

WestminsterResearch

<http://www.westminster.ac.uk/westminsterresearch>

Machine learning the vanishing order of rational L-functions

Oliver, T., Bieri, J., Butbaia, G., Costa, E., Deines, A., Lee, K.-H., Lowry-Duda, D., Qi, Y. and Veenstra, T.

This is an author's accepted manuscript of an article published in *Advances in Theoretical and Mathematical Physics*, 29 (8), pp. 2381-2401 2025.

The final definitive version is available online at:

<https://doi.org/10.4310/ATMP.260412213616>

The WestminsterResearch online digital archive at the University of Westminster aims to make the research output of the University available to a wider audience. Copyright and Moral Rights remain with the authors and/or copyright owners.

ADV. THEOR. MATH. PHYS.
Volume 19, Number 3, 701–722, 2015

Machine learning the vanishing order of rational L -functions

JOANNA BIERI, GIORGI BUTBAIA, EDGAR COSTA, ALYSON DEINES,
KYU-HWAN LEE, DAVID LOWRY-DUDA, THOMAS OLIVER,
YIDI QI, AND TAMARA VEENSTRA

In this paper, we apply data-scientific methods to learn the vanishing order of rational L -functions for a heterogeneous dataset. Each L -function is represented in our data by finitely many Dirichlet coefficients, the normalisation of which depends on the context. We observe murmuration-like patterns in averages across our dataset, find that PCA clusters rational L -functions by their vanishing order, and record that LDA and neural networks accurately predict this quantity using a relatively small number of Dirichlet coefficients. In general, there is no rigorous method to compute vanishing order; thus, fast and accurate predictions of these values would be useful in a variety of applications.

1. Introduction

In this paper, we apply simple machine learning methods to a heterogeneous dataset of rational L -functions. Our dataset includes L -functions arising from Artin representations, arithmetic curves (such as elliptic curves and genus 2 curves defined over number fields), and certain modular forms (such as classical, Hilbert, and Bianchi modular forms). We are motivated in part by recent work applying machine learning to these homogeneous sub-datasets, especially elliptic curves, which notably led to the discovery of murmurations [HLOP24].

Our main objective is to accurately predict the vanishing order of rational L -functions at their central point. This quantity is conjecturally related to the underlying arithmetic, as per the conjectures of Birch and Swinnerton-Dyer and, more generally, Beilinson–Bloch–Kato. Nonetheless, there is no rigorous method to compute this quantity (cf. Section 2.2). In those cases

for which methods are known, the computation requires $O(\sqrt{N})$ Dirichlet coefficients, where N is the conductor. One of the main takeaways of this work is that machine learning methods can heuristically predict the vanishing order with high accuracy using a relatively small number of Dirichlet coefficients (168).

Although there is some prior work concerning the distribution of vanishing orders within certain subclasses, such as elliptic curves, one novelty of our work stems from the fact that there is no literature on the distribution within a mixed dataset such as ours. We use techniques such as principal component analysis, linear discriminant analysis, feed-forward neural networks, and convolutional neural networks. These methods are simple and deterministic. This is a deliberate choice, so as to set an interpretable benchmark for future research. Comparable techniques were applied to arithmetic curves in [HLO22b, HLO23, HLOP24], to number fields in [HLO22a, AHL⁺23], and to a tiny heterogeneous set of rational L -functions in [Oli24]. While neural networks have been applied to elliptic curve datasets in [KV23] and [Poz24], the novelty of this work lies in extending these techniques to a large, heterogeneous dataset comprising L -functions with diverse origins. Furthermore, we explore transfer learning from one sub-dataset to another.

Following this introduction, Section 2 provides the necessary definitions, discusses the difficulties involved in computing orders of vanishing rigorously, and examines various patterns in the averages of the coefficients. These are closely related to murmurations. Section 3 applies principal component analysis, linear discriminant analysis, and neural networks to datasets of rational L -functions, aiming to learn their vanishing orders. In Section 4 we summarize our main results and offer some future directions. In Appendix A, we provide murmuration patterns for data outside of the dataset studied in the main text.

Acknowledgements

Costa was supported by Simons Foundation grants 550033 and SFI-MPS-Infrastructure-00008651. Lee was supported by Simons Foundation grant 712100. Lowry-Duda was supported by Simons Foundation grant 546235. Qi was supported by the NSF grant PHY-2019786 (the NSF AI Institute for Artificial Intelligence and Fundamental Interactions). We also express

our gratitude to the organizers of the Mathematics and Machine Learning Program at CMSA during the Fall 2024 semester, where this project was initiated.

2. Preliminaries

In this section, we define the terminology used, describe our primary dataset, and document various murmuration-like patterns in averages across the dataset.

2.1. Definitions

Loosely speaking, an L -function is a Dirichlet series $L(s) := \sum_{n=1}^{\infty} a_n n^{-s}$ that has an Euler product and satisfies a functional equation. Each L -function is a generating function for the sequence of coefficients $\{a_n\}_{n=1}^{\infty}$ and the analytic properties of $L(s)$ relate to arithmetic properties of the coefficients. A precise axiomatisation was suggested by Selberg [Sel92].

In greater detail, the functional equation for an L -function is easier to describe after multiplying by a function of the form $N^s \prod_{i=1}^k \Gamma(\omega_i s + \mu_i)$, in which $N \in \mathbb{Z}$, $\omega_i \in \mathbb{R}_{>0}$ and $\mu_i \in \mathbb{C}$ with $\operatorname{Re}(\mu_i) > 0$. We refer to the integer N as the *conductor* of $L(s)$ and $L_{\infty}(s) = \prod_{i=1}^k \Gamma(\omega_i s + \mu_i)$ as the factor *at infinity*. The product $\Lambda(s) = N^s L_{\infty}(s) L(s)$ is called the completed L -function and, for each L -function we consider in this paper, satisfies a functional equation of the shape $\Lambda(s) = \varepsilon \Lambda(\delta - s)$ for some $\delta > 0$ and ε with $|\varepsilon| = 1$.

