Backtest overfitting in financial markets

David H. Bailey * Jonathan M. Borwein† Amir Salehipour ‡
Marcos López de Prado § Qiji Zhu ¶

February 9, 2016

1 Introduction

In mathematical finance, *backtest overfitting* means the usage of historical market data (a *backtest*) to develop an investment strategy, where many variations of the strategy are tried on the same dataset. Backtest overfitting is now thought to be a primary reason why quantitative investment models and strategies that look good on paper (based on backtests) often disappoint in practice. Models suffering from this condition target the specific idiosyncrasies of a limited dataset, rather than any general behavior, and, as a result, often perform poorly when presented with new data.

Backtest overfitting is an instance of the more general phenomenon of multiple testing in scientific research, where a large number of variations of a model are tested on the same data, without accounting for the increase in false positive rates. Standard overfitting techniques, such as the hold-out method, fail to identify this problem, because they are designed to evaluate the complexity of a model relative to the dataset, still assuming that a single test has taken place.

An example will clarify this difference: Suppose that a new compound XYZ is developed to treat headaches. We wish to test for the hypothesis that XYZ is actually effective. A false positive occurs when we falsely determine that XYZ has been effective. This can occur for a variety of reasons: the patient was misdiagnosed, the pain associated with headache oscillated closely to the threshold level to declare the condition, etc. Suppose that the probability of false positive is only 5%. We could test variations of the compound by changing an irrelevant characteristic (the color, the taste, the shape of the pill), and it is expected that at least 1 in 20 of those variations will be (falsely) declared effective. The problem does not lie with biology or the complexity of the compound. Instead, the researcher has conducted multiple tests while treating each individually, not realizing that in doing so she has incurred in an increasing probability of false positives. Full body scans and other current technology-driven medical diagnoses and methods are often compromised for the same reason.

Likewise, in finance it is common to conduct millions, if not billions, of tests on the same data. Authors do not typically provide the number of experiments involved in a particular discovery, and as a result it is likely that many published investment theories or models are false positives. For example, in [3] the authors show that if only five years of daily stock market data are available as a backtest, then no more than 45 variations of a strategy should be tried on this data, or the resulting strategy will be overfit, in the specific sense that the strategy's Sharpe Ratio (SR) is likely to be 1.0 or greater just by chance (even though the true SR may be zero or negative).

The Sharpe Ratio (SR) and similar statistics are used to allocate capital to the best performing strategy. SR quantifies the performance of an investment strategy [6, 7], and is the ratio between average excess returns on capital, in excess of the return rate of a risk-free asset, and the standard deviation of the same returns [3]. Thus, the higher the ratio, the greater the return relative to the risk involved.

Anyone who develops or even merely invests in a systematic investment strategy (or in an exchange traded fund based on such a strategy) needs to understand the degree to which strategies can be overfit,

*Lawrence Berkeley National Laboratory (retired), 1 Cyclotron Road, Berkeley, CA 94720, USA, and University of California, Davis, Department of Computer Science. E-mail: david@davidhbailey.com.
†CARMA, University of Newcastle NSW 2308, Australia. E-mail: jonathan.borwein@newcastle.edu.au.
‡CARMA, University of Newcastle NSW 2308, Australia. E-mail: a.salehipour@newcastle.edu.au.
§Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley, CA 94720, USA. E-mail: lopezdeprado@lbl.gov.
¶Department of Mathematics, Western Michigan University, Kalamazoo, MI 49008, USA. E-mail: qiji.zhu@wmich.edu.

in order to avoid unexpected financial losses. For this reason, we have developed two online tools: the *Backtest Overfitting Demonstration Tool* (BODT) and the *Tenure Maker Simulation Tool* (TMST). The major goal of the tools is to demonstrate how easy is to overfit an investment strategy, and how this overfitting may impact the financial bottom-line performance. These two tools stem from two broad types of investment strategies [3]:

- Those based on general trading rules, e.g. seasonal opportunities (BODT targets this type).
- Those based on forecasting equations, e.g. econometric models (TMST targets this type).

