LAPPEENRANTA-LAHTI UNIVERSITY OF TECHNOLOGY LUT
School of Engineering Science
Computational Engineering
Bachelor’s thesis

Pauli Anttonen

Fourier transform techniques for noise reduction

Examiner: Teemu Härkönen

ABSTRACT

Lappeenranta-Lahti University of Technology LUT

School of Engineering Science

Computational Engineering

Pauli Anttonen

Fourier transform techniques for noise reduction

Bachelor's thesis

2022

32 pages, 22 figures, 4 tables

Examiner: Teemu Härkönen

Keywords: Fourier transform; noise reduction; white noise; digital signal processing;

The aim of the thesis was to study properties of Fourier transform and its applicability in noise reduction. Mathematical properties behind Fourier transform and white noise was first explained. Three Fourier transform based noise reduction techniques were then considered and validated using two synthetic data sets and applied to IBM stock data to remove effects of stochastic perturbations.

Based on visual observations, all three method were produced credible results. Amplitude thresholding method effective noise reduction method with trigonometric background process. Two low-pass filters produced weaker results on synthetic data but all three methods gave convincing results on stock data. To further improve results, bigger sampling frequency could have been chosen which had produced more data points to base the transform on.

TIIVISTELMÄ

Lappeenrannan-Lahden teknillinen yliopisto LUT
School of Engineering Science
Laskennallisen tekniikan koulutusohjelma

Pauli Anttonen

Fourier-muunnoksen käyttö kohinan poistossa

Kandidaatintyö

2022

32 sivua, 22 kuvaa, 4 taulukkoa

Ohjaaja: Teemu Härkönen

Avainsanat: Fourier-muunnos, kohinan poisto, valkoinen kohina, signaalin digitaalinen käsittely

Työn tarkoituksena oli perehtyä Fourier-muunnokseen ja tarkastella sen soveltuvuutta kohinan poistoon. Ensin työssä käsiteltiin Fourier-muunnoksen ja valkoisen kohinan teoriaa ja ominaisuuksia. Kolmea Fourier-muunnosta hyödyntävää menetelmää testattiin kahdelle kohinaiselle synteettiselle datalle sekä IBM:n osakedatalle stokastisten häiriöiden poistamiseksi osakkeen hinnasta.

Visuaalisen tarkastelun pohjalta kaikki kolme menetelmää poistivat kohinaa datasta onnistuneesti. Taajuuskomponenttien rajaaminen amplitudin perusteella osoittautui kaikista toimivimmaksi menetelmäksi trigonometrisen taustaprosessin approksimointiin. Kaksi muuta testattua alipäästösuodatinta tuottivat heikompia tuloksia synteettiselle datalle, mutta kaikki kolme menetelmää antoivat hyvän approksimaation osakedatalle. Tulosten parantamiseksi datan näytteenottotaajuutta voitaisiin lisätä.

Contents

1 INTRODUCTION

1.1 Overview

Fourier transform is one the of the most widely used techniques in applied mathematics such as in the fields of signal processing and data compression. In brief, Fourier transform is based on the idea that functions of suitable properties can be represented with a linear combinations of trigonometric functions. This decomposition using trigonometric functions can be seen as extracting frequency-domain information from a time-domain signal, allowing for alternative or more efficient ways of investigating or modifying a signal.

Initial work on the Fourier transform and the use of trigonometric functions in analysis traces back to works of mathematicians Leonhard Euler (1707-1783), Alexis-CLaude Clairaut (1713-1765), and Joseph Louis Lagrange (1736-1813). Euler was the first one to give a formula for coefficients of the Fourier series. Based on Euler's studies, Clairaut published what we currently know as the first formula for the discrete Fourier transform (DFT) in 1754. In 1805 Carl Friendrich Gauss (1777-1855) published formula for DFT that was not dependent on interpolating using only odd or even periodic functions. In 1822, Joseph Fourier (1768-1830) published work on heat flow which provides the foundation for the Fourier transform and origin for its name. [1]

DFT is applicable to finite sequency of equally-spaced samples. While effective, DFT suffers from a cubic computational complexity. This was amended by the invention of the fast Fourier transform (FFT) algorithm by James Cooley and John Tukey in 1965 which has enabled a wide range of numerical applications in the digital age [1]. These applications include JPEG data compression, analysis of crystal structures, solving partial differential equations, and handling noisy signals [1, 2].

This thesis focuses introducing theoretical properties behind chosen Fourier transform based noise reduction techniques and their numerical implementation. Two sets of synthetic data sets are used to validate the applicability of the noise reduction techniques by comparing the obtained results to the underlying smooth functions used to generate the data sets. Lastly, the techniques are applied to International Business Machines Corporation (IBM) stock data to estimate a smooth mean function of the stochastic stock prices. In this thesis, measurement errors are modelled as Gaussian white noise.

1.2 Noisy data

Analysing measured data has its own challenges involving unpredictable conditions and systematic measurement errors. These factors introduce noise into the data, thereby complicating analysis and possibly causing biased or incorrect conclusions. For many application, it can be considered useful to find a function to model the background process instead of using noisy measurements themselves. In mathematics and engineering, white noise is a widely used tool used to model effects of stochastic processes in measured data [3].

Commonly used methods for noise reduction with one-dimensional data signal include adaptive filtering, signal smoothing, and noise cancellation in frequency domain [4]. Adaptive noise cancellator locates points where a signal is unpredictable and aims to adjust the value of these points to fit rest of the data better. Smoothing algorithms reduce noise by smoothing the curve using various techniques. One way this can be accomplished is by calculating mean value of three consecutive points, thereby reducing noise by partly averaging out the noise. Alternatively, one could consider smoothing or noise reduction in the frequency domain. As previously mentioned, this can be achieved by the use of Fourier transforms.

Noise filtering via Fourier transforms has seen numerous applications. Examples of such applications are canceling out electromagnetic conditions in radar measurements [5], echo suppression in audio processing [6], and 2D noise filter in image processing [7]. In the above, Fourier transform has proven to be functional and efficient tool for noise reduction.

1.3 Structure of thesis

Further down, the thesis is structured as follows. In Section 2, theory of the Fourier transform and Gaussian white noise is covered and introduced. Models used to create the synthetic data sets are presented in Section 3. In Section 4, results are presented and comparison between different techniques is presented. Discussion on the results is given in Section 5.

2 THEORY

2.1 Fourier transform

Fourier transforms decomposes a function into coefficients of a linear combination of trigonometric functions. Fourier transform for a function $f(t)$ can be given as

$$\hat{f}(\omega) := \int_{-\infty}^{\infty} f(t)e^{-i\omega t}dt, \tag{1}$$

where f denotes an arbitrary function dependent on time t and the Fourier transform of f is given as $\hat{f}$ dependent on frequency ω. The corresponding inverse Fourier transform is

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(\omega)e^{i\omega t}d\omega. \tag{2}$$

In practice, we only have finite sequences of discrete measurements. For finite evenly-spaced sequences, we apply the discrete Fourier transform (DFT) given as

$$Y_k = \sum_{n=0}^{N-1} X_n \cdot e^{\frac{-2\pi i}{N}kn}, \tag{3}$$

where Y_k denotes the k-th element in the discrete Fourier transform vector and X_n denotes the n-th element in the original time sequence. N is the number of elements in the time series. Similarly to Eq. (2), we have the inverse discrete Fourier transform (IDFT) defined as

$$X_n = \frac{1}{N}\sum_{k=1}^{N} Y_k \cdot e^{\frac{2\pi i}{N}kn}. \tag{4}$$

Computation of the DFT and its inverse transform can be optimized significantly by pairing up even and odd functions during computation. This pairing of even and odd functions is called the fast Fourier transform (FFT) with corresponding inverse fast Fourier transform (IFFT) which eases the computational complexity of the DFT from $O(N^2)$ to $O(N\log N)$.

2.2 Fourier transforms of trigonometric functions

Trigonometric functions are commonly used in engineering and mathematics to model periodical fluctuations. Fourier transforms of trigonometric functions can be given as

$$\int_{-\infty}^{\infty} \sin(\Omega t)e^{-i\omega t}dt = i\pi[\delta(\omega+\Omega)+\delta(\omega-\Omega)], \tag{5}$$

and

$$\int_{-\infty}^{\infty} \cos(\Omega t)e^{-i\omega t}dt = \pi[\delta(\omega+\Omega)-\delta(\omega-\Omega)], \tag{6}$$

where coefficient Ω is the frequency of trigonometric function and δ denotes the Dirac delta function. As can be seen from Eqs. (5) and (6), the Fourier transforms of sines and cosines consist of sharp spikes located at the frequencies of the original functions. An illustration of this behaviour is shown in Figure 1.