The degree of $L(s)$ is given by the expression $d = 2 \sum_{i=1}^k \omega_i$ and is conjectured to be an integer for L -functions arising naturally in number theory. (In this paper, we consider L -functions of degrees 1, 2, and 4). The product of two L -functions is another L -function. An L -function is said to be *primitive* if it cannot be written as a product of L -functions with smaller degree. To each L -function we associate a measure of complexity called the *analytic conductor*, given by

$$A = N \cdot \exp \left(2 \operatorname{Re} \left(\frac{L'_{\infty}(1/2)}{L_{\infty}(1/2)} \right) \right),$$

where $L_{\infty}(s)$ is the factor at infinity as defined above. To compare across different degrees, we also use the *root analytic conductor*, which is the d th root of the analytic conductor if L has degree d .

We refer to the sequence $\{a_n\}_{n=1}^\infty$ as the Dirichlet coefficients of $L(s)$. The Dirichlet coefficients of an L -function are multiplicative in the sense that, if $\gcd(m, n) = 1$, then $a_m a_n = a_{mn}$. Though we refer to the coefficients as a_n , there is a choice of normalisation. For example, if $L(s)$ satisfies a functional equation $\Lambda(s) = \Lambda(\delta - s)$, then multiplying a_n by $\sqrt{n}$ gives another L -function with a functional equation of the shape $\Lambda(s) = \Lambda((1 + \delta) - s)$. We can scale an L -function by multiplying by a certain power n^α . This yields the *analytic normalisation* $L_{\text{an}}(s)$, wherein the functional equation has shape $s \mapsto 1 - s$.

We say that $L(s)$ is an *arithmetic* L -function if its Dirichlet coefficients are algebraic numbers. For an arithmetic L -function $L(s)$ given in its analytic normalisation $L_{\text{an}}(s)$, we define its *motivic weight* to be the minimal $w \in \mathbb{Z}_{\geq 0}$ such that, for all $n \geq 1$, the number $a_n n^{w/2}$ is an algebraic integer. In such a case we also define the *coefficient field* of $L(s)$ to be the number field generated by $\{a_n n^{w/2}\}_{n=1}^\infty$. If the coefficient field is $\mathbb{Q}$, then we refer to $L(s)$ as a *rational* L -function. For example, Dirichlet characters determine rational L -functions with motivic weight 0 and elliptic curves over $\mathbb{Q}$ determine rational L -functions with motivic weight 1.

Different normalisations can make certain number theoretic properties clearer. Taking an arithmetic L -function in the analytic normalisation $L_{\text{an}}(s)$ and scaling by the motivic weight by multiplying each a_n by $n^{w/2}$ gives the *arithmetic normalisation*, which we refer to as $L(s)$ (so that $L_{\text{an}}(s) = L(s + \frac{w}{2})$). In this normalisation, the coefficients are algebraic integers, the functional equation has the shape $s \mapsto 1 + w - s$, and the central point is $s = (w + 1)/2$. If $L(s)$ is rational, then, with the arithmetic normalisation, its Euler product can be written as a product over rational primes $L(s) = \prod_p L_p(p^{-s})^{-1}$, with $L_p(T) \in \mathbb{Z}[T]$.

2.2. Computation of vanishing order

Despite being a concrete problem, provably computing the vanishing order for a generic L -function is currently only possible when the vanishing order is 0 or 1. The state of the art reduces to rigorously computing the L -function to sufficient accuracy. Tools such as the approximate functional equation [IK04, §5], when combined with standard bounds for coefficients of L -functions, allow one to compute an L -function $L(s)$ to any finite precision. One can then

apply the argument principle and rigorously compute the winding number along a circular contour Γ around the central point $s = (w + 1)/2$ to find the number of zeros (with multiplicity) near the central point. For a similar method, see [Bob13].

If the winding number is 0, then the order of vanishing is 0. Similarly, if the winding number is 1 (and if we know from the functional equation that there is at least a simple zero), then the order of vanishing is exactly 1. Unfortunately, for higher winding numbers, these methods cannot distinguish contributions coming from a high order zero at the central point or collections of multiple, lower order zeros *very close* to the central point. Hence, this generic method is not sufficient, and can only prove upper bounds on the order of vanishing if the order of vanishing is 2 or higher. Given this, away from *extremely* specialized circumstances, this method only computes the order of vanishing *heuristically*, and this is how the order of vanishing in our data set was computed.

More generally, the vanishing order of an L -function at its central point $r := \text{ord}_{s=(w+1)/2} L(s)$ is expected to reflect underlying arithmetic structure. For example, the conjecture of Birch and Swinnerton-Dyer asserts that the vanishing order of the L -function attached to an abelian variety defined over a number field is equal to the rank of that abelian variety as a finitely generated abelian group. Moreover, in the arithmetic normalisation, the leading Taylor coefficient is conjecturally given by a formula involving other arithmetic quantities.

In this context, the goal of this paper is to see if machine learning can predict a function with no known method of rigorous computation. Further, current heuristic methods are computationally expensive, whereas the machine learning techniques investigated are very rapidly computable. One application is to searches that sieve through families of L -functions to identify exceptional behavior (such as high rank curves, e.g. [EK20]); computational methods that are both fast and highly accurate can facilitate searches on far larger datasets.