BODT employs a simplified version of the process many financial analysts use to create investment strategies, namely to use a computer program to find the optimal strategy based on historical market data (often termed "in-sample" (IS) data), by adjusting variables such as the holding period, the profit taking and stop loss levels, etc. Similarly, TMST applies forecasting and econometric equations in order to find the "optimal" strategy. If care is not taken to avoid backtest overfitting, such strategies may look great on paper (based on tests using historical market data), but then give rather disappointing results when actually deployed on a different dataset (often termed "out-of-sample" (OOS) data). Figure 1 illustrates this phenomenon; the left plot shows how an optimal strategy (associated with the blue line) can be developed based on a historical dataset or IS dataset (which in this case is merely a pseudorandomly generated set of daily closing prices and is associated with the green line) by varying *entry day*, *holding period*, *stop loss* and *side* parameters (later in Section 2.2 we discuss these parameters in more detail). This optimal strategy has a SR of 1.59 on IS dataset. The right plot, on the other hand, illustrates the same optimal strategy performs poorly on OOS dataset and results in the SR of -0.18, evidencing that the strategy has been overfit on the IS data; in fact, the optimal strategy actually *lost* money here.

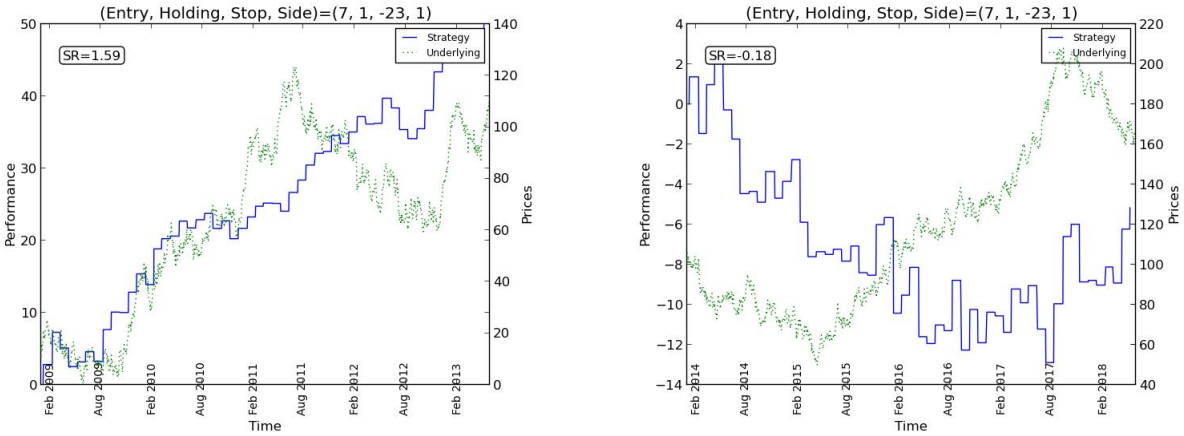

Figure 1: Sharpe Ratio (SR) optimized for the IS data (left) and observed on the OOS data (right); on the OOS data, the SR is negative, indicating that it lost money.

The online BODT and TMST focus on demonstrating the impact of overfitting. A more technical version of BODT and TMST were introduced in [4]. The impact of backtest overfitting has been discussed in detail in [5]. For the single testing case, Bailey, Borwein, López de Prado and Zhu [3] proposed the *Minimum Backtest Length* (MinBTL) as a metric to avoid selecting a strategy with a high SR on IS data, but zero or less on OOS data. A *probabilistic Sharpe Ratio* (PSR) was proposed in [1] to calculate the probability of an estimated SR being greater than a benchmark SR. For the multiple testing case, Bailey and López de Prado [2] developed the *Deflated Sharpe Ratio* (DSR) to provide a more robust performance statistic, in particular, when the returns follow a non-normal distribution.

The remainder of the note is organized as follows. In Section 2 we explain the Backtest Overfitting Demonstration Tool (BODT). This includes the structure of the tool, the datasets, the parameters, and the types of experiments available to the user, as well as the graphical and numerical outcomes of BODT. Similarly, in Section 3 we explain the Tenure Maker Simulation Tool (TMST). This includes the structure of the tool, the parameters, the types of experiments available to the user, and the graphical outcomes. The paper ends with a few conclusions.

2 The Backtest Overfitting Demonstration Tool

Seasonal strategies are very popular among investors, and are marketed every day in TV shows, business publications and academic journals. In this section we illustrate how trivial it is to overfit a backtest involving a seasonal strategy. Backtest Overfitting Demonstration Tool (BODT) finds optimal strategies on random (unpredictable)/real-world stock market data, and demonstrates that high Sharpe ratios (SR) on backtest in-sample data are meaningless unless investors control for the number of trials.