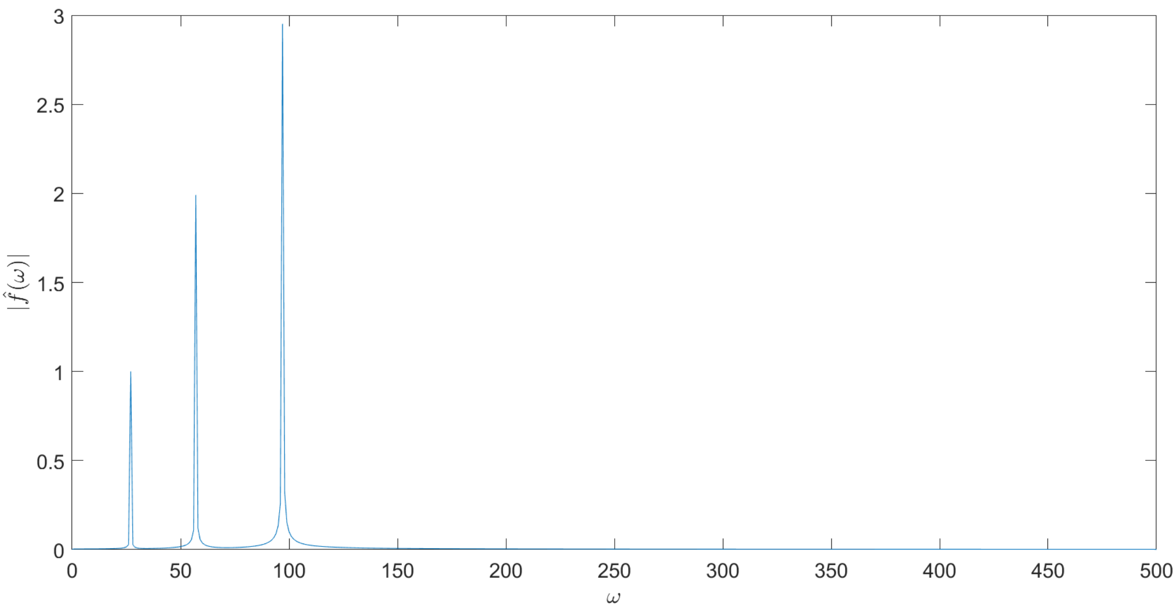

Figure 1. Fourier transform of trigonometric functions. Frequency spikes at certain ω-values suggest that the example function consists from three trigonometric components. Coefficients for these components can be seen from y-axis values.

2.3 Decay of the Fourier coefficients

Differentiable functions are ubiquitous in mathematics and engineering including common functions such as polynomials and exponential functions amongst a plethora of others. Fourier transforms of differentiable functions form decaying frequency spectrums depending on the smoothness of the functions. Mathematically, this is described in Theorem 1.

Theorem 1 (Fourier component decay)**.** *Let Fourier transform exist for $f^{(k)}$ when $0 \leq k \leq p$ and that $\int f^{(p)}(t)dt < \infty$. Then, we have that the Fourier transform of f bounded according to $|f(\omega)| < C|\omega|^{-p}$, where C is a constant independent of w.*

Proof. According to the differentiation rule of Fourier transforms, we have

$$f^{(k)}(t) = \frac{d^k}{dt^k}\left(\frac{1}{2\pi}\int e^{i\omega t}\hat{f}(\omega)d\omega\right) = \frac{1}{2\pi}\int (i\omega)^k e^{i\omega t}\hat{f}(\omega)d\omega. \tag{7}$$

Taking inverse Fourier transform and choosing $k = p$ gives expression

$$(i\omega)^{(p)}\hat{f}(\omega) = \int e^{-i\omega t} f^{(p)}(t)dt. \tag{8}$$

Upper bound for the differentiated expression can be given as

$$|\omega|^{(p)}|\hat{f}(\omega)| \leq \int |f^{(p)}(t)|dt. \tag{9}$$

With the integral having finite value, it can be assumed that there exist constant C giving upper bound to Fourier coefficients as

$$|\hat{f}(\omega)| \leq C|\omega|^{-p}. \tag{10}$$

□

In simple terms, the number of derivatives corresponds to the smoothness of the function. The larger the p and smoother the curve, the faster the decay of the Fourier transform. Upper bound is visualised in Figure 2.

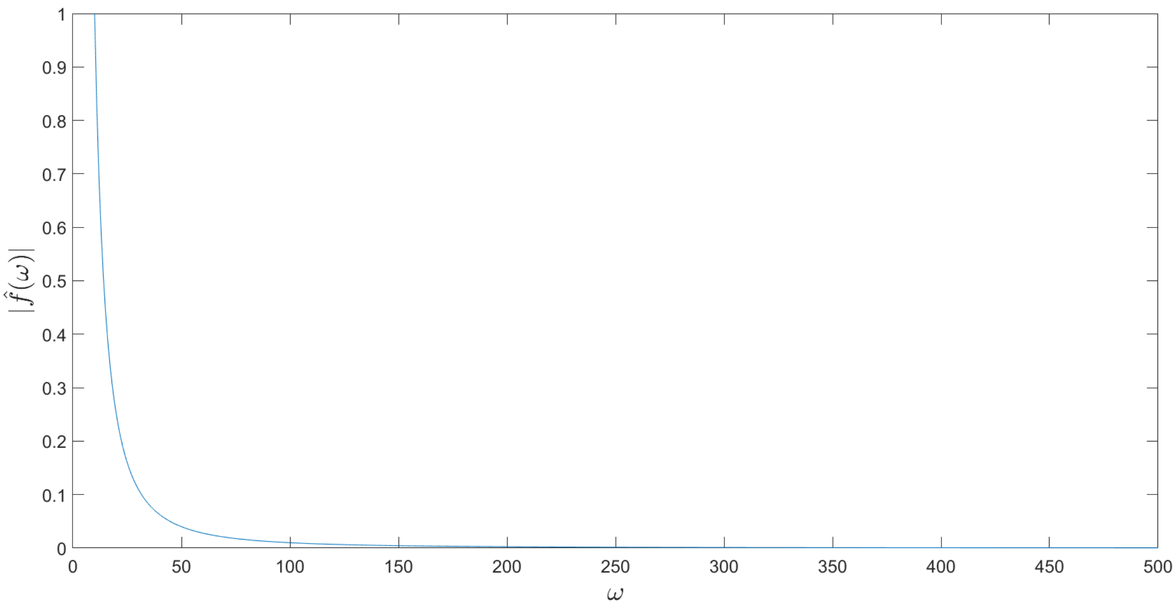

Figure 2. Visualization of the upper limit of Fourier coefficients. Value of p has strong impact on speed of decay.

2.4 Gaussian white noise

White noise is a conceptual model used to describe effects of multiple uncorrelated random variables. Gaussian white noise is a specific white noise model where the noise is normally distributed. A vector of measurement errors $\mathbf{e} = (e_1, .., e_N)$ is Gaussian white noise where each $e_n \sim N(0, \sigma^2)$.

Decomposing a time series into frequency components gives a distribution of the spectral density among different frequencies. These spectral density values form a power spectrum for the time series. Power spectrum of band-limited Gaussian white noise is

$$P = \begin{cases} \sigma^2, & |\omega| < B \\ 0, & \text{otherwise,} \end{cases} \tag{11}$$

where B denotes the band-limiting frequency and σ^2 is the noise variance. From the theoretical power spectrum, the total power of the noise P_T can be calculated as

$$P_T = 2B\sigma^2. \tag{12}$$

2.5 Distortion caused by noise

The distortion can be quantified as the total difference between the original noisy function and the modified function after noise reduction,

$$\epsilon_T^2 = ||f(t) - f_{\text{mod}}(t)||_2^2 = \int_{-\infty}^{\infty} (f(t) - f_{\text{mod}}(t))^2 dt, \tag{13}$$

where f_{mod} denotes modified function. Using Plancherel's identity, the difference can be given in frequency domain as

$$\epsilon_T^2 = \int_{-\infty}^{\infty} (\hat{f}(\omega) - \hat{f}_{\text{mod}}(\omega))^2 d\omega, \tag{14}$$

where $\hat{f}_{\text{mod}}(\omega)$ denotes Fourier transform of the modified function. To estimate effectiveness of the noise reduction model ϵ_T^2 can be compared to theoretical power of the noise if the variance in Eq. (12) is known. Optimal method should result in

$$\epsilon_T^2 \approx 2B\sigma^2. \tag{15}$$

In practice however, the noise variance can be unknown and power of the noise can only be estimated. An alternative way of computing effectiveness of the noise reduction is to measure difference between modified Fourier transform and original function.

$$d = ||f_{\text{mod}}(t) - f_{\text{orig}}(t)||_2^2 = \int_{-\infty}^{\infty} (\hat{f}_{\text{mod}}(\omega) - \hat{f}_{\text{orig}}(\omega))^2 d\omega, \tag{16}$$

where f_{orig} denotes the background function without the noise. With the discrete Fourier transform, these integrals are calculated using summations.

2.6 Noise reduction methods

In the following Sections, we introduce three approaches for noise reduction which are justified by the properties discussed in Section 2. The methods are called amplitude thresholding, frequency cut-off, and coefficient scaling. The latter two similar as they both are based on filtering out higher frequency fluctuations.

After using FFT to transfer the data to the frequency domain, the results are obtained by suitable modification of the Fourier transform coefficients. Zero-padding the original data with an equal amount of points. Figures 1 and 2 show that Fourier transforms of smooth and periodical background functions have distinct properties which should be considered when choosing the correct noise reduction method. Information on which components are vital for the background process and which of them are caused by noise helps us determine the properties of the original function ultimately giving us the approximation without the noise.

Amplitude thresholding

Because of the power spectrum of white noise (11), we can choose an amplitude threshold below which all noise components should remain under. Using information of the Fourier transform of trigonometric functions, dominant components can be distinguished from the noisy data. After filtering out weaker components we can compute IDFT of the remaining coefficients resulting in a filtered signal in the time domain. The method should work exceptionally well on signals with signal consisting of trigonometric functions as shown in Figure 3.

