# Approximate uncertainty analysis principle: A robust non-linear fitted curve extracting from incomplete frequency information


## ABSTRACT

The parametric non-linear curve fitting algorithms require to determine the start points of model parameters. However, the estimation of parameter is error-prone, and incorrect parameters often destroy the quality of fitted curves. In this paper, we turn to the non-parametric non-linear fitted curve problem: given the sequence of the measurement data $x$ of length $n$, extract the $k$ most important features of measurement data, and use these core features to carry out a non-linear fitted curve without any guess of initial parameter values. This method can be widely applied for uncertainty analysis, and the quality control of data in science and engineering domain.

We come up with a new algorithm for this problem. The algorithm exploits the techniques from signal processing domain, the Fourier transform. Unlike the typical parametric fitted curve methods, our algorithm does not require complex guess of initial parameters. Furthermore, we work out two new estimators for the estimation of large Fourier coefficients. The resulting algorithm is structurally similar with the best-fit outcomes in different synthetic, and practical data. As a result, we can extend the sparsity of $k$ to meet the requirements of different applications.



## Keywords

ACM proceedings, LaTeX, text tagging :


## 1. INTRODUCTION

It is challenging to estimate model parameters of a non-linear curve fitting model, in particular when one model contains many unknown parameters [16, 15]. Parametric curve fitting methods need to determine the start points of parameters before stepping into regression analysis tasks. A good estimation of initial values requires the deep knowledge of data or the guidance of human. Without any guidances, and clues, the estimation of parameters becomes error-prone. Note that the inadequate initial values of model parameters often slow down the converge speed of a curve fitting model. What the worst is that incorrect values of model parameters would destroy the results of fitted curves (figure 1). Thus, the parametric methods make non-linear curve fitting algorithms become vulnerable.

The goal of this paper is to reduce the complexity of the guess of initial parameter values in non-linear curve fitting algorithms. However, all iterative curve fitting methods such as Levenberg-Marquardt algorithm[11], and Gaussian-Newton method require to determine initial values of model parameters. Fortunately, we found an opportunity from Fourier transform, and leverage it to carry out a new non-linear curve fitting algorithm. In general, Fourier transform claims that the measurement data can be reconstructed from the combination of the sequence of exponential functions, and amplitudes ordered by frequency. In addition, Fourier transform can handle aperiodic signal samples, and does not require any guess of initial parameter values.

The amount of data in a typical exact frequency analysis is proportional to its input size, which is $\theta(n)$. However, in many applications, most of amplitudes in Fourier frequency domain are often small and are close to zero. For instance, a typical 8x8 block in a video frame, 89% of Fourier coefficients are negligible[7, 3]. Actually, image and video processing often leverage this sparsity to carry out efficient compression schemes. Other applications using sparse rationale include principle component analysis (PCA)[12], random projection[13], and compress sensing[5]. Inspired by the above sparse applications, the $k$ largest Fourier amplitudes stands for the core features of the measurement data, and we want to leverage these few Fourier coefficients to carry out a non-linear fitted curve.

Formally, the fitted-curve problem is that a given measurement data $x, x \in \mathbb{R}$, whose results of the fitted-curve is $\hat{x}$. The output of our algorithm is one approximate fitted-curve $\hat{x}'$, and that satisfies the following guarantee:

$$\left|\left|\hat{x} - \hat{x}'\right|\right|_2 \leq \epsilon, min\left|\left|x - \hat{x}\right|\right|_2$$

where $\epsilon$ is the approximation factor, and the best-fit non-linear fitted-curve is the minimization of the deviation between the fitted-curve $\hat{x}$ and measurement data $x$. In contrast with best-fit curve, the $k$-sparse approximation only uses $O(k)$ number of large Fourier transform coefficients to construct one approximate fitted-curve, and neglect the rest of small ones. Since the data is often noisy, the results from a thorough analysis of the big data may not be worth the cost for completeness. Thus, an approximate clue from big data is enough for us to obtain perfect answers to make a good decision similar to results from a thorough analysis.