2.1 The structure of the tool

BODT has two modules: the optimization module, which is the core of BODT (coded in the programming language Python), and the communication module, which is an online interface providing a bridge between the user and the optimization module. In particular, the online interface collects and/or sets the parameters values, supplies them to the optimization program, and reports the outcomes from the optimization program. BODT performs the following four steps:

1. **Importing data and setting parameters.** This include importing/setting the parameters, and importing S&P500 real-world stock market data/generating pseudorandom data (depending on the type of the experiment chosen by the user, see Section 2.3). If random experiments are chosen, given three parameters, the *sample length (number of days or the length of the time series)*, the *standard deviation* and the *seed*, simulated daily closing prices of a stock are derived by drawing returns from a Gaussian distribution with mean zero. If the real-world experiment is chosen, the data values are daily closing prices of the S&P500 index between January 1962 and February 2014. In each case, the sample data is equally divided into two sets: the in-sample (IS) dataset (also known as the *training set*), and the out-of-sample (OOS) dataset (also known as the *testing set*).

2. **Obtaining the "optimal" strategy.** BODT generates all investment strategies. Investment strategies are formed by successively adjusting the four parameters the *holding period*, the *stop loss*, the *entry day*, and the *side* (it performs a brute-force search by trying all combinations of the four parameters). Every strategy is evaluated by calculating the Sharpe Ratio (SR), on the IS sample data, and the optimal trading strategy, in terms of optimizing the SR, is chosen.

3. **Evaluating the optimal strategy on the OOS data.** The "optimal" strategy obtained above is then applied to the OOS data and the SR statistic is computed. In particular, the strategy is evaluated over the IS set in Step 2; then after exploring the best performing strategy, it is evaluated over the OOS set. Note that the OOS set is not used in the design of the strategy. A backtest is said to be *realistic* when the IS performance is consistent with the OOS performance, after controlling for the number of experiments that have taken place.

4. **Visualization.** The outcomes of BODT include three plots, a movie and a summary of the numerical values. The first two graphs, which are similar to Figure 1, show results on the IS set, i.e., the backtest (the graph on the left) and the OOS data (the graph on the right). In these two graphs, the green line is the underlying time series, and the blue line shows the performance of the strategy. In most runs, the SR of the right graph (i.e., the final strategy on the OOS data) is either negative or at the very least much lower than the SR of the final left graph (i.e., the final strategy on the IS data), evidencing that the strategy has been overfit on the IS data. A "movie" showing the progression of the generation of the optimal strategy on the IS data can be played by clicking on the graph on the left (in the online demo).

   The third graph, which is depicted in Figure 2, shows the value of the advanced Deflated Sharpe Ratio (DSR) statistic [2] over changes in the value of the *number of trials*, as a blue line. The same for a benchmark setting (skewness: -3 and kurtosis: 10) has been shown as a red line, only to give an idea of different behavior given a change in the values of skewness and kurtosis. Finally, a set of numerical values will be reported in a table similar to Table 1. These include the used parameters as well as values of SR and DSR statistics.

The execution time of BODT is typically less than two minutes. The values for the maximum holding period, the stop loss and the sample length significantly affect the number of iterations performed by the program; the larger these values are, the longer the program will run. BODT is available to the public for free and can be accessed through https://carma.newcastle.edu.au/backtest/. A more detailed explanation is documented in a tutorial and may be downloaded from the above webpage.

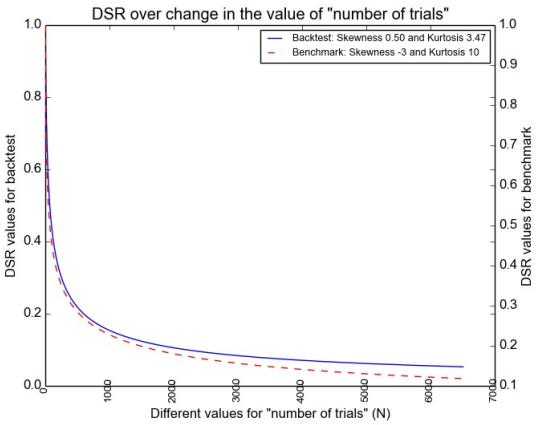

Figure 2: An advanced DSR analysis performed by BODT over the change in the number of trials. Note also how changes in skewness and kurtosis affect DSR.