2.3. Describing the dataset

Throughout this paper, we use a dataset **RAT** [CL25] consisting of rational L -functions taken from [LMF24], and in particular based on work from [Cre81,

Cre84, Whi90, Cre92, BDKM⁺13, JR17, BSS⁺16, DV21, BBB⁺21, CN21]. The **RAT** dataset contains information about 248,359 rational L -functions with root analytic conductor less than 4. We restrict to root analytic conductor < 4 because this dataset contains a relatively balanced set of L -functions from different origins in this range. The available data from the LMFDB becomes more skewed for larger root analytic conductor. The **RAT** dataset includes the sub-datasets **CMF** (classical modular forms), **ECNF** (elliptic curves defined over number fields), **DIR** (Dirichlet characters) and **G2Q** (genus 2 curves defined over $\mathbb{Q}$). Within **CMF**, we find familiar sub-datasets such as **ECQ** (elliptic curves over $\mathbb{Q}$) and **ART** (Artin representations¹). Similarly, within **ECNF** we find **BMF** (Bianchi modular forms) and **HMF** (Hilbert modular forms). In Figure 2.1, we present an UpSet plot for **RAT**, which is a method for visualising data with more than three intersecting subsets (cf. [LGS⁺14]).

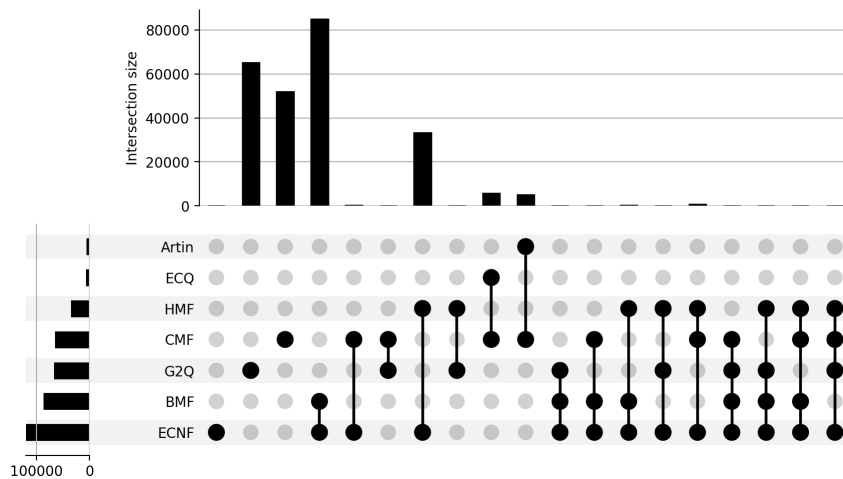


Figure 2.1. UpSet plot for **RAT**. Each row corresponds to the named subset, and each column corresponds to an intersection of subsets. A black circle is used to indicate that a subset is involved in a particular intersection (the vertical lines simply indicate the orientation of the plot). The horizontal bars show the number of datapoints in each subset (row), and a vertical bar above shows the size of an exclusive intersection, i.e., it represents the number of elements unique to that specific intersection.

¹There is one exception. The Riemann zeta function is contained in **ART** but not in **CMF**.

Contained within the **RAT** dataset is a sub-dataset **PRAT** consisting of 186,114 primitive rational L -functions. Since imprimitive L -functions can be written as a product of primitive L -functions, we expect **RAT** to contain theoretically redundant information (for example, all the classical modular forms that do not arise from elliptic curves). We caution the reader that this may not be entirely true in practice, since there is some chance that the imprimitive L -functions in **RAT** do not factor into primitive L -functions that are included in **PRAT**. Nevertheless, for our experiments, we restrict to **PRAT**, in order to exclude the possibility of redundant information. There are other idiosyncrasies in the **PRAT** dataset. For example, there is no overlap between **CMF** and **ECNF** (although this is not true in general). The number of datapoints in each sub-dataset is given in Table 2.1, and the number of datapoints in various pairwise intersections is given in Table 2.2.

In the (P)**RAT** dataset, each row contains information about one L -function and includes several columns [CL25]. This includes the following properties of the L -function:

- primitivity of the L -function;
- N , the conductor of the L -function;
- w , the motivic weight of the L -function;
- d , the degree of the L -function;
- $\{a_p\}_{p \leq 1000}$, the Dirichlet coefficients (in the arithmetic normalisation) at primes less than 1000;
- r , the vanishing order at the central point;
- $t \subseteq \{\text{CMF}, \text{ECQ}, \text{ART}, \text{ECNF}, \text{BMF}, \text{HMF}, \text{DIR}, \text{G2Q}\}$, identifying the type of objects present in [LMF24] that give rise to this L -function (at time of creation of this dataset).

In this paper, we are particularly interested in learning the order of vanishing r column from the Dirichlet coefficients $\{a_p\}_{p \leq 1000}$.

2.4. Murmurations in PRAT

In (P)**RAT**, the Dirichlet coefficients $\{a_p\}_{p \leq 1000}$ are given in the arithmetic normalisation. In the context of murmurations, it is common to (additionally)

sub-dataset	number of datapoints	intersection	number of datapoints
CMF	9,675	$\text{ART} \cap \text{CMF}$	2,556
ECQ	5,860	$\text{BMF} \cap \text{ECNF}$	81,805
ART	2,557	$\text{BMF} \cap \text{G2Q}$	72
ECNF	113,489	$\text{BMF} \cap \text{HMF}$	272
BMF	81,805	$\text{CMF} \cap \text{ECQ}$	5,860
HMF	31,964	$\text{ECNF} \cap \text{G2Q}$	113
DIR	274	$\text{ECNF} \cap \text{HMF}$	31,956
G2Q	62,789	$\text{G2Q} \cap \text{HMF}$	50

Table 2.1. Number of datapoints for various sub-datasets of PRAT.