*WOODSTOCK* '97 El Paso, Texas USA

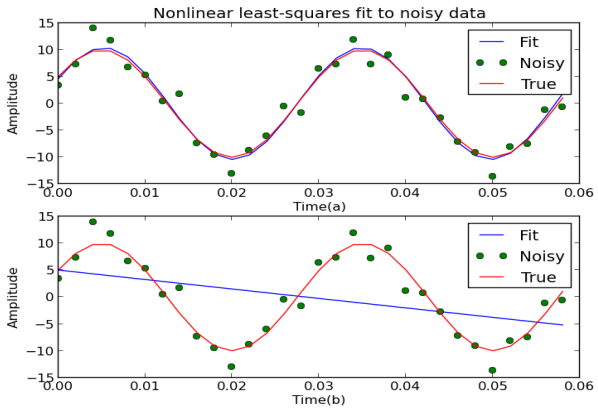


**Figure 1: Nonlinear least-squares fit to noisy data** The true curve is composed of $y_{true} = A \cdot sin(2\pi kx + \theta)$, $[A, k, \theta] = [10, 1.0/3e-2, \pi/6]$. The initial values of the fitted curve in figure (a) is $[A, k, \theta] = [8, 1.0/3e-2, \pi/3]$, and the one in figure (b) is $[A, k, \theta] = [10, 1/2, \pi/3]$. We can see the fitted curve in figure (b) is failed with the bad initial values of model parameters.

**Our contribution:** Our non-parametric fitted curve algorithm serves three main challenges. First, to reduce the erroneous threats caused by the guess of initial parameter values. Inspired by the sparse coefficients in Fourier frequency domain, and we add this feature in the construction of fitted-curve, and carry out a simple, practical and scalable non-linear fitting curve algorithm. Second, to summarize the distribution density of the measurement data from the sampling data, and use the approximate distribution density scheme to infer the frequency with the large Fourier amplitude. This approximation allows our method to be scalable for the size of data. Third, it is an economical approach for the inference of large Fourier amplitudes. This approximation enables us to focus on the trade off approximation error $\epsilon$ between the measurement data and its approximation.

This paper is organized as follows. Section 2 discusses the related works and background materials. Section 3 discusses our non-parametric non-linear curve fitting algorithm. Section 4 accounts for two estimator in our fitted curve method. Section 5 shows the experiment results, and section 6 makes a conclusion.

## 2. BACKGROUND AND RELATED WORKS

Fitted curves are often used in statistical inference for uncertainties, errors hidden in measurement data such as the detection and defense of cyber attack, diagnose violations of large scale systems, and quality control of genomic, and DNA sequence analysis. In some cases, an approximate fitted-curve is sufficient to pick out most anomalies. For instance, the values of anomalies are far away from the mean of the data. The curve fitting method can be seen as a numerical optimization task. The curve fitting problem starts with the experimental data that consists of data tuple ($x_i$, $y_i$). The objective function of the curve fitting method is to minimize the error sum of squares Q, which is given by

$$min \quad Q = \sum_{i=1}^{n} |x_i - y_i|$$

Sparse Fourier transform[9, 6, 7] showed the way to use Gaussian and Dolph-Chebyshev filters to carry out an approximate sparse distribution in frequency domain. Thus, we want to carry out a non-linear fitted-curve from partial Fourier frequency amplitudes without complicated guess of parameters.

However, the estimation of model parameters is error-prone and has not had an optimal parameter estimation strategy so far. In contrast with parametric method, the non-parametric method reduces the costs of parameter estimation, and should circumvent threats caused by wrong parameter estimation.

An approximate-fitted curve is a sufficient venue for decision-making. Curve-fitting algorithms always pursue the best-fitted curves. However, since the data is often noisy, the results from a thorough analysis of the big data may not be worth the cost for completeness. Thus, an approximate clue from big data is enough for us to obtain perfect answers to make a good decision similar to results from a thorough analysis. Fitted curves are often used in statistical inference for uncertainties, errors hidden in measurement data such as the detection and defense of cyber attack, diagnose violations of large scale systems, and quality control of genomic, and DNA sequence analysis. In some cases, an approximate fitted-curve is sufficient to pick out most anomalies. For instance, the values of anomalies are far away from the mean of the data. Hence, this paper presents an approximate curve-fitting algorithm, which enables people to understand the structure of big data, and to detect some abnormal events in big data.