Parameters for this run	
Maximum Holding Period	20
Maximum Stop Loss	23
Sample Length	1152
Standard Deviation	2
Seed	308
Real-World Stock Market Data Used	No
Sharpe Ratio (SR) of OOS Data	-0.2260
Deflated Sharpe Ratio (DSR) of IS Data	0.3744

Table 1: Illustration of numerical outcomes of BODT. The values of Table 1 are only for illustration purposes.

2.2 Parameters

Table 2 shows the parameters of BODT. The user has no control over some of these parameters, which are denoted by ✓ in column "Fixed value"; for these parameters, BODT uses the default values as shown in column "Default." Note if the user does not enter a value or enters a value that is outside the permissible ranges, a default value will be used. The reason for these feasible ranges is to place an upper limit to the number of trials (or optimization iterations) conducted. Such a limit does not imply a loss of generality with regards to the analysis. On the contrary, we show that overfitting can deliver significantly high performance (in-sample) even for a relatively small number of iterations. The parameters of BODT are:

1. **Max. holding period:** the number of days that a stock can be held before it is liquidated (sold). It is given in a whole number of trading days. BODT tries all integer values less or equal to the maximum given by the user.
2. **Max. stop loss:** the percentage of invested capital that can be lost before the position is liquidated (closed). BODT only tries integer percentages up till the maximum given by the user.
3. **Sample length:** the number of observations used in-sample.
4. **Standard deviation:** the standard deviation of random returns used to generate daily prices.
5. **Seed:** a seed for the pseudorandom numbers used to generate the random returns.
6. **Entry day:** the day that one enters into the market in each trading month. Every trading month is assumed to include 22 entry days. All 22 possibilities are tried by BODT.
7. **Side:** the side of the held positions, either *long*, which is to make profits when stock prices are rising, or *short*, which is to make profits when stock prices are falling. Both options are tried by BODT.

Parameters	Fixed value	Default	Random data experiments			Real-world S&P500
			Exper. 1	Exper. 2	Exper. 3	Exper. 4
Max. holding period	×	7	20	[5, 20]	[5, 20]	[5, 20]
Max. stop loss	×	10	23	[10, 40]	[10, 40]	[10, 40]
Sample length	×	1000	1152	[1000, 2000]	[1000, 2000]	[5000, 6000]
Standard deviation	×	1	2	any pos. integer	any pos. integer	from data
Seed	×	1	308	any pos. integer	any pos. integer	not relevant
Entry date	✓	$1, \ldots, 22$				
Side	✓	$(-1, +1)$				

Table 2: The parameters of BODT over different running options.

2.3 Four types of experiments

To study the impact of overfitting, BODT performs four different types of experiments, which are explained below. The first three are based on randomly generated data (daily closing prices) from the Gaussian distribution with the standard deviation and the seed values/ranges as given in Table 2. The last experiment is based on S&P500 data.

1. **Experiment 1.** *Replicating a specific example.*

   The first experiment replicates a specific example which is associated with two plots of Figure 1 (the same plots are displayed on the webpage of BODT as well). Thus, the user can replicate this experiment by calling the pre-set values for parameters.

2. **Experiment 2.** *Generating parameters randomly.*

   The second experiment uses randomly generated integer parameters, from the ranges allowed for each parameter.

3. **Experiment 3.** *User-defined parameter values.*

   The third experiment asks the user to enter parameters. The user may enter any values from the specified ranges for the first five parameters of Table 2. If any parameter is left blank, then a random value is generated from the feasible ranges by BODT. In this experiment, the user has the option to impact the data generation by choosing the standard deviation and the seed values.

4. **Experiment 4.** *Using actual stock market data.*

   The fourth experiment asks the user to enter parameters for real financial data, i.e., for S&P500 stock market data, where daily closing prices are taken from January 1962 to February 2014. Our preference for this index is motivated by its wide acceptance as a benchmark and financial instrument. Standard deviation is implied by the data and seed parameter is not relevant in this experiment. Note that due to the size of the S&P500 index data, the ranges for the parameter *sample length* has changed.