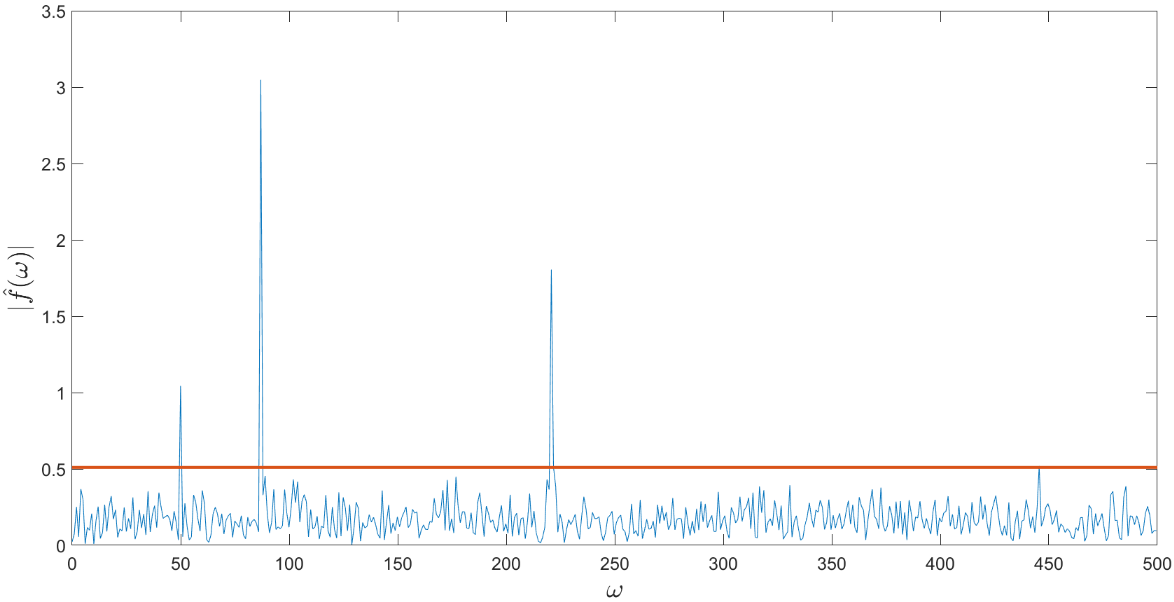

Figure 3. Visualization of amplitude thresholding method. Threshold shown with red line on frequency domain. Three trigonometric components can be distinguished from the transform.

Frequency cut-off

Based on the decay of the Fourier coefficients (10), most of the information for smooth background functions should be located in the low frequency values. Deficiency of this method comes apparent if original data has high frequency variation. This method will overlook any coefficients above chosen threshold and smooth out the curve as shown in Figure 4.

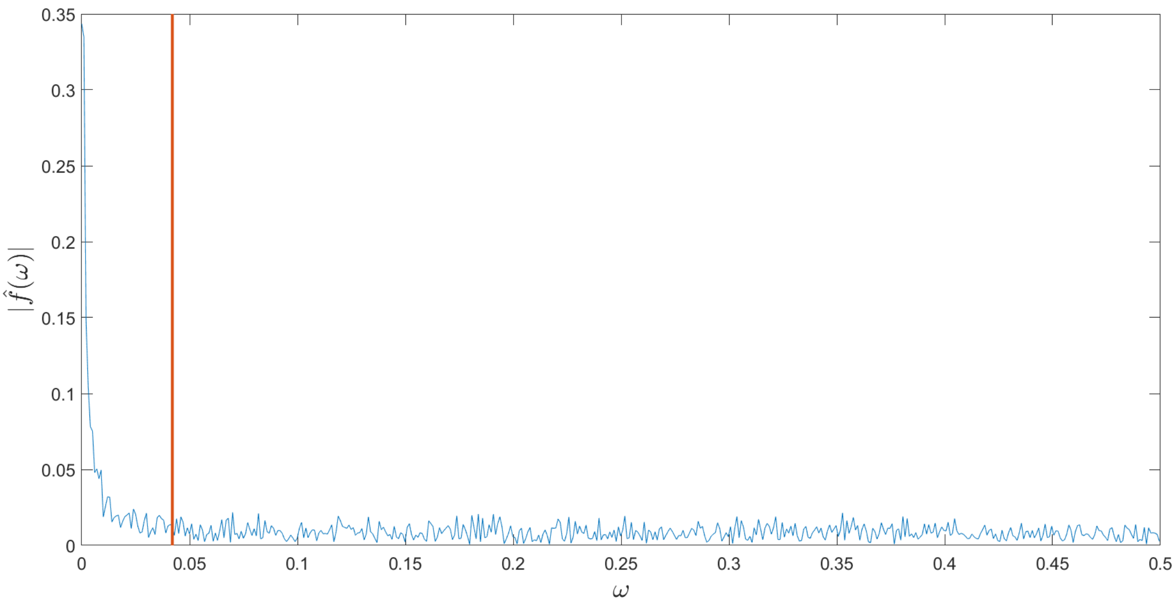

Figure 4. Visualization of frequency cut-off method. Cut-off frequency shown with red line on frequency domain.

Coefficient scaling

Decay of the Fourier transform (10) suggests that the Fourier transform of a continuous function should follow a certain shape depending on smoothness of the function. With this, it can be assumed that $\hat{f}(\omega) \to 0$, when $|\omega| \to \infty$. We scale the Fourier transform coefficients with a decaying function to smooth out high-frequency components. Cauchy distribution was chosen for its resemblance to the polynomial form of the decay.

Density function of the chosen unnormalized Cauchy distribution can be given as

$$f_C(x) = \frac{\gamma^2}{x^2 + \gamma^2}, \tag{17}$$

where γ is the shape parameter of the Cauchy distribution.

3 DATA

3.1 Synthetic data

We apply the discussed noise reduction methods to two synthetic data sets and IBM stock data. The first data set consists of two trigonometric functions, given by

$$f_{\text{orig}}(t) = A_1 \sin(2\pi\omega_1 t) + A_2 \cos(2\pi\omega_2 t), \tag{18}$$

where ω_1 and ω_2 denote the frequencies of the trigonometric components with A_1 and A_2 being the corresponding amplitudes. Parameter values are detailed in Table 1. White noise was generated with variance of $\sigma^2 = 1$. Data points were generated using 1000 linearly spaced points with $t \in [0,1]$.

A_1	$\frac{1}{3}$
A_2	$\frac{1}{2}$
ω_1	19
ω_2	29

Table 1. Parameters used to produce the first data set.

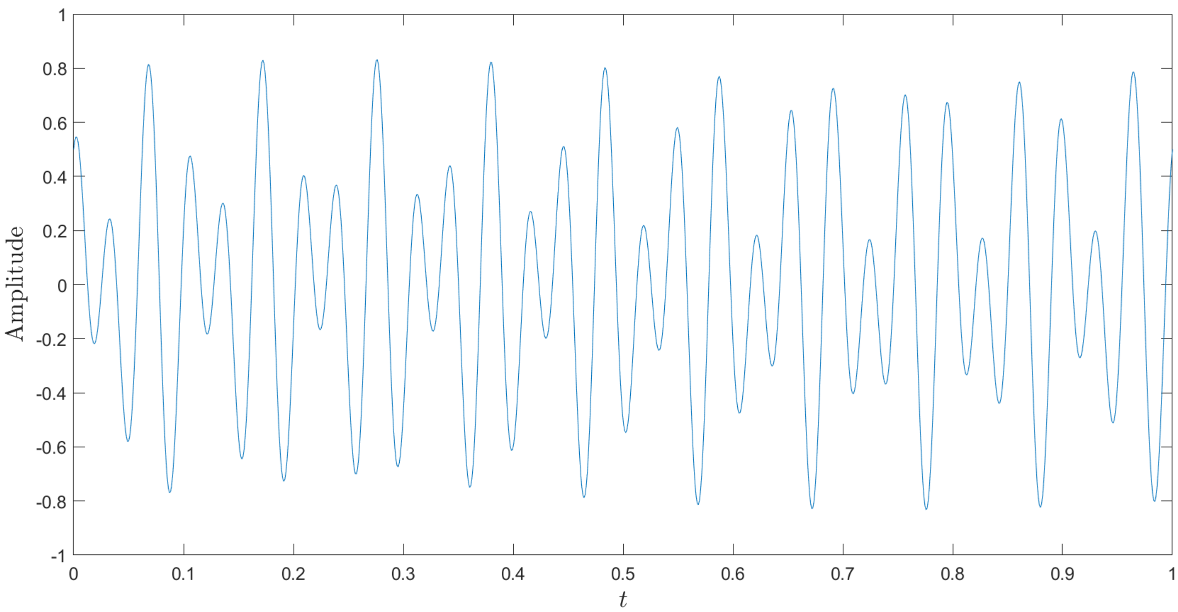

Figure 5. In blue, the synthetic function $f_{\text{orig}}(t)$ consisting of two trigonometric functions used to generate the first data set.

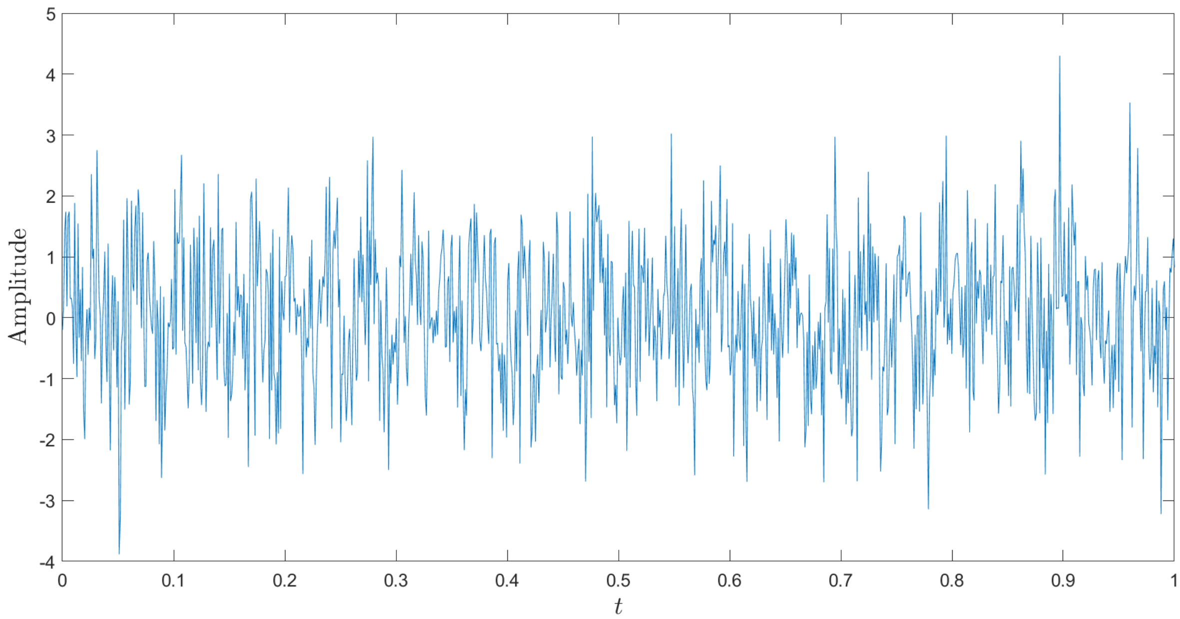

Figure 6. In blue, the data set consisting of two trigonometric functions corrupted with Gaussian white noise. The underlying trigonometric functions are hardly distinguishable.