Table 2.2. Intersection size for various sub-datasets of PRAT.

normalize the Dirichlet coefficients of prime index by defining

$$(2.1) \quad \tilde{a}_p = \frac{a_p}{p^{(w-1)/2}} \in [-d\sqrt{p}, d\sqrt{p}].$$

We note that the normalisation in equation (2.1) is unnatural to both number theorists and data scientists, but, for historical and aesthetic reasons, it seems to be the right choice for the purposes of this section. For example, equation (2.1) specialises to the normalisation in the initial papers and correspondence on murmuration phenomena [HLOP24, Sut22]. In this section, we plot murmuration-like averages for sub-datasets in PRAT. True murmuration behavior relates averages of coefficients in particular conductor ranges [Sar23]. Since our datasets include large ranges of conductor, these figures are not exactly murmurations. For ease of exposition, we refer to these plots simply as “murmurations”.

The counts for each vanishing order in PRAT and varying sub-datasets are recorded in Table 2.3. Since PRAT only contains 9 data-points with vanishing order equal to 4, we restrict to the dataset $\text{PRAT}_{\leq 3}$ consisting of L -functions in PRAT with vanishing order ≤ 3 . In Figure 2.2, we plot the average value of $\tilde{a}_p$ for primitive rational L -functions in $\text{PRAT}_{\leq 3}$ with each vanishing order.

In Table 2.4 (resp. 2.5), we record the number of datapoints in $\text{PRAT}_{\leq 3}$ with given motivic weight (resp. degree). We observe that the vast majority

sub-dataset	vanishing orders				
	0	1	2	3	4
PRAT	59,863	93,657	29,748	2,837	9
ECNF	42,558	61,243	9,661	27	
BMF	28,280	44,773	8,724	26	
HMF	14,443	16,582	938	1	
G2Q	10,827	29,155	19,988	2,810	9
CMF	6,245	3,330	100		
ECQ	2,901	2,862	97		
Artin	2,557				
DIR	274				

Table 2.3. Counts for different orders of vanishing in sub-datasets of PRAT.

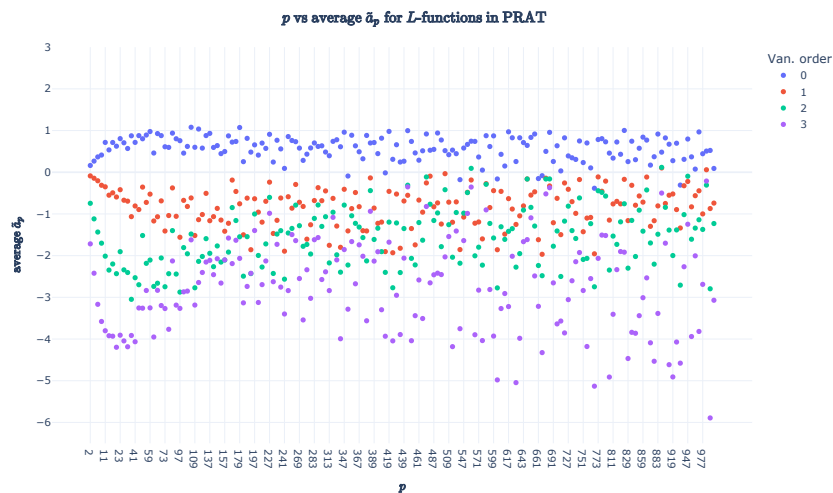


Figure 2.2. Average value of $\tilde{a}_p$ for primitive rational L -functions with specified vanishing order, excluding the 9 L -functions with vanishing order 4.

of L -functions in this dataset have degree 4 and motivic weight 1. Restricting to datapoints in $\text{PRAT}_{\leq 3}$ with this degree and motivic weight excludes a relatively small number of outliers and leaves behind a dataset PRAT^* containing 176,156 L -functions, all of which are instances of either ECNF or G2Q (and some of which are both). In particular, the dataset ECQ is not contained

w	count
1	182,025
0	2,830
3	544
2	177
5	169

Table 2.4. Counts for different motivic weights in $\text{PRAT}_{\leq 3}$.

d	count
4	176,165
2	9,675
1	274

Table 2.5. Counts for different degrees in $\text{PRAT}_{\leq 3}$.

within PRAT^* . On the other hand, we note that ECQ is already extensively studied in the literature [HLO23, HLOP24, Poz24, KV23]. (For comparison, we examine the L -functions in PRAT but not in PRAT^* in Section A.)

In what follows, we analyze the murmuration patterns for PRAT^* and various sub-datasets. For all murmuration plots in this section, L -functions with larger vanishing orders yield smaller average values of a_p . This is consistent with [HLOP24, Figure 8], and is comparable to the theory of Mestre–Nagao sums.

In Figure 2.3, we plot the average value of $\tilde{a}_p$ for L -functions in PRAT^* , grouped by vanishing order (since $w = 1$, we have $\tilde{a}_p = a_p$ in PRAT^*). In Figure 2.4, we plot ECNF and G2Q separately. We observe initial separation in the graphs. More Dirichlet coefficients would allow further exploration of murmuration behavior.

In Figure 2.5, we present a Venn diagram for the largest sub-dataset of PRAT^* , namely ECNF , which includes BMF and all but 8 datapoints in HMF . In Figure 2.6, we plot the average value of $\tilde{a}_p$ over BMF and HMF .