In many applications, most amplitudes in Fourier frequency domain are often small and are close to zero. For example, a typical 8x8 block in a video frame, 89% of Fourier coefficients are negligible[7, 3]. Thus, we want to carry out a non-linear fitted-curve from partial Fourier frequency amplitudes without complicated guess of parameters. Let the sequence of measurement data $x = (x_0, x_1, ..., x_{N-1})$, $N \in \mathbb{Z}^+$, $x \in \mathbb{C}^{\mathbb{N}}$ converts into a series frequency amplitudes $A = (A_0, A_1, ..., A_{N-1})$, $A \in \mathbb{C}^{\mathbb{N}}$, via Discrete Fourier Transform (DFT) that is given by:

$$A_k = \sum_{n=0}^{N-1} x_n \cdot e^{\frac{-i2\pi kn}{N}}$$

where $k = (0, ..., N-1)$. The $x_k$ can be reversed from inverse-DFT formula, which is given by:

$$x_k = \frac{1}{N} \sum_{n=0}^{N-1} A_n \cdot \omega$$

where $\omega = e^{\frac{i2\pi kn}{N}}$ is the frequency of $A$. In this paper, we pose three problems:

1. How to reconstruct a non-linear fitted curve from $m$ largest frequency amplitudes $\hat{A} = (\hat{A_1}, ..., \hat{A_m})$, $m << N$?

2. How to infer $\omega = (\omega_1, ..., \omega_m)$ of $m$ largest $\hat{A}$ via discrete measurement samples $\Omega$ with cardinality less than N?

3. How to estimate the $m$ largest $\hat{A}$ from the measurement data samples $\Omega$?

In contrast with previous curve-fitting algorithms, we exploit the sparsity of Fourier transform to carry out an efficient non-linear curve-fitting algorithm. In this paper, we explore few large amplitudes to build up an approximate-fitted curve. In addition, we design two estimators to pick out large Fourier amplitudes, and their frequency based on the small discrete time-domain measurement samples. In sum, this paper includes the following contributions.

1. An approximate non-linear curve-fitting algorithm: We present a novel non-linear curve-fitting algorithm based on the sparsity of Fourier frequency amplitudes. This curve-fitting algorithm is non-parametric, simple, practical, and scalable. Furthermore, this new method is robust and can alleviates threat, and reduce costs of the parameter estimation.

2. A bootstrapping estimation method for Fourier high frequency: In contrast with previous methods[9, 6, 7], we correct the problems about the selection of windows function, and we exploit the bootstrapping to improve the correctness of the estimation of high frequency.

3. An estimation method with random projection for Fourier high amplitude: We circumvent the ill-condition problems occurred in matrix inversion, and add random projection in our curve-fitting algorithm to reduce the operations of high amplitude estimation for massive data.

# 3. THE NON-PARAMETRIC NON-LINEAR FITTED CURVE ALGORITHM

Typically, the body of a paper is organized into a hierarchical structure, with numbered or unnumbered headings for sections, subsections, sub-subsections, and even smaller sections. The command `\section` that precedes this paragraph is part of such a hierarchy.[1] LaTeX handles the numbering and placement of these headings for you, when you use the appropriate heading commands around the titles of the headings. If you want a sub-subsection or smaller part to be unnumbered in your output, simply append an asterisk to the command name. Examples of both numbered and unnumbered headings will appear throughout the balance of this sample document.

Because the entire article is contained in the **document** environment, you can indicate the start of a new paragraph with a blank line in your input file; that is why this sentence forms a separate paragraph.