3 The Tenure Maker Simulation Tool

Section 2 illustrated how easy is to overfit a backtest involving a seasonal strategy. But what about the rest of strategies? Are strategies based on academic econometric or statistical methods easy to overfit as well? Unfortunately the answer is that these pseudo-mathematical investments are even easier to overfit. The Tenure Maker Simulation Tool (TMST) looks for econometric specifications that maximize the predictive power (in-sample) of a random (unpredictable) time series. The resulting Sharpe ratios tend to be even higher than in the "seasonal" counterpart. The implication is that most scientific strategies published in rigorous academic journals are likely to be overfit. These publications are the basis on which lecturers receive a tenure, hence the tool's name.

3.1 The structure of the tool

Similar to BODT, the core of the *Tenure Maker Simulation Tool* or TMST is an optimization program coded in the programming language Python (the optimization module) and is communicated to the user

via an online interface (the communication module). The online interface collects and/or sets the parameters values, supplies them to the optimization program, and reports the outcomes from the optimization program. TMST is a free tool; a more detailed explanation is documented in a tutorial which may be downloaded from the webpage of the online TMST at https://carma.newcastle.edu.au/tenuremaker/. TMST performs the following four steps.

1. **Generating returns.** A series of IID (independent, identically distributed) Normal returns are generated. The sample data is considered the in-sample (IS) set.
2. **Generating time series model.** A set of time series models are generated, where the series is forecasted as a fraction of past realizations of that same series; the forecasted series is considered the out-of-sample (OOS) set. The time series models include:
   (a) Rolling sums of the past series;
   (b) Polynomials of the past series;
   (c) Lag of the past series; and
   (d) Cross-products of the above.
3. **Strategy evaluation:** A forward-selection algorithm evaluates the generated strategies, in terms of optimizing SR, and selects the improved model.
4. **Visualization:** TMST outputs two graphs, which are shown in Figure 3. The first graph (on the left) shows the backtest, i.e. how the "optimal" strategy is obtained. In this graph, the green line represents the trading strategy behavior, and the blue line represents the market behavior. The second graph (on the right) shows "inflation" progress in the annualized Sharpe Ratio (aSR). A "movie" showing the progression of the generating the strategies can be played by clicking on the graph on the left (in the online demo).

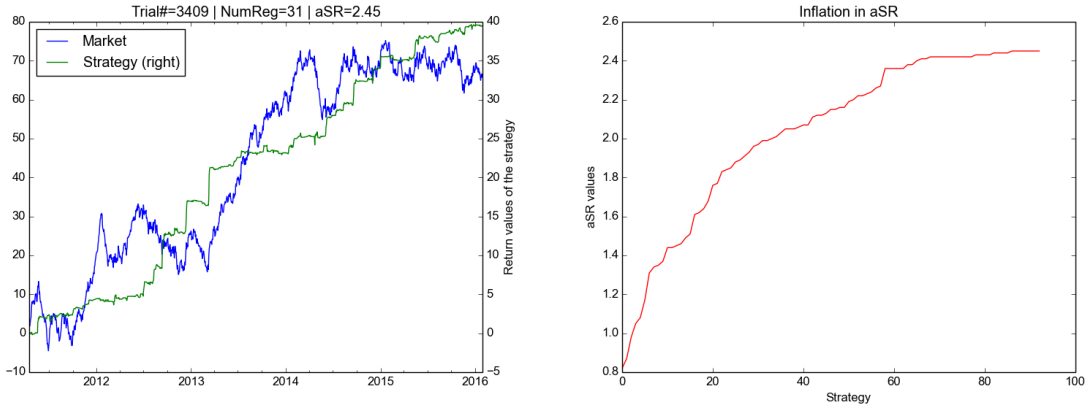

Figure 3: Annualized Sharpe Ratio (aSR) optimized for the IS data (left) and inflation progress in aSR resulted by generating and selecting strategies (right). As the right graph shows, SR is increasing, as a result of overfitting.

The green line, in the left graph of Figure 3, gets more and more profitable over time as the program continues to optimize the system to fit historical data. In a matter of seconds or minutes, the program creates what appears to be a very profitable equity curve (with a very high SR) based on the input dataset. In fact, we are predicting future realizations of the series by using past realizations, which is of course impossible by construction. The SR is even more inflated than in the "seasonal" counterpart (those based on general trading rules). This is one justification that why econometric specifications are so flexible that it is even easier to generate a large number of independent trials.