3. Machine learning the vanishing order of rational L -functions

We present an unsupervised approach (PCA) and two supervised approaches (LDA and neural networks). PCA for elliptic curves was documented in [HLO22b, HLOP24], and neural networks were applied to elliptic curves in [KV23, Poz24].

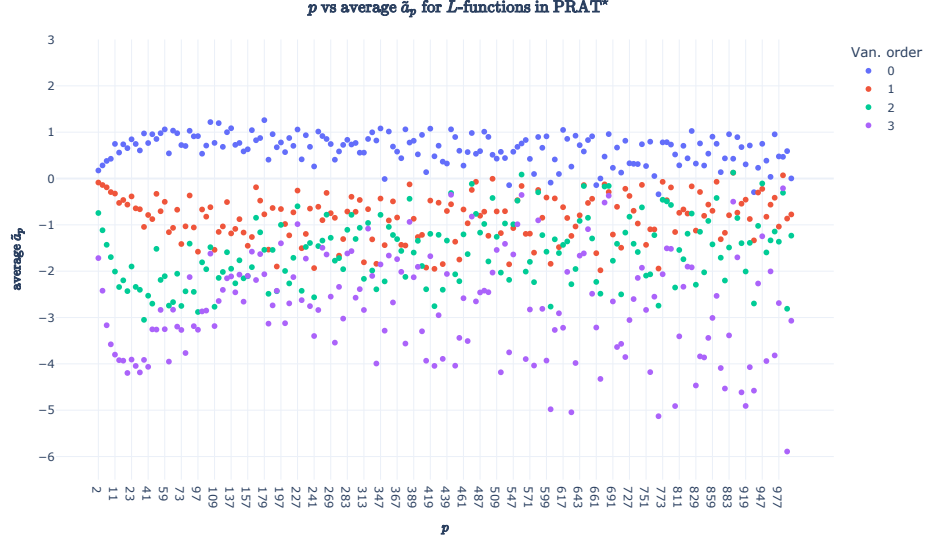


Figure 2.3. Average value of $\tilde{a}_p$ over L -functions with specified vanishing order in PRAT^{*}.

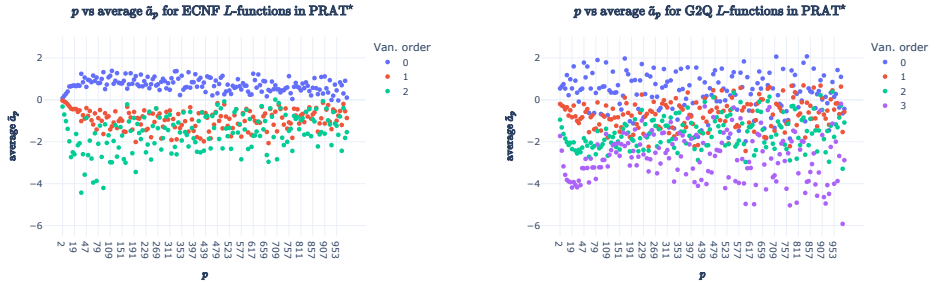


Figure 2.4. Average value of $\tilde{a}_p$ over L -functions with specified vanishing order in (left) ECNF and (right) G2Q excluding the 27 L -functions with vanishing order 3 for ECNF.

3.1. Feature selection and normalisation

As in Section 2.4, we define our features using the Dirichlet coefficients column $\{a_p\}_{p \leq 1000}$. Since the machine learning techniques we use are sensitive to the scale of features and we want to prevent features with larger magnitudes

712

B, B, C, D, L, LD, O, Q, and V

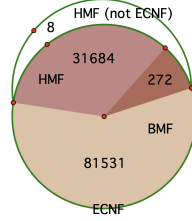


Figure 2.5. Venn diagram for ECNF.

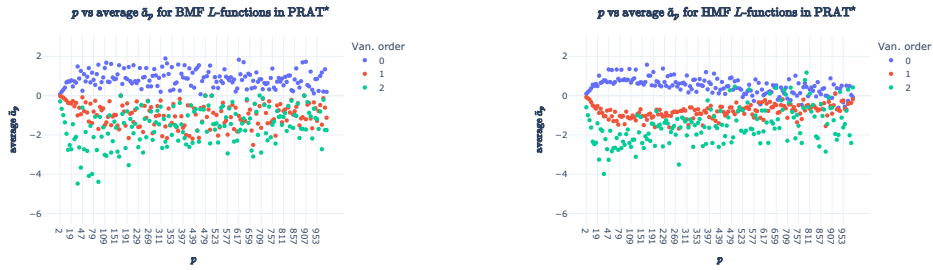


Figure 2.6. Average value of $\tilde{a}_p$ for (left) BMF and (right) HMF, excluding the 26 (resp. 1) L -functions with vanishing order 3.