# 4. APPROXIMATE FREQUENCY ANALYSIS

## 4.1 Frequency estimator

## 4.2 Amplitude estimator

You may want to display math equations in three distinct styles: inline, numbered or non-numbered display. Each of the three are discussed in the next sections.

### 4.2.1 Inline (In-text) Equations

A formula that appears in the running text is called an inline or in-text formula. It is produced by the **math** environment, which can be invoked with the usual `\begin. . .\end` construction or with the short form `$. . .$`. You can use any of the symbols and structures, from $\alpha$ to $\omega$, available in LaTeX[10]; this section will simply show a few examples of in-text equations in context. Notice how this equation: $\lim_{n\to\infty} x = 0$, set here in in-line math style, looks slightly different when set in display style. (See next section).

### 4.2.2 Display Equations

A numbered display equation – one set off by vertical space from the text and centered horizontally – is produced by the **equation** environment. An unnumbered display equation is produced by the **displaymath** environment.

Again, in either environment, you can use any of the symbols and structures available in LaTeX; this section will just give a couple of examples of display equations in context. First, consider the equation, shown as an inline equation above:

$$\lim_{n\to\infty} x = 0 \tag{1}$$

Notice how it is formatted somewhat differently in the **displaymath** environment. Now, we'll enter an unnumbered equation:

$$\sum_{i=0}^{\infty} x + 1$$

and follow it with another numbered equation:

$$\sum_{i=0}^{\infty} x_i = \int_0^{\pi+2} f \tag{2}$$

just to demonstrate LaTeX's able handling of numbering.

## 4.3 Citations

Citations to articles [1, 4, 2, 8], conference proceedings [4] or books [14, 10] listed in the Bibliography section of your article will occur throughout the text of your article. You should use BibTeX to automatically produce this bibliography; you simply need to insert one of several citation commands with a key of the item cited in the proper location in the `.tex` file [10]. The key is a short reference you invent to uniquely identify each work; in this sample document, the key is the first author's surname and a word from the title. This identifying key is included with each item in the `.bib` file for your article.

The details of the construction of the `.bib` file are beyond the scope of this sample document, but more information can be found in the *Author's Guide*, and exhaustive details in the *LaTeX User's Guide*[10].

This article shows only the plainest form of the citation command, using `\cite`. This is what is stipulated in the SIGS style specifications. No other citation format is endorsed or supported.

## 4.4 Tables

Because tables cannot be split across pages, the best placement for them is typically the top of the page nearest their initial cite. To ensure this proper "floating" placement of tables, use the environment **table** to enclose the table's contents and the table caption. The contents of the table itself

[1]This is the second footnote. It starts a series of three footnotes that add nothing informational, but just give an idea of how footnotes work and look. It is a wordy one, just so you see how a longish one plays out.

**Table 1: Frequency of Special Characters**

| Non-English or Math | Frequency | Comments |
|---|---|---|
| Ø | 1 in 1,000 | For Swedish names |
| $\pi$ | 1 in 5 | Common in math |
| \$ | 4 in 5 | Used in business |
| $\Psi_1^2$ | 1 in 40,000 | Unexplained usage |

**Figure 2: A sample black and white graphic (.eps format).**

must go in the **tabular** environment, to be aligned properly in rows and columns, with the desired horizontal and vertical rules. Again, detailed instructions on **tabular** material is found in the *LaTeX User's Guide*.

Immediately following this sentence is the point at which Table 1 is included in the input file; compare the placement of the table here with the table in the printed dvi output of this document.

To set a wider table, which takes up the whole width of the page's live area, use the environment **table*** to enclose the table's contents and the table caption. As with a single-column table, this wide table will "float" to a location deemed more desirable. Immediately following this sentence is the point at which Table 2 is included in the input file; again, it is instructive to compare the placement of the table here with the table in the printed dvi output of this document.

## 4.5 Figures

Like tables, figures cannot be split across pages; the best placement for them is typically the top or the bottom of the page nearest their initial cite. To ensure this proper "floating" placement of figures, use the environment **figure** to enclose the figure and its caption.

