLL, the C++ code consists in five objects. The five objects are quickly described above to help the reader that would like to pursue the research.

- **SLRM** : This object permits to compute a Simple Linear Regression Modell. It calculates the vector of coefficients for a Linear Regression.

- **SimpleReg** : This object is used to test and study the stability of beta. It's an object that can be used from Excel, Visual Basic or XploRe.

- **Tracker** : This object permits to calculate the optimal weights according to the Wagner Test and the adaptive procedure of V.Spokoiny and W.Härdle.

- **TrackingPortfolio** : This object permits to generate and test all the tracking portfolios related in the paper. It uses all the others objects and constitutes the kernel of the soft

- **TrackMachine** : This object is an interface used to make it posssible for Excel, Visual Basic or XploRe to use the object TrackingPortfolio.

The code of the objects figures in the annexe and in the disk that accompanies this paper. For each object there is a header and a program file.

For the Excel files we take some conventions. We put in blue the input parameters and in green the output parameters. So befor runing the program the user has to choose the blue parameters.

Each file contains two important sheets for the inputs. The other sheets are dedicated to the outputs. The main sheet is usually named “Control Panel”. It is the sheet where the user has to input the parameters like λ and μ and to choose the procedure he wants to use to compute the betas or the portfolios. The second input sheet contains Data. It is named “Variations” in the file BetaStudie.xls and “Kurse” in the file Finalsoft.xls. It contains the historical time-series of the stocks and the index and has to be updated regularly by the user as explained in the next paragraph.

The speadsheet is livered with data on the german stock market. It is possible to add some stock or to update the data. It is also possible to change the market and to input an other index and others stocks. The only convention is to give for each time-serie the line of the first figure in each column and the line of the last figure. The time-series don’t have to have the same number of observations. The algorithms automatically adjusts to compute over the deepest period.

Nevertheless the two Excel files differ in one point. In BetaStudie.xls the input time-series are the returns whereas in FinalSoft.xls the input time-series are the stock prices.

Let now approach more specifically each file.

5.2 Correlation estimation object

This part is relative to the object SinpleReg of the DLL and to the Excel file BetaStudie.xls. It explained the architecture of the object, its visual Basic link with Excel and the functioning of the Excel spreadsheet.

The sheet “Control Panel” contains in its upper left corner a rectangle where the the inputs have to be entered (in blue). There are six inputs. The first one is the stock number. Each stock is associated to a number. The list is given on the right side of the sheet. It is also possible to study the correlation between two stocks by putting a stock number in the line of index number instead of 0. the two next parameters are “Max length” and “Jump”. They correspond to the parameters M and m_0 in our modelisation. Finally the two last parameters are “lambda” and “mu”.

Once the parameters have been chosen, you just have to put on the button “RUN” and the betas are computed with all the algorithms exposed in this paper. The columns named “Normal” in the results’ table correspond to the regression over a fixed number of days M. The columns named “Chow” correspond to algotithm using the Chow test and the columns named “Spok” correspond to the adaptive procedure.

You can then read in green in the same sheet the results. A first serie of figures gives some statistics about the calculated betas and the regression coefficients R2. The table below gives some statistics about the forecasts generated by the models over one and five days.

STOCK	3
INDEX	0
Max Length	250
Jump	50
Lambda	3
Mu	2
Initiation	Done
BetaNormal	Done
BetaChow	Done
BetaSpok	Done
Stock	ALV
Observations	2173

RUN

3	ALV
4	BAS
5	BAY
6	CBK
7	DCX
8	DBK
9	DRS
10	DTE
11	LHA
12	MUV
13	SAP
14	SCH
15	SIE
16	THY
17	VEB
18	VOW
50	SIMUL
Beta1	
Vol1	
Beta2	
Vol2	

	R2			Beta		
	Normal	Chow	Spok	Normal	Chow	Spok
Min	47%	21%	21%	1,0251	0,6228	0,6228
Max	83%	91%	91%	1,3345	1,8046	1,8046
Mean	68%	67%	66%	1,1657	1,1770	1,1724
Var	1%	2%	2%	0,0031	0,0282	0,0377

	Forecast - 1 Day			Forecast - 5 Day Mean		
	Normal	Chow	Spok	Normal	Chow	Spok
Min	-5,76%	-6,37%	-6,37%	-2,63%	-2,56%	-2,56%
Max	10,20%	10,44%	10,44%	2,54%	2,64%	2,64%
Mean	-0,03%	-0,04%	-0,04%	-0,03%	-0,04%	-0,04%
MSE	0,02%	0,02%	0,02%	0,00%	0,00%	0,00%
MAE	0,93%	0,96%	0,96%	0,44%	0,45%	0,45%

Figure 17 – Screen overview of the Control Panel sheet of BetaStudy.xls

Finally four other sheets give a graphic overview on the results. The sheets that end with “-1” represent with a curve the fluctuation of the betas over the time and with an histogram the optimal depth m_0 computed at each time point to calculate beta. The sheets that end with “-2” give the same representation of the betas’fluctuation (in blue) and superpose on it the beta computed with a normal regression over a fixed period of *M* days (rose curve). They also print an histogram (in orange) of the relativ variations between the stock considered and the index (return of the stock minus return of the index). As the adaptive beta curve goes over the normal beta curve we can deduce that the stock becomes in comparison with the index more volatile than it was in the past. On the contrary, when the adaptive curve goes under the normal one the stock is becoming less volatile as it was comparatively to the index. These observations constitute a first analysis.