from dominating the learning algorithm, we switch normalisation methods. The normalisation used in Section 2.4 produced feature vectors with bounds depending on d and p . We now normalize the Dirichlet coefficients with prime index by:

$$(3.1) \quad \bar{a}_p = \frac{\tilde{a}_p}{d\sqrt{p}} = \frac{a_p}{dp^{w/2}} \in [-1, 1].$$

where $\tilde{a}_p$ is as in equation (2.1), d and w are the degree and motivic weight of the L -function, see Section 2.1. Whilst there are several techniques to create features with a similar scale, we chose equation (3.1) due to its arithmetic nature (reflecting Hasse bounds). As noted in Section 2.4, the majority of the data has degree 4 and motivic weight 1, so to eliminate outliers we restrict to datapoints in PRAT*. For L -functions in PRAT*, equation (3.1) amounts to division by $4\sqrt{p}$. Each L -function in PRAT* is then represented by the

following vector:

$$(3.2) \quad v(L) = (\bar{a}_2, \bar{a}_3, \bar{a}_5, \dots, \bar{a}_{997}) \in \mathbb{R}^{168},$$

which is indexed by primes < 1000 . Equation (3.2) determines a pointcloud:

$$\mathcal{D} = \{v(L) : L \in \text{PRAT}^*\} \subset \mathbb{R}^{168}.$$

3.2. Principal component analysis on $\mathcal{D}$

As depicted in Figure 3.1, we ran 2-dimensional PCA on the pointcloud $\mathcal{D}$. Though some separation is visible, we observe that 2-dimensional PCA on $\mathcal{D}$ does not fully distinguish the data by vanishing order. One observes similar behavior for 3-dimensional PCA.

The projection onto each principal component is given by a linear combination of $\bar{a}_p$

$$(3.3) \quad \sum_{p \leq 997} w_p \bar{a}_p, \quad w_p \in \mathbb{R}.$$

We refer to w_p as the *weight* at p . We may calculate the weights by diagonalising the covariance matrix for $\mathcal{D}$. Equation (3.3) is somewhat analogous to Mestre–Nagao sums of the form

$$(3.4) \quad S(B) = \frac{1}{\log(B)} \sum_{p < B} \frac{a_p \log(p)}{p},$$

which have been used in the detection of high rank elliptic curves. Murmurations for Mestre–Nagao sums were observed in [BKN24].

3.3. LDA for vanishing order

We used Linear Discriminant Analysis (LDA) to predict the order of vanishing for L -functions in subsets of PRAT^* from the feature vectors $v(L)$. We divide the data into training and validation sets, with a ratio of 80:20, stratified by the order of vanishing to ensure that we preserve the proportions of the classes in both the training and validation sets. Table 3.1 shows the results both for the full PRAT^* dataset and various sub-datasets. Over all the data, LDA predicts the vanishing order with an accuracy of 95.9% with an

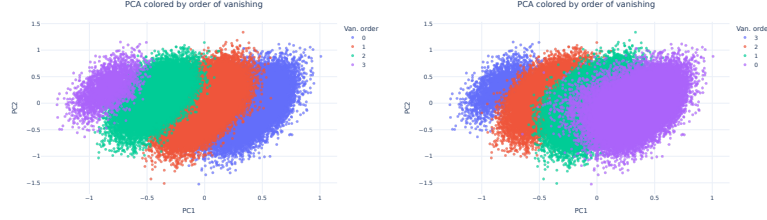


Figure 3.1. Two-dimensional PCA for the PRAT* dataset. Each datapoint is colored by the vanishing order for the underlying L -function at its central point. In the left (resp. right) image we plot the points in ascending (resp. descending) order for the order of vanishing.

explained variance ratio of 0.982. This means that our linear discriminant is highly accurate at separating the classes and is thus highly informative for the classification of vanishing order. If we consider specific subsets, we find some have higher accuracy. For example, LDA predictions on just $G2Q$ are 97.1% accurate with an explained variance of 0.997.

3.4. Neural networks for vanishing order

We may also apply other supervised learning techniques to classify the order of vanishing. To that end, we trained two neural networks: one with inputs given by the principal components, and the other with inputs given by $v(L)$. In both cases, the dataset $\mathcal{D}$ is randomly split into training and test sets exactly as in Section 3.3.

For the network architecture, we tested both feed-forward neural networks (FNN) and 1D convolutional neural networks (CNN). We focus on CNN from this point onward, as it achieved better accuracy than FNN. We also experimented with different problem types, namely, classification (outputting the probability of each vanishing order) and regression (outputting the estimated value for the vanishing order). We observe that most of the phenomena appear to be consistent regardless of the network architecture and problem type.

For the hyperparameters of the CNN, we use three 1D convolutional layers, with 16, 32 and 64 channels, kernel size $K = 3$ and padding $P = 1$, each followed by a ReLU activation function and Max Pooling layer with

Dataset	Training obs.	Validation obs.	Accuracy	Explained Variance	Counts	
PRAT*	140,924	35,232	0.959	0.982	0	53,344
					1	90,327
					2	29,648
					3	2,837
BMF	65,442	16,361	0.958	0.979	0	28,280
					1	44,773
					2	8,724
					3	26
ECNF	90,791	22,698	0.956	0.983	0	42,558
					1	61,243
					2	9,661
					3	27
G2Q	50,224	12,556	0.971	0.997	0	10,827
					1	29,155
					2	19,988
					3	2,810
HMF	25,571	6,393	0.963	0.988	0	14,443
					1	16,582
					2	938
					3	1

Table 3.1. LDA results for predicting vanishing order in PRAT* and various subsets.

$K = 2$ and $P = 1$. After a dropout layer $D = 0.5$, the convolutional blocks are followed by two fully connected layers, each with 128 neurons, and an output layer whose width is equal to the number of possible orders of vanishing in the dataset. The networks are trained with cross-entropy loss using Adam optimizer, setting batch size to 3000 and learning rate to 0.001. We use the same set of hyperparameters for both cases.

When training with principal components, we observe that a CNN can achieve an 91% overall accuracy using only the first two principal components. The percentage accuracy curves and the final test accuracies for the named subsets of PRAT* in this case are presented in Figure 3.2 and Table 3.2. In Figure 3.3, we present the weights of each principal component. We note that similar results were observed in [KV23] using different datasets and features.