Second data consists of a quartic polynomial, given by

$$f_{\text{orig}}(t) = a_1 t^4 + a_2 t^3 + a_3 t^2 + a_4 t, \tag{19}$$

where parameters a_1, a_2, a_3, and a_4 denoting coefficients for the variables. Parameter values are detailed in Table 2. White noise was generated with variance of $\sigma^2 = 1$. Data points were generated using 1000 linearly spaced points with $t \in [0, 1]$.

a_1	-70
a_2	120
a_3	-60
a_4	12

Table 2. Parameters used to produce the second data set.

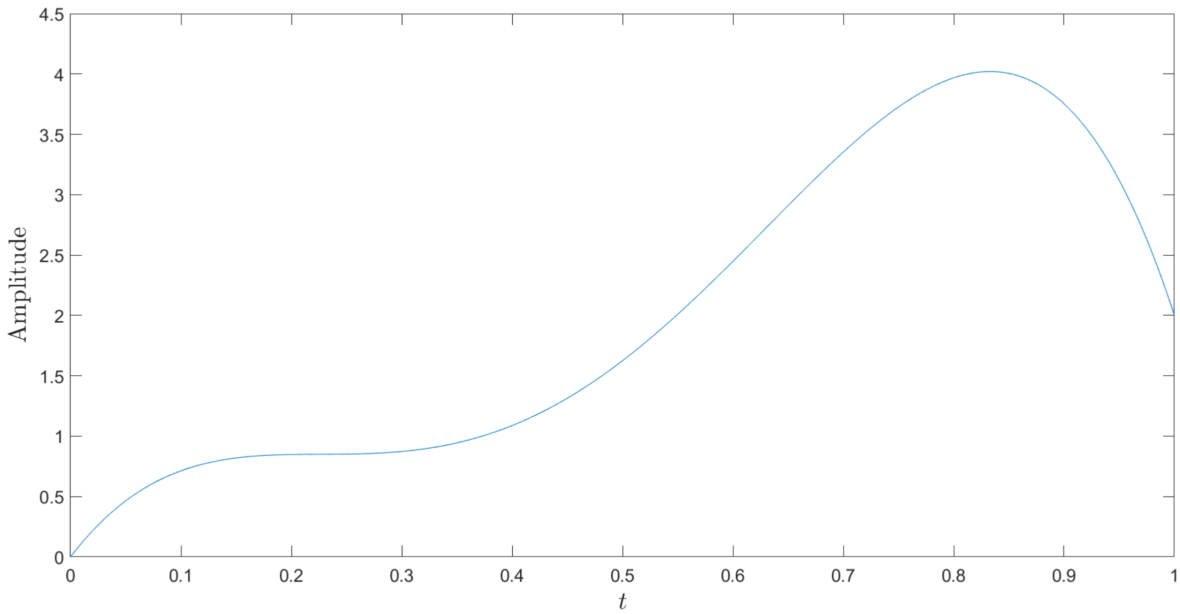

Figure 7. In blue, the polynomial $f_{\text{orig}}(t)$ used to create the second data set. Parameters were chosen so that curve would have some variation.

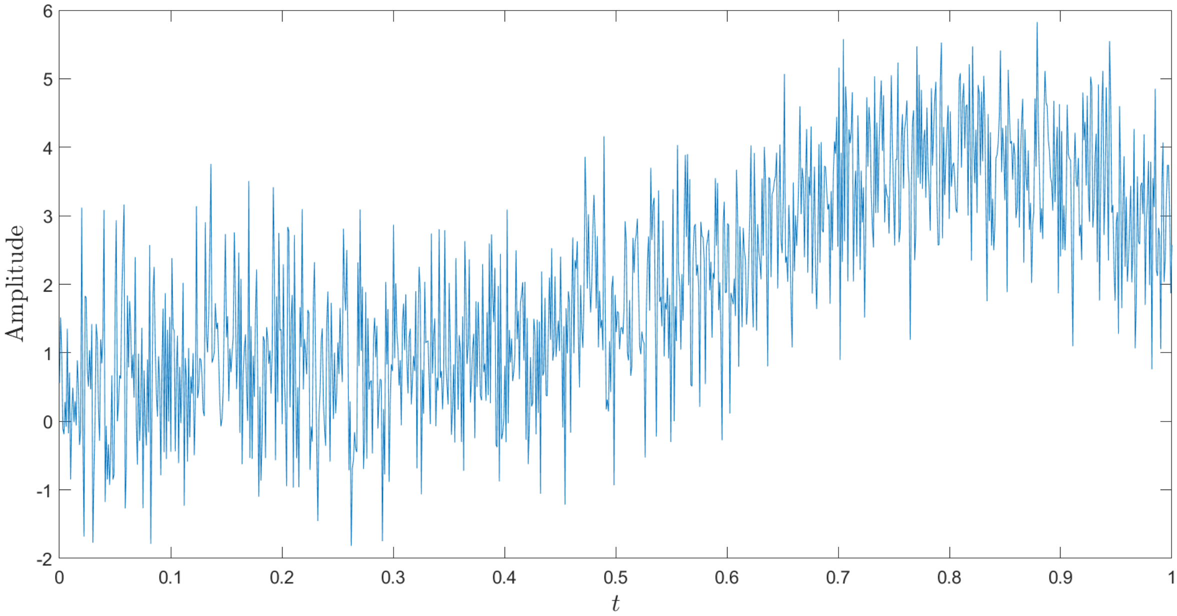

Figure 8. In blue, the polynomial data set corrupted with Gaussian white noise. The original function $f_{\mathrm{orig(t)}}$ can still be visually observed.

3.2 IBM stock data

As a practical case, we apply the methods discussed in Section 2.6 to IBM stock price data. Data for the year 2021 was downloaded using Yahoo Finance Python application programming interface. The data consists of 1761 stock prices recorded with 1 hour interval during the open hours of the Nasdaq stock market shown in Figure 9. Note that while the data set is not contaminated by errors per se, the data is stochastic or, in other words, consists of random fluctuations. We aim to apply the noise reduction techniques to estimate a smooth mean function of the stochastic data, analogous to noise removal.

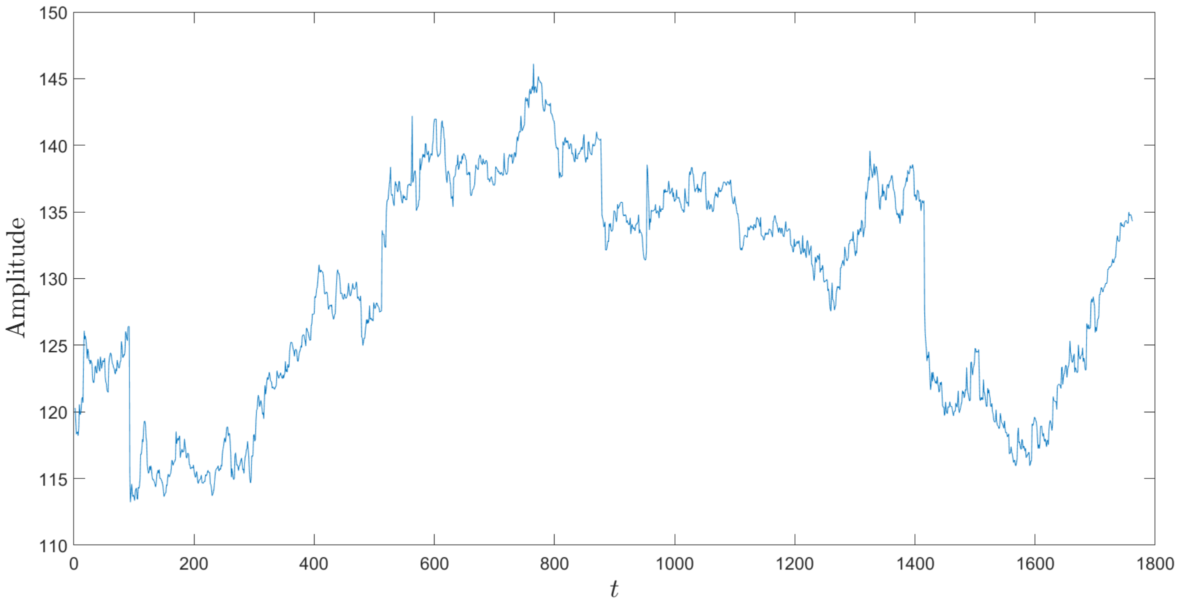

Figure 9. Recorded IBM stock prices from year 2021 shown in blue.

4 RESULTS

4.1 Data set 1

First synthetic data set shown in Figures 5 and 6 was analysed using amplitude thresholding. The threshold was set to $|\hat{f}(\omega)| > 0.2$ which eliminates all but two dominant components according to method shown in Figure 3. Approximation was created using IFFT on the modified frequency spectrum.