An other procedure permits to optimize lambda and mu. It corresponds to the button “Optimize” in the sheet “Control Panel”. On the left-side of the button a small rectangle permit you to input some parameters necessary to the computation. You have to put the maximal value for lambda and mu and the number of steps you want the algorithm makes. The procedure will then iterate the optimization process with various lambda and mu and finally retain the values that lead to the best results. For example if you put 5 as maximal value for lambda and mu, and 20 as number of steps the algorithm will run for the couple (lamdba,mu) equal to (0.25, 0.25), (0.25, 0.50), (0.25, 0.75), …, (5.00, 4.75), (5.00, 5.00) and will finally retain the couple that leads to the best results in term of forecast..

5.3 Tracking portfolio builder

The Excel file FinalSoft.xls does not differ much from the file BetaStudie.xls. The input parameters are the same except one that has been added, betap. It corresponds to the parameter β_P in our model. In order to choose the stocks you want to put in the tracking portfolio select it on the right-side of the sheet by clicking on it with the mouse.

You can then run separately each algorithm discribed in this paper.

- Bench indicates the benchmark portfolio as discribed in part 3.3. In order to use this algorithm you have to fill the sheet "Gewichte" with the time-series of the real weights of each stock in the index.

- Normal indicates that the beta are always computed over a fixed period given by the parameter "Depth". As we have seen in the study this algorithm generally permits to reduce approximatively the tracking error by two.

- Chow indicates the algorithm where the beta are calculated using the Chow test to check their stability.

- Spok indicates the algorithm with adaptive beta estimations. As we point out in the paper these two last algorithms do not improve significanlty the tracking error on the Dax-index but results to an other distributions of the weights for each stocks.

For each procedure a table gives some statitics on the results : the minimal error, the maximal error, the average error, the standard error, the mean squared error and the mean absolute error. This last indicator is important to compare the performance of the procedures. As said befor, the results ara printed in green.

Finally two graphics give a representation of the results for the last used procedure. The sheet "Tracking Error" is an histogram representation of the tracking error and the sheet "Weights" represents the evolution of the weights of each stock in the tracking portfolio. The data used to construct the graphics are disponible in the sheet "Results". Actually it could be possible to link this sheets directly with an other application and to pursue the research on the option market for examle.

6 Conclusion and outlook

A new algorithm for minimizing the tracking error with a limited number of given stocks is proposed in this paper. It assumed that a local constant approximation of the correlation between stock and index holds over some unknown interval. The issue of filtering this interval of time homogeneity out of the return time series is considered, and a nonparametric approach is used to optimize some parameters.

Finally the study points out two main results and let a software usefull for further developments. So we can resume it in three points :

- Considering the correlation between stocks and index the tracking error has been divided by two.
- Introduction of adaptive estimators does not improve significantly the tracking error but gives an other distribution of weights.
- A soft has been implemented for further developments and studies of correlations in equity derivatives.

Comparatively to the model we take has benchmark we succeed in dividing the tracking return error by two. This result is of importance to construct arbitrage. The direct application of this algorithm would be to my mind intra-day base arbitrage between cash and future.

Although the adapative estimation we propose doesn't improve significantly more the tracking return error as the model with constant regression period, it permits to build a tracking portfolio with other stocks' weights. This consideration is of importance if we intervene on the option market because we will take into account the implied volatility structure of the stocks.

Applicate the results of this paper to the option market would be the next step. I see it as a topic for future research. First we can just use the adaptive correlation calculation object to buid other application or other portfolios. Secondly we can directly use the tracking portfolio builder object. The final application on the option market would be to determine the optimal weight for being short index-options, long several stock-options and extract a positiv PnL from gamma and vega valorisation.

7 References

Blume, M.E (1975). Betas and their regression tendencies, *Journal of Finance* **30** : 785ff.

Bollerslev, T. Generalised autoregressive conditional heteroskedasticity, in Engel (1995b).

Brockmann, M., Gasser, T., Herrmann, E. (1993). Locally adaptive bandwidth choice for kernel regression estimators, *Journal of the American Statistical Association* **88** : 1302-1309

Carroll, R. and Ruppert, D. (1988). *Transformation and Weighting in Regression*, Chapman and Hall, New York. Chapter **4**.

Clements, M.P. and Hendry, D. F. (1998). *Forecasting Economic Time Series*, Cambridge University Press, Cambridge.

Engel, R.F. (1995a). Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation, in *ARCH, selected readings* (Engle, 1995b).

Gouriéroux, C. (1997). *ARCH Models and Financial Application*, Springer-Verlag, Berlin.

Härdle, W., Spokoiny, V., Teyssiere, G. (2000). Adaptive estimation for a time-inhomogeneous stochastic-volatility model, Discussion Paper, Sonderforschungsbereich 737, Humboldt University, Berlin.

Härdle, W., Herwatz, H. and Spokoiny, V. (2000). Time inhomogeneous multiple volatility modelling, Discussion Paper, Sonderforschungsbereich 737, Humboldt University, Berlin.

Harvey, A., Ruiz, E. and Shephard, N. (1995). Multiple stochastic variance models, *in* Engle (1995b).

Lepski, O. (1990). One problem of adaptive estimation in Gaussian white noise, *theory Probability Application* **35** : 459-470.

Pollak, M. (1995). Optimal detection of a change of distribution, *Annals of Statistics* **13** : 206-227.

Rudd, A., Rosenberg, B. (1979). Realisic Portfolio Optimization, in *Studies in the Management Sciences* **11**, North-holland, Amsterdam, New-Yrk, Oxford.

Taylor, S. J. (1986). *Modelling Financial Tima Series*, Wiley, Chichester.

Wagner, N.F. (1996). Approximative Nachbildung des Deutschen Aktienindexes, *Finanzmarkt und Portfolio Management* **3** : 375ff.