This sample document contains examples of **.eps** and **.ps** files to be displayable with LaTeX. More details on each of these is found in the *Author's Guide*.

As was the case with tables, you may want a figure that spans two columns. To do this, and still to ensure proper "floating" placement of tables, use the environment **figure*** to enclose the figure and its caption. figure*, not figure!

Note that either **.ps** or **.eps** formats are used; use the `\epsfig` or `\psfig` commands as appropriate for the different file types.

## 4.6 Theorem-like Constructs

Other common constructs that may occur in your article are the forms for logical constructs like theorems, axioms, corollaries and proofs. There are two forms, one produced by the command `\newtheorem` and the other by the command `\newdef`; perhaps the clearest and easiest way to distinguish them is to compare the two in the output of this sample document:

This uses the **theorem** environment, created by the `\newtheorem` command:

**Figure 3: A sample black and white graphic (.eps format) that has been resized with the `epsfig` command.**

**Figure 5: A sample black and white graphic (.ps format) that has been resized with the `psfig` command.**

THEOREM 1. *Let $f$ be continuous on $[a, b]$. If $G$ is an antiderivative for $f$ on $[a, b]$, then*

$$\int_a^b f(t)dt = G(b) - G(a).$$

The other uses the **definition** environment, created by the `\newdef` command:

*Definition 1.* If $z$ is irrational, then by $e^z$ we mean the unique number which has logarithm $z$:

$$\log e^z = z$$

Two lists of constructs that use one of these forms is given in the *Author's Guidelines*.

There is one other similar construct environment, which is already set up for you; i.e. you must *not* use a `\newdef` command to create it: the **proof** environment. Here is a example of its use:

PROOF. Suppose on the contrary there exists a real number $L$ such that

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = L.$$

Then

$$l = \lim_{x\to c} f(x) = \lim_{x\to c} \left[ gx \cdot \frac{f(x)}{g(x)} \right] = \lim_{x\to c} g(x) \cdot \lim_{x\to c} \frac{f(x)}{g(x)} = 0 \cdot L = 0,$$

which contradicts our assumption that $l \neq 0$. □

Complete rules about using these environments and using the two different creation commands are in the *Author's Guide*; please consult it for more detailed instructions. If you need to use another construct, not listed therein, which you want to have the same formatting as the Theorem or the Definition[14] shown above, use the `\newtheorem` or the `\newdef` command, respectively, to create it.

## A *Caveat* for the TeX Expert

Because you have just been given permission to use the `\newdef` command to create a new form, you might think you can use TeX's `\def` to create a new command: *Please refrain from doing this!* Remember that your LaTeX source code is primarily intended to create camera-ready copy, but may be converted to other forms – e.g. HTML. If you inadvertently omit some or all of the `\def`s recompilation will be, to say the least, problematic.

# 5. CONCLUSIONS

This paragraph will end the body of this sample document. Remember that you might still have Acknowledgments or Appendices; brief samples of these follow. There is still the Bibliography to deal with; and we will make a disclaimer about that here: with the exception of the reference to the LaTeX book, the citations in this paper are to articles which have nothing to do with the present subject and are used as examples only.

**Table 2: Some Typical Commands**

| Command | A Number | Comments |
|---|---|---|
| `\alignauthor` | 100 | Author alignment |
| `\numberofauthors` | 200 | Author enumeration |
| `\table` | 300 | For tables |
| `\table*` | 400 | For wider tables |

**Figure 4: A sample black and white graphic (.eps format) that needs to span two columns of text.**

## 6. ACKNOWLEDGMENTS

This section is optional; it is a location for you to acknowledge grants, funding, editing assistance and what have you. In the present case, for example, the authors would like to thank Gerald Murray of ACM for his help in codifying this *Author's Guide* and the **.cls** and **.tex** files that it describes.

## 7. ADDITIONAL AUTHORS

Additional authors: John Smith (The Thørväld Group, email: `jsmith@affiliation.org`) and Julius P. Kumquat (The Kumquat Consortium, email: `jpkumquat@consortium.net`).