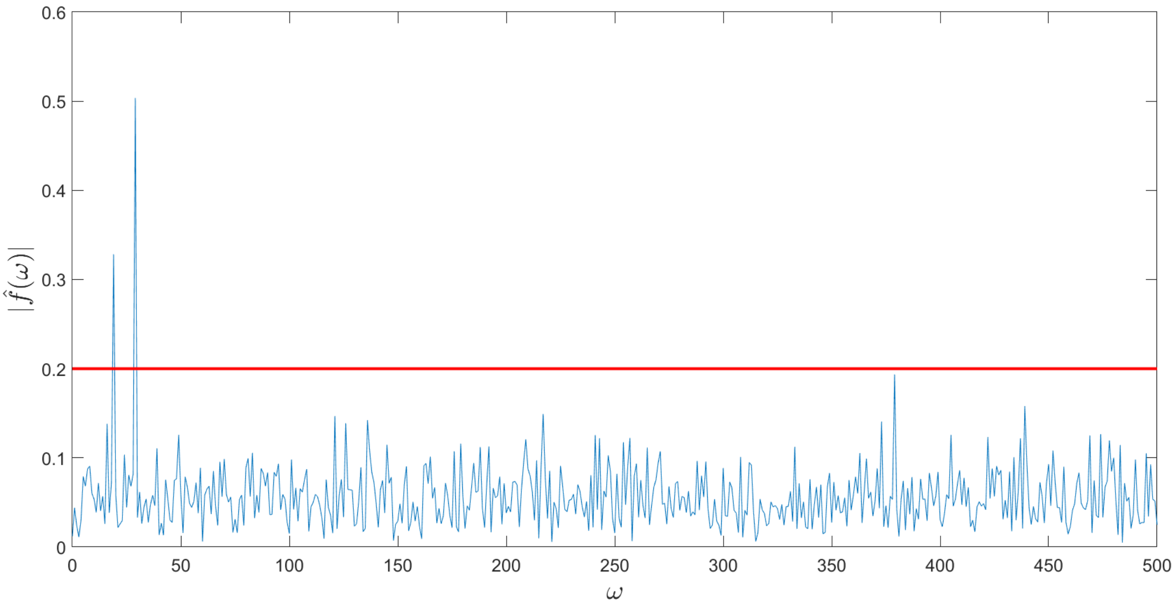

Figure 10. Absolute values of the Fourier transform coefficients for the trigonometric data set with the chosen amplitude threshold. Absolute values of the Fourier coefficients are shown in blue and the chosen amplitude threshold with red line. Two frequency spikes can easily be distinguished from the noise.

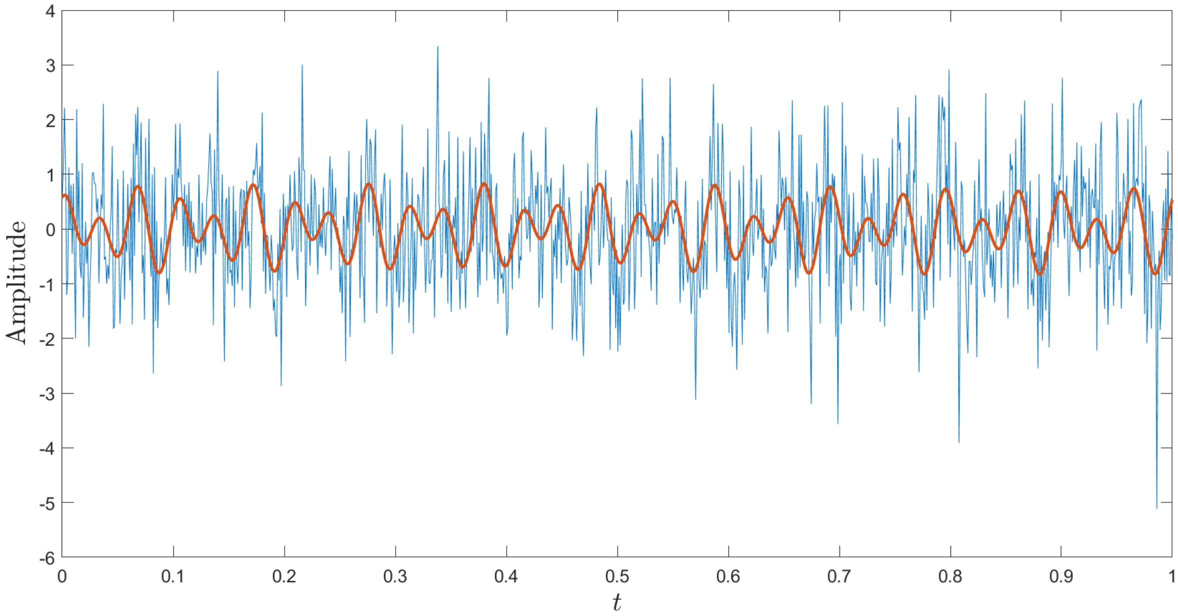

Figure 11. Original data shown with blue line and data after noise reduction with red line.

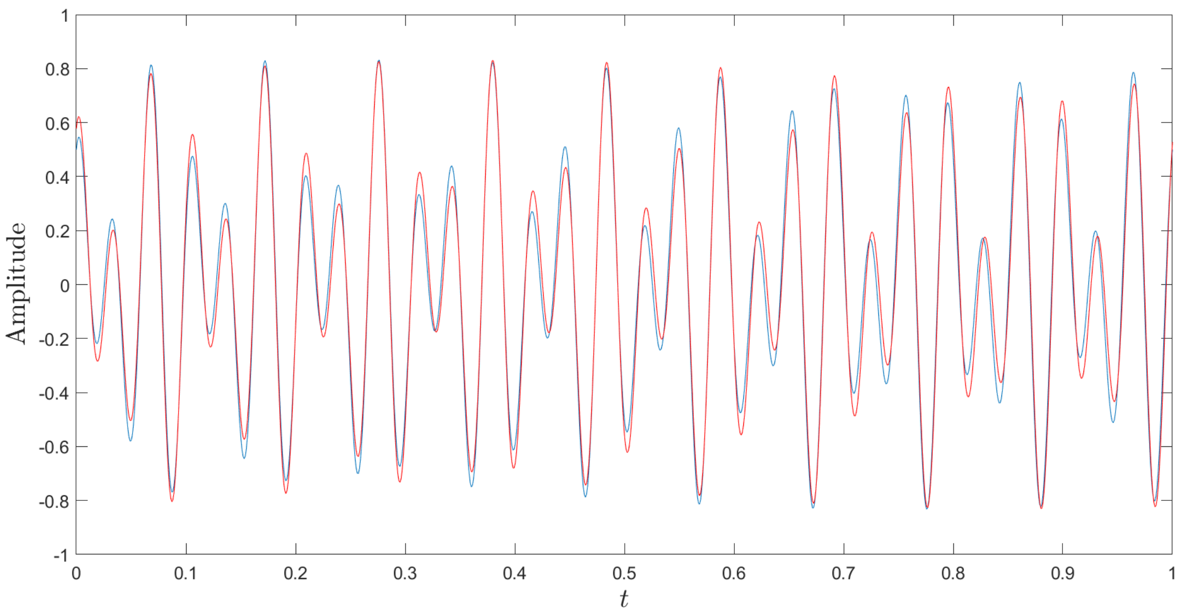

Figure 12. Original function $f_{\text{orig}}(t)$ shown in blue and function produced from noisy data shown in red. Functions produce similar curve.

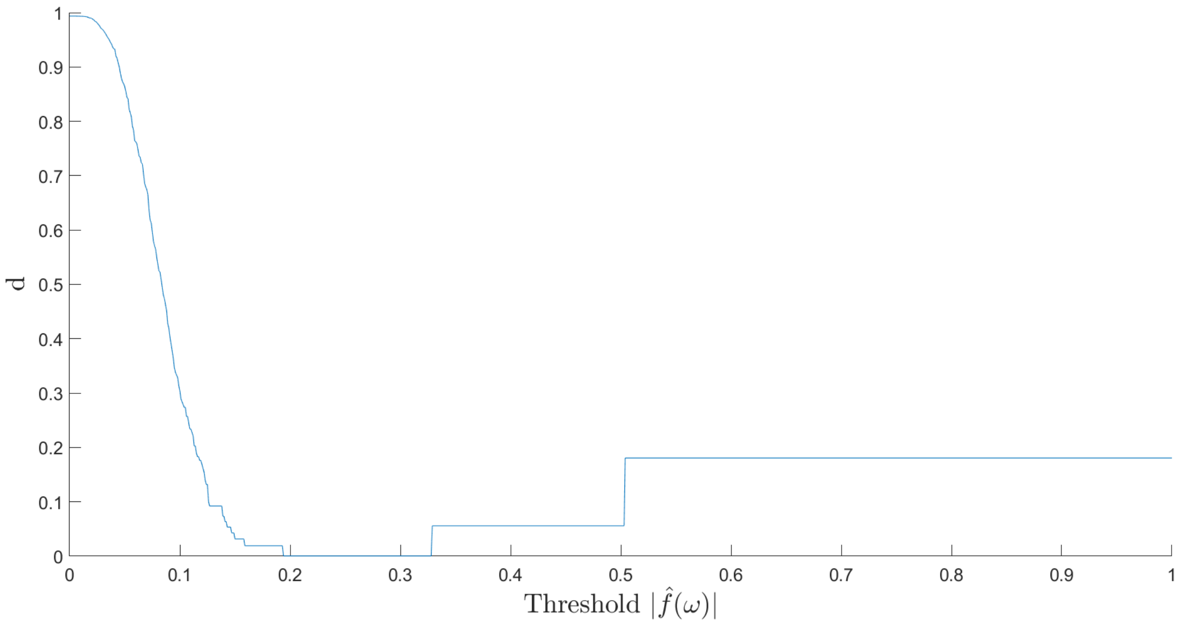

Figure 13. Total difference between data after noise reduction and original function. Difference function reaches the minimum value when $\omega \in [0.2, 0.33]$.