When training with $v(L)$, the CNN can achieve over 95% accuracy across all different types of L -functions. The results are shown in Figure 3.4 and

716 B, B, C, D, L, LD, O, Q, and V

Table 3.3. To show the capability of transfer learning of the neural network, we first train on only the ECNF subset and test on G2Q, then reverse the process by training on G2Q and testing on ECNF. The results are shown in Figure 3.5, with both achieving over 90% accuracy on the test set. This suggests that the features learned from ECNF and G2Q are highly transferable and the model generalizes well across PRAT^{*}, indicating a strong similarity between L -functions with the same motivic weight and degree.

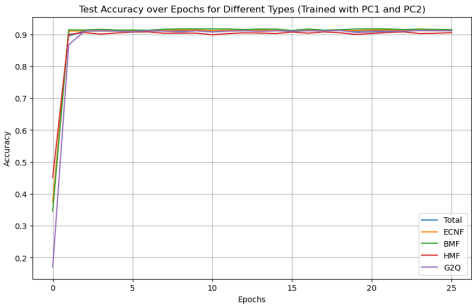


Figure 3.2. Learning the vanishing order from the first and second principal components: percentage accuracy against epoch with CNN. Each colour corresponds to one category of L -functions.

sub-dataset	test accuracy
ECNF	91.22%
BMF	91.48%
HMF	90.54%
G2Q	91.13%

Table 3.2. Test accuracy at the last epoch for each category of L -functions in PRAT^{*}, trained with the first and second principal components.

4. Concluding Remarks

The work in this paper gives rise to many open questions about mathematical structure to be found in a data-focused study of rational L -functions. Predictions using both LDA and neural networks were highly accurate; however, an analysis of the model inaccuracies could give insight into cases where vanishing orders are harder to predict.

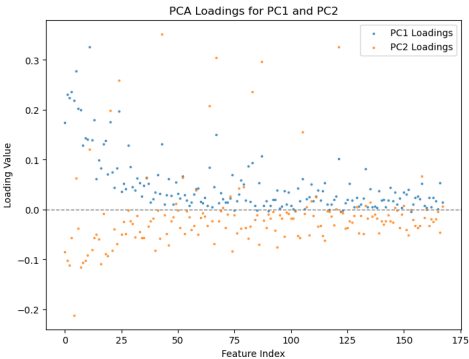


Figure 3.3. The weights in the first and second principal components used in the training.

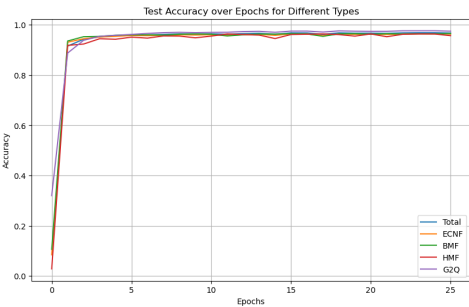


Figure 3.4. Learning the vanishing order in PRAT^{*} from $v(L)$: percentage accuracy against epoch with CNN. Each colour corresponds to one category of L -functions

sub-dataset	test accuracy
ECNF	95.37%
BMF	95.48%
HMF	95.04%
G2Q	95.71%

Table 3.3. Test accuracy at the last epoch for each sub-dataset in PRAT^{*}, trained with $v(L)$.

Our work also considered an exploration of transfer learning and augmented training sets, using principal components, for example. We showed

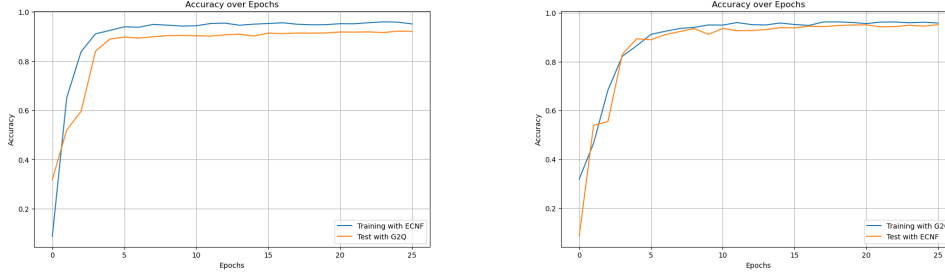


Figure 3.5. Train with ECNF then test with G2Q, and vice versa.

that transfer learning is possible, but further exploration is warranted. For example, to what extent is the accuracy of our classifiers dependent on the conductor range? Can we train our model using L -functions from a minimal conductor range and then use that model to predict order of vanishing across larger ranges? If we train on one data subset, how effective are we at predicting in others? Which Dirichlet coefficients are most important in predicting the vanishing order? Additionally, how does the prediction of vanishing order trained by principal component results compare to Mestre–Nagao sums?

Appendix A. Murmuration Patterns

In what follows, we analyze the murmuration patterns for the various sub-datasets in $\text{PRAT}_{\leq 3}$ that were not contained in PRAT^* . The murmuration patterns are quite different for these subsets of the data, most likely because the ranges for conductors in the dataset for those L -functions are much smaller. The L -functions not included in PRAT^* include CMF , which includes ARTIN and ECQ , and DIR . The L -functions in ARTIN and DIR all have vanishing order 0, so we do not include those graphs. Murmurations of Dirichlet characters, grouped by parity (as opposed to vanishing order), were rigorously studied in [LOP25]. In Figure A1, we present a Venn diagram for CMF , which includes ARTIN and ECQ . In Figure A2, we plot the average value of $\tilde{a}_p$ over CMF and ECQ .