3.2 Parameters

TMST has six parameters. Five of these parameters are not available to the user (those denoted by ✓ in column "Fixed value"); for these parameters, TMST sets pre-specified values as shown in column "Default." The six parameters are:

1. **Sample length:** the number of observations (IID returns) generated.
2. **Width:** sample length used as the look-back period in the rolling sum regression models.
3. **Polynomial degree:** degrees of the polynomial fit used in the polynomial regression model.
4. **Number of lags:** number of lagged variables included in the lagged regression model.
5. **Number of cross products:** size of the cross product regressors.
6. **Max. computational time:** this is the one parameter that is available to the user and is the total computational time in seconds, that the optimization module is allowed to generate the strategies. It is in the ranges [30, 900] seconds, and the default value is 90 seconds. Only integer values are allowed. Moreover, if the user does not enter any value for this parameter or if the value is out of the specified ranges, the default value, which is 90, will be used.

Parameters	Fixed value	Default	Experiments	
			Experiment 1	Experiment 2
Max. computational time	×	90	not relevant	[30, 900]
Sample length	✓	1250	1250	
Width	✓	3	3	
Polynomial degree	✓	3	3	
Number of lags	✓	1	1	
Number of cross products	✓	3	3	

Table 3: The parameters of TMST over different running options.

3.3 Two types of experiments

The following two options are available.

1. **Experiment 1.** *Full.*

   The program stops when all the strategies are generated, which may take up to 10 minutes.

2. **Experiment 2.** *Limited.*

   The user limits generation of the strategies (in the optimization module) by setting the maximum computational time.

4 Conclusion

Financial research is increasingly reliant on computational techniques to simulate a large number of alternative investment strategies on a given dataset. One problem with this approach is that the standard Neyman-Pearson hypothesis testing framework was designed for individual experiments. In other words, when multiple trials are attempted, the significance level (i.e., probability of a false positive) is higher than the value set by the researcher.

Academic articles and investment proposals almost never disclose the number of trials involved in a particular discovery. Consequently it is highly likely that many published findings are just statistical flukes. The practical implication is that investors are being lured into allocating capital to irrelevant discoveries, financial theories or investment products.

The *Backtest Overfitting Demonstration Tool* (BODT) and the *Tenure Maker Simulation Tool* (TMST) are, to our knowledge, the first scientific software to illustrate how overfitting impacts financial investment strategies and decisions in practice. In particular, it shows how the *optimal* strategy identified by backtesting the in-sample data almost always leads to disappointing performance when applied to the out-of-sample data. Our main goal with BODT and TMST is to raise awareness regarding the problem of backtest overfitting in the financial literature. We invite the reader to explore the tools and provide feedback: https://carma.newcastle.edu.au/backtest/ and https://carma.newcastle.edu.au/tenuremaker/.

References

[1] D. H. Bailey and M. López de Prado. "The Sharpe Ratio Efficient Frontier". In: *The Journal of Risk* 15.2 (2012), pp. 3–44.

[2] D. H. Bailey and M. López de Prado. "The Deflated Sharpe Ratio: Correcting for Selection Bias, Backtest Overfitting, and Non-Normality". In: *Journal of Portfolio Management* 40.5 (2014), pp. 94–107.

[3] D. H. Bailey, J. M. Borwein, M. López de Prado, and Q. J. Zhu. "Pseudomathematics and financial charlatanism: The effects of backtest over fitting on out-of-sample performance". In: *Notices of the AMS* 61.5 (2014), pp. 458–471.

[4] D. H. Bailey, J. M. Borwein, A. Salehipour, M. L. de Prado, and Q. J. Zhu. "Online Tools for Demonstration of Backtest Overfitting". In: *SSRN* 2597421 (2015). URL: `http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2597421`.

[5] D. H. Bailey, J. M. Borwein, M. López de Prado, and Q. J. Zhu. "The probability of backtest overfitting". In: *Journal of Financial Mathematics* (Forthcoming, accepted March 2015).

[6] W. F. Sharpe. "Mutual fund performance". In: *Journal of Business* (1966), pp. 119–138.

[7] W. F. Sharpe. "The Sharpe ratio". In: *The Journal of Portfolio Management* 21.1 (1994), pp. 49–58.