4.2 Data set 2

Second data set, shown in Figures 7 and 8, featured smooth curve which was analysed using two low-pass filters. Fourier transform of the data set was modified with the frequency cut-off and coefficient scaling techniques.

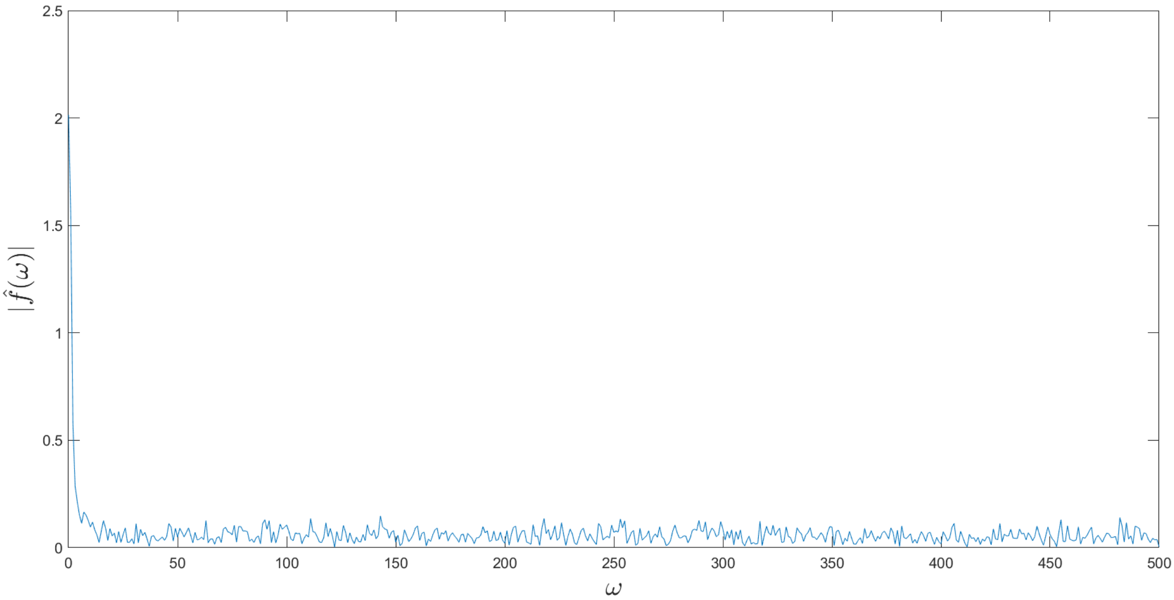

Figure 14. Fourier transform of the second data set. A fast decaying curve stands out from the background noise close to origin.

To analyse the second data frequency cut-off method described in Section 2.6 was used. By visual inspection, the best limit frequency was chosen to be $\omega = 7$. The data was transformed to the frequency domain using FFT and frequencies $\omega \in [8, 500]$ were set to zero.

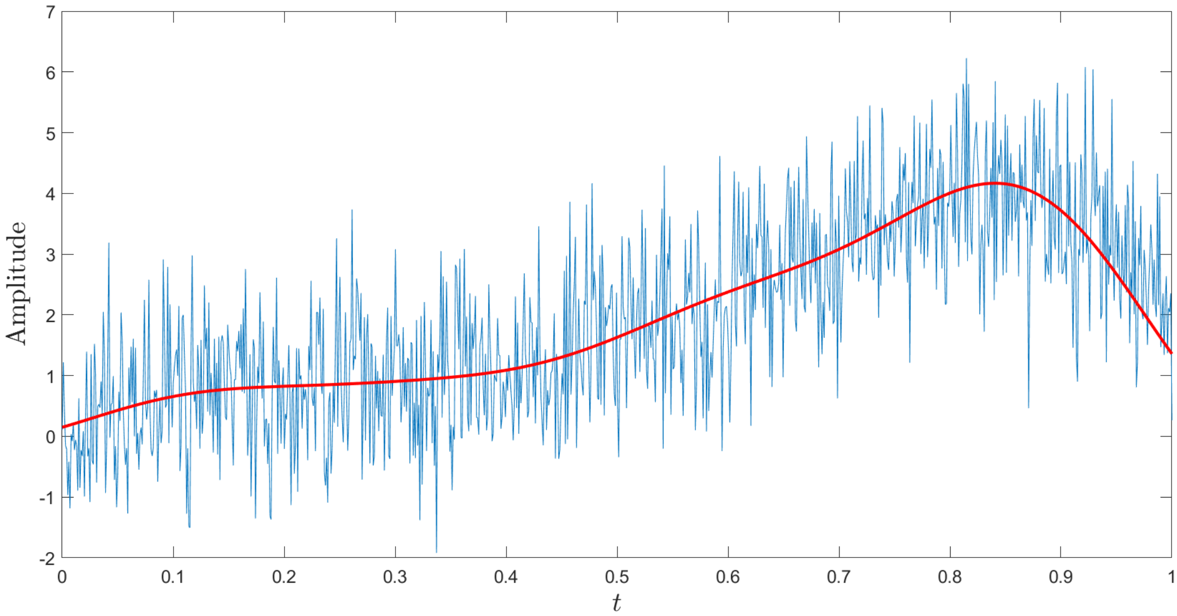

Figure 15. The original data shown in blue and the resulting approximation result from filtering high frequency components shown in red.

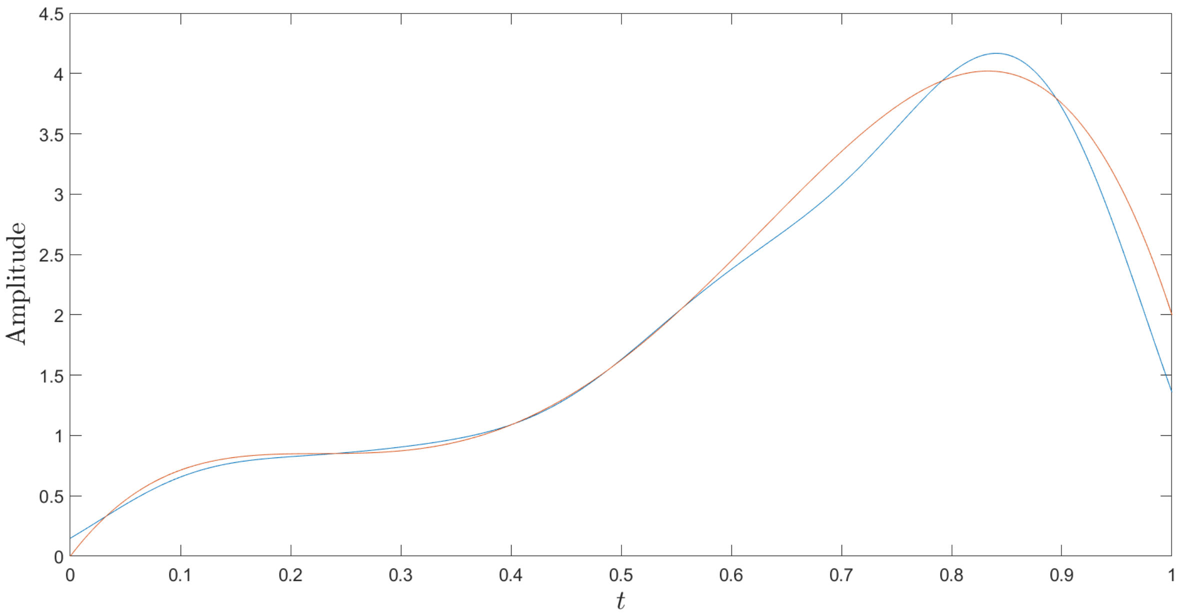

Figure 16. Original function used to create data shown with red line and approximation result in blue. The approximation seems to visually fit well to the original function.

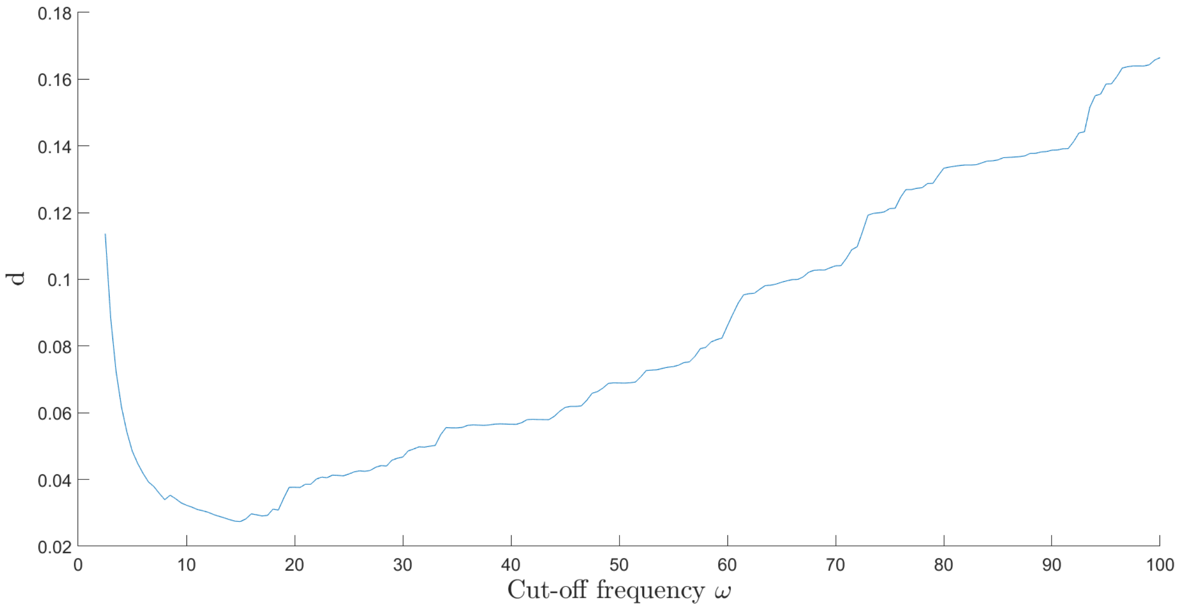

Figure 17. Difference between the original function and the approximation calculated with different cut-off frequencies. Minimal value of this function differs from the visually observed best-fitting value for cut-off frequency.