Previous work on murmurations has shown that normalizing the x -axis by dividing by the conductor is the correct normalisation for plots. By restricting to root analytic conductor < 4 in [CL25], the conductor range for

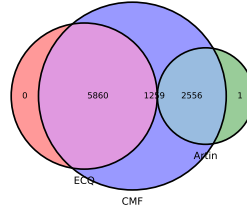


Figure A1. (left) Venn diagram for CMF

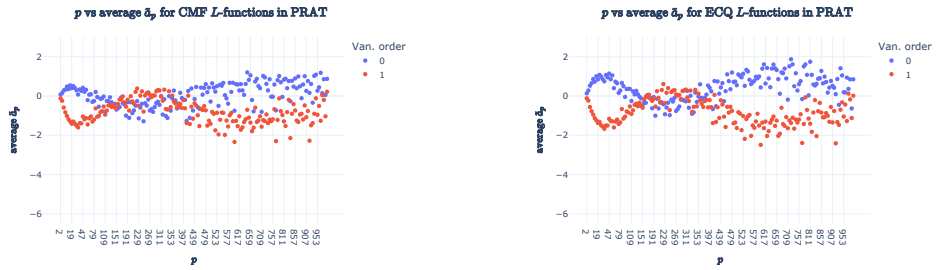


Figure A2. Average value of $\tilde{a}_p$ for (left) CMF and (right) ECQ.

L -functions of degree 2 is much smaller than the conductor range for L -functions of degree 4. Thus when we restrict the dataset **PRAT** to L -functions of degree 2, e.g., **CMF** and **ECQ**, we obtain tighter conductor ranges. This yields murmuration plots with more definition when compared to those plots in Section 2.4.

References

- [AHL⁺23] Malik Amir, Yang-Hui He, Kyu-Hwan Lee, Thomas Oliver, and Eldar Sultanow. Machine-learning of the class number of real quadratic fields. *International Journal of Data Science in the Mathematical Sciences*, (1(2)):107–134, 2023.
- [BBB⁺21] Alex J. Best, Jonathan Bober, Andrew R. Booker, Edgar Costa, John E. Cremona, Maarten Derickx, Min Lee, David Lowry-Duda, David Roe, Andrew V. Sutherland, and John Voight.

Computing classical modular forms. In *Arithmetic geometry, number theory, and computation*, Simons Symp., pages 131–213. Springer, Cham, [2021] ©2021.

- [BDKM⁺13] Jonathan Bober, Alyson Deines, Arian Klages-Mundt, Benjamin LeVeque, R. Andrew Ohana, Ashwath Rabinathan, Paul Sharaba, and William Stein. A database of elliptic curves over $\mathbb{Q}(\sqrt{5})$: a first report. 1:145–166, 2013.
- [BKN24] Zvonimir Bujanović, Matija Kazalicki, and Lukas Novak. Murmurations of mestre-nagao sums. <http://arxiv.org/abs/2403.17626v1>, 2024. arXiv: math.NT: 2403.17626v1.
- [Bob13] Jonathan W. Bober. Conditionally bounding analytic ranks of elliptic curves. In *ANTS X—Proceedings of the Tenth Algorithmic Number Theory Symposium*, volume 1 of *Open Book Ser.*, pages 135–144. Math. Sci. Publ., Berkeley, CA, 2013.
- [BSS⁺16] Andrew R. Booker, Jeroen Sijsling, Andrew V. Sutherland, John Voight, and Dan Yasaki. A database of genus-2 curves over the rational numbers. *LMS J. Comput. Math.*, 19:235–254, 2016.
- [CL25] Edgar Costa and The LMFDB Collaboration. Rational L-functions in LMFDB with root analytic conductor less than 4. <https://doi.org/10.5281/zenodo.14774042>, 2025.
- [CN21] J. E. Cremona and Filip Najman. $\mathbb{Q}$ -curves over odd degree number fields. *Res. Number Theory*, 7(4):Paper No. 62, 30, 2021.
- [Cre81] J. E. Cremona. *Modular symbols*. PhD thesis, University of Oxford, 1981.
- [Cre84] Jonh E. Cremona. Hyperbolic tessellations, modular symbols, and elliptic curves over complex quadratic fields. *Compositio Math.*, 51(3):275–324, 1984.
- [Cre92] J. E. Cremona. *Algorithms for modular elliptic curves*. Cambridge University Press, Cambridge, 1992.

- [DV21] Steve Donnelly and John Voight. A database of Hilbert modular forms. In *Arithmetic geometry, number theory, and computation*, Simons Symp., pages 365–373. Springer, Cham, [2021] ©2021.
- [EK20] N. Elkies and Z. Klagsbrun. New rank records for elliptic curves having rational torsion. <http://arxiv.org/abs/2003.00077>, 2020. arXiv: math.NT: 2003.00077.
- [HLO22a] Yang-Hui He, Kyu-Hwan Lee, and Thomas Oliver. Machine-learning number fields. *Math. Comput. Geom. Data*, 2(1):49–66, 2022.
- [HLO22b] Yang-Hui He, Kyu-Hwan Lee, and Thomas Oliver. Machine-learning the Sato-Tate conjecture. *J. Symbolic Comput.*, 111:61–72, 2022.
- [HLO23] Yang-Hui He, Kyu-Hwan Lee, and Thomas Oliver. Machine learning invariants of arithmetic curves. *J. Symbolic Comput.*, 115:478–491, 2023.
- [HLOP24] Yang-Hui He, Kyu-Hwan Lee, Thomas Oliver, and Alexey Pozdnyakov. Murmurations of elliptic curves. *Experimental Mathematics*, pages 1–13, 2024.
- [IK04] Henryk Iwaniec and Emmanuel Kowalski