## 8. REFERENCES

[1] M. Bowman, S. K. Debray, and L. L. Peterson. Reasoning about naming systems. *ACM Trans. Program. Lang. Syst.*, 15(5):795–825, November 1993.

[2] J. Braams. Babel, a multilingual style-option system for use with latex's standard document styles. *TUGboat*, 12(2):291–301, June 1991.

[3] Chandrakasan, Anantha, V. Gutnik, and T. Xanthopoulos. Data driven signal processing: an approach for energy efficient computing. In *In Proceedings of the 1996 international symposium on Low power electronics and design*, pages 347–352, 1996.

[4] M. Clark. Post congress tristesse. In *TeX90 Conference Proceedings*, pages 84–89. TeX Users Group, March 1991.

[5] D. L. Donoho. Compressed sensing. *Information Theory, IEEE Transactions*, 52(4):1289–1306, 2006.

[6] Hassanieh, Haitham, P. Indyk, D. Katabi, and E. Price. Nearly optimal sparse fourier transform. In *In Proceedings of the 44th symposium on Theory of Computing*, pages 563–578. SOTC, 2012.

[7] Hassanieh, Haitham, P. Indyk, D. Katabi, and E. Price. Simple and practical algorithm for sparse fourier transform. In *In Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1183–1194. SODA, 2012.

[8] M. Herlihy. A methodology for implementing highly concurrent data objects. *ACM Trans. Program. Lang. Syst.*, 15(5):745–770, November 1993.

[9] P. Indyk, M. Kapralov, and E. Price. (nearly) sample-optimal sparse fourier transform. In *In Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2014.

[10] L. Lamport. *LaTeX User's Guide and Document Reference Manual*. Addison-Wesley Publishing Company, Reading, Massachusetts, 1986.

[11] D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.

[12] B. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *Automatic Control, IEEE Transactions*, 26(1):17–32, 1981.

[13] L. Ping, T. J. Hastie, and K. W. Church. Very sparse random projections. In *In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 287–29. SIGKDD, 2006.

[14] S. Salas and E. Hille. *Calculus: One and Several Variable*. John Wiley and Sons, New York, 1978.

[15] Transtrum, M. K., B. B. Machta, and J. P. Sethna. Why are nonlinear fits to data so challenging? In *Physical review letters*, pages 06201–1–06201–4, February 2010.

[16] A. Zielesny. *The Levenberg-Marquardt algorithm: implementation and theory*. Springer, 2011.

## APPENDIX

## A. HEADINGS IN APPENDICES

The rules about hierarchical headings discussed above for the body of the article are different in the appendices. In the **appendix** environment, the command **section** is used to indicate the start of each Appendix, with alphabetic order designation (i.e. the first is A, the second B, etc.) and a title (if you include one). So, if you need hierarchical structure *within* an Appendix, start with **subsection** as the highest level. Here is an outline of the body of this document in Appendix-appropriate form:

### A.1 Introduction

### A.2 The Body of the Paper

#### *A.2.1 Type Changes and Special Characters*

#### *A.2.2 Math Equations*

*Inline (In-text) Equations.*

*Display Equations.*

#### *A.2.3 Citations*

#### *A.2.4 Tables*

#### *A.2.5 Figures*

#### *A.2.6 Theorem-like Constructs*

#### *A Caveat for the TEX Expert*

### A.3 Conclusions

### A.4 Acknowledgments

### A.5 Additional Authors

This section is inserted by LaTeX; you do not insert it. You just add the names and information in the `\addition-alauthors` command at the start of the document.

### A.6 References

Generated by bibtex from your .bib file. Run latex, then bibtex, then latex twice (to resolve references) to create the .bbl file. Insert that .bbl file into the .tex source file and comment out the command `\thebibliography`.

## B. MORE HELP FOR THE HARDY

The sig-alternate.cls file itself is chock-full of succinct and helpful comments. If you consider yourself a moderately experienced to expert user of LaTeX, you may find reading it useful but please remember not to change it.