For the other low-pass filter, a parameter value of $\gamma = 0.013$ was chosen visually as the best-fitting. The data was transformed to the frequency domain and a Cauchy distribution was used to scale down coefficients to model the decay of the Fourier components. The approximation was created using IFFT on the modified coefficients.

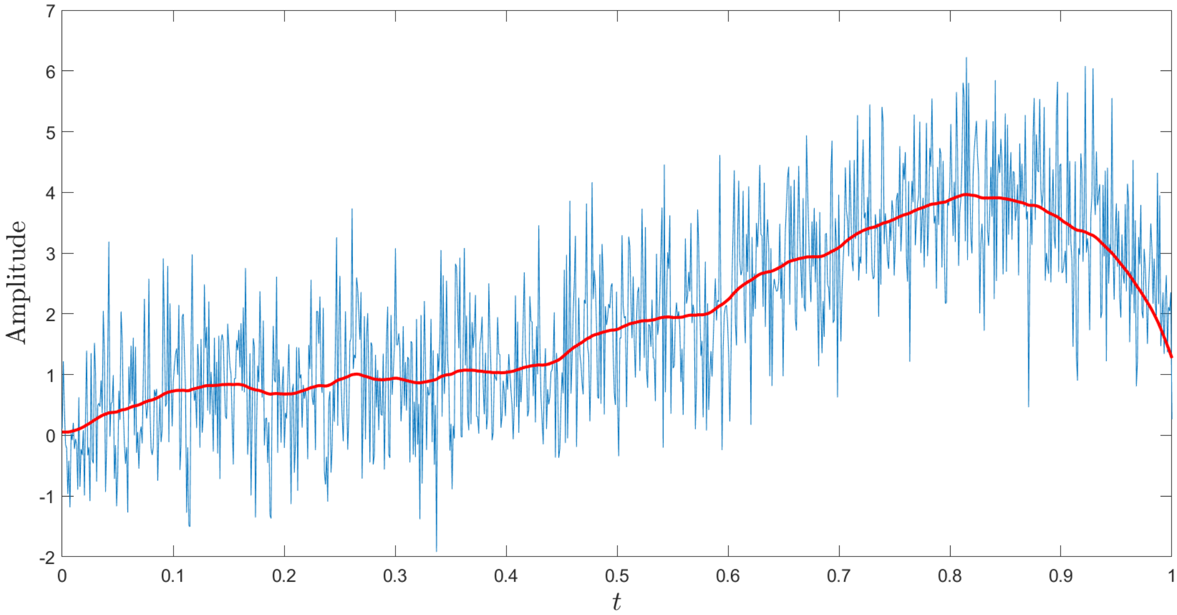

Figure 18. Original data shown in blue and approximated function in red.

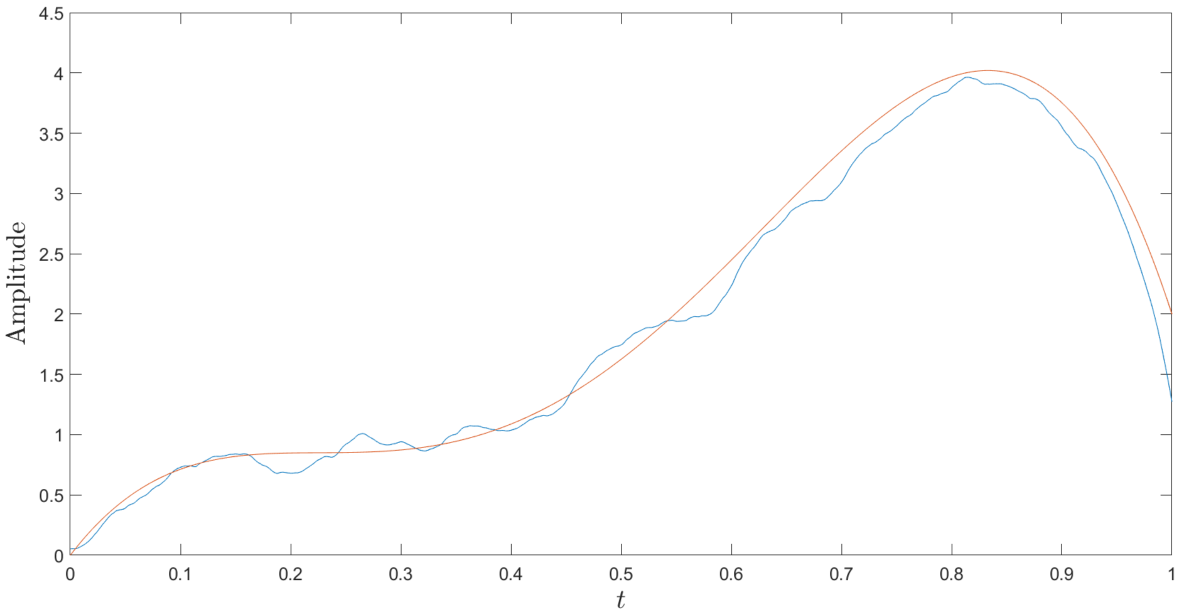

Figure 19. Approximated function shown with blue line and the original function used to create the data shown with red line.

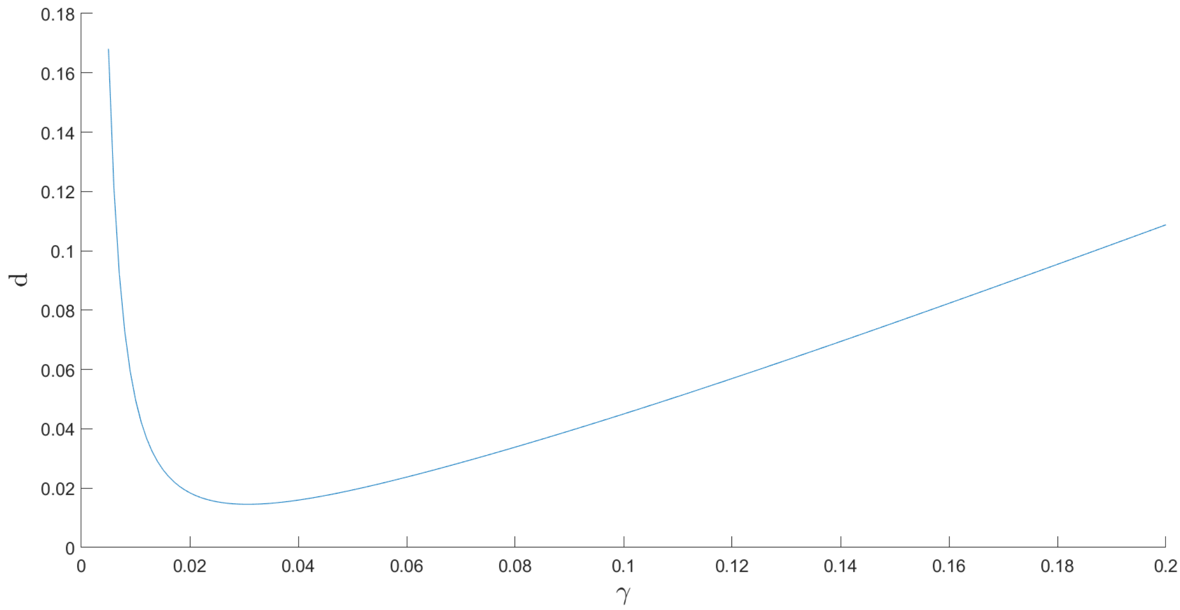

Figure 20. Difference between the original function and the approximation calculated with different γ-values.

4.3 Method comparison

In addition to visually selecting the best-fitting parameters for the filters, d-values were calculated with different parameter values. Parameters that produced the lowest difference were chosen and compared to methods corresponding values. Tables 3 and 4 also include total power of the reduced noise compared to the theoretical value.

DATA 1	d	$\lvert e_T^2 - 2B\sigma^2 \rvert$
Amplitude thresholding	$4.3 \cdot 10^{-4}$	$2.8 \cdot 10^{-3}$
Frequency cut-off	0.078	0.020
Coefficient scaling	0.057	0.15

Table 3. First data set analysed using the three methods. Parameters for the filters were chosen as the best-fitting to the original function.

DATA 2	d	$\lvert e_T^2 - 2B\sigma^2 \rvert$
Amplitude thresholding	0.061	0.011
Frequency cut-off	0.039	$2.5 \cdot 10^{-3}$
Coefficient scaling	0.027	0.037

Table 4. Second data analysed using different methods. Parameters for the filters were chosen as the best-fitting to the original function.

IBM stock data, shown in Figure 9, was analysed using all three noise reduction methods. Parameters for the filters were chosen as the visually best-fitting.

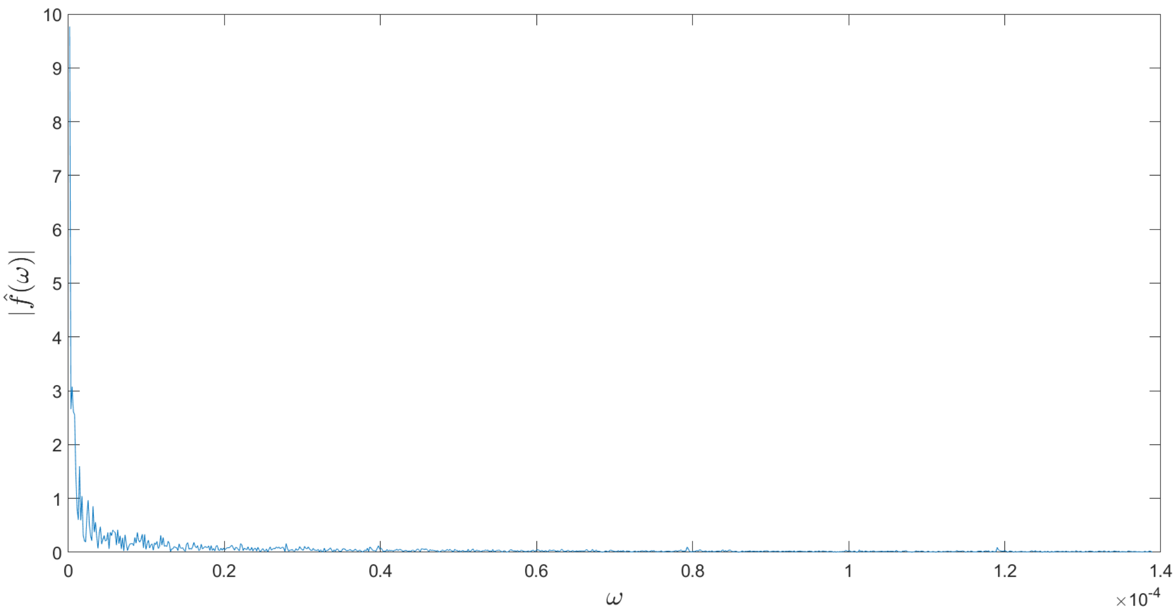

Figure 21. Fourier transform of the IBM stock data. Coefficient for the constant component $\omega = 0$ is removed to better visualize decay.

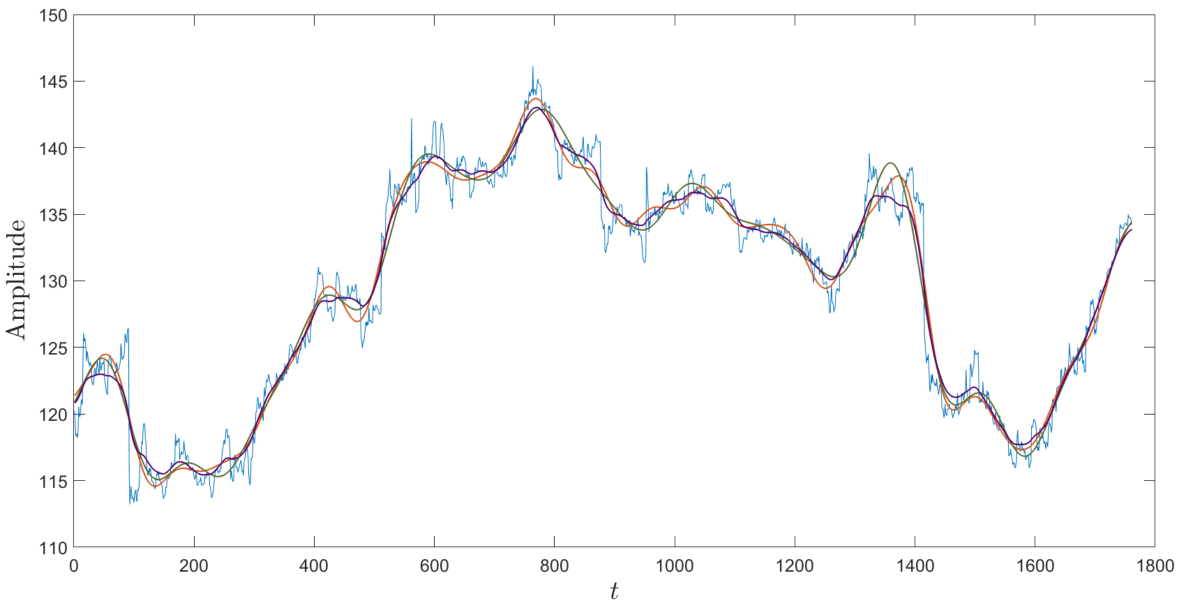

Figure 22. Original stock data is shown in blue. The approximations obtained using amplitude thresholding, frequency cut-off, and filtering are shown in red, green, and dark blue.

5 DISCUSSION

Using synthetic data as testing data made it possible to calculate the accuracy of each approximation and compare the noise removal approaches. Knowing the popularity and previous successes of the Fourier transform in similar applications made the positive results to be expected. Methods were tested on the both synthetic data sets and results were documented in Tables 3 and 4. All techniques used performed at the expected level and produced good approximations for the background function.

Amplitude thresholding was the most effective tool of the comparison. Figures 10 – 13 show results of methods used on the first data set. A background process that can be described using periodic trigonometric functions, the results show that the noise was eliminated accurately without losing critical information. In real applications, hardly any process can be described with perfect trigonometric functions which can make method more inaccurate. Identifying frequency components is used widely in signal processing and is one of the main applications of the Fourier transform. With a smooth background process, amplitude thresholding was not as effective as other methods used in comparison.

With the synthetic data set consisting of trigonometric functions, the amplitude thresholding method produced credible results with the obtained result being almost identical to the original function. One of the benefits of using this method was that threshold can be chosen without any knowledge of the background process. In Figure 10, two frequency component clearly stand out from the white noise. Locations of these spikes in the ω-axis seems to closely match to the frequency of trigonometric components. Amplitudes of these spikes was also approximately equal to coefficients for these components. White noise affects all coefficients in the frequency domain, including ones that are used to create the data resulting in the approximations almost never being perfect.

Including only frequency components below chosen value proved to be an efficient method for noise reduction for smooth functions. Figures 15 – 17 show results of the method used on the second data set. The Fourier transform of the synthetic smooth function in Figure 14 followed principles of the Fourier component decay which suggest that the most consequential components are located at low frequencies. Using only a small number of components resulted in an accurate approximation.

Other low-pass used in the comparison was also based on Fourier component decay (10). Figures 18 - 20 show results of method used on the second data set. In paper, filter outperformed other methods used but results show the modified DFT had higher frequency components that reduced the smoothness of the curve. The Idea of scaling the Fourier coefficient to remove effects of noise turned out to reduce the smoothness of the approximated curve and resulted in losing some information on the background process. In the comparison, noise had minimal effect on low-frequency components and simply removing high-frequency components proved to be most efficient way to remove noise without losing vital information on the background process and retaining the smoothness of the approximation.

Second data set was more challenging than the first one due to the absence of periodical trigonometrical components. To improve the results given by these filters, the sampling frequency could be increased to have more data points for the Fourier transform. White noise would lose impact on frequency spectrum and coefficients of the original function would have more impact on the result. Variance of the white noise was also significant compared to variation of original function which can make analysis more imprecise. Larger padding also increases the number of frequency components which can sharpen the approximation.

When selecting parameters for the filtering, visual observation gave better approximations than choosing parameters based on difference. Using a minimal amount of frequency components gives smoother curve without higher frequency variation. Visual selection of parameters was based on testing values that produced a reasonable result while using as few coefficients as possible. Minimizing the difference between removed noise components and the theoretical power of the noise gave similar results to visual selection.

Fourier transform of the IBM stock data suggests that the background process follows the decay properties of smooth curve with background noise creating constant variation along the power spectrum. Some of the variation in the data seemed periodical which justified usage of the amplitude thresholding method. Approximated function produced by this method fitted well to the periodic fluctuations in the data but some parts that appeared to be smoother in the original data had some variation due to higher frequency components included in the approximation.

Three methods used in the comparison were tested on the synthetic data in Figure 22. As the amount of data points increases, the decay of Fourier components stands out more. Synthetic data sets had larger amount of noise which made the low-pass filters more imprecise. Reducing the amount of noise and decreasing the sampling frequency sharpens the approximation for low-pass filters. The frequency cut-off method produced smoother curve than the method that was based on scaling down coefficients. The previously underperforming coefficient scaling method clearly produced the most accurate approximation for the process. Approximation from the frequency cut-off method followed a similar overall shape but some of the vital higher frequency components were not included and some information on the original function was lost.

6 SUMMARY

The noise reduction techniques were successfully implemented and the methods performed as expected. Discussed theory was proven to be functional in light of the results for the synthetic data sets and the mean function was, by visual inspection, successfully constructed. For the data sets considered, amplitude thresholding proved to be most efficient tool for noise reduction. Low-pass filters underperformed with lower sample sizes but with more frequent sampling on stock data both filters improved their performance.

References

[1] Michael T Heideman, Don H Johnson, and C Sidney Burrus. “Gauss and the history of the fast Fourier transform”. In: *Archive for history of exact sciences* 34.3 (1985), pp. 265–277.

[2] Brad G. Osgood. *Lectures on the Fourier transform and its applications*. 3rd ed. Vol. 33. 10. 2019.

[3] Zhang Yonglin and Zhang Jiafan. “Numerical simulation of stochastic road process using white noise filtration”. In: *Mechanical Systems and Signal Processing* 20.2 (2006), pp. 363–372.

[4] M Sathya and DSP Victor. “Noise reduction techniques and algorithms for speech signal processing”. In: *International Journal of Scientific & Engineering Research* 6.12 (2015), pp. 317–322.

[5] Xuebao Wang, Gaoming Huang, Zhiwen Zhou, and Jun Gao. “Radar emitter recognition based on the short time Fourier transform and convolutional neural networks”. In: *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. 2017, pp. 1–5.

[6] Feiran Yang, Ming Wu, and Jun Yang. “Stereophonic acoustic echo suppression based on Wiener filter in the short-time Fourier transform domain”. In: *IEEE Signal Processing Letters* 19.4 (2012), pp. 227–230.

[7] Nasser Tairan, Saudia Subash, and Justin Varghese. “Fourier transform based windowed adaptive switching minimum filter for reducing periodic noise from digital images”. In: *IET Image Processing* 10 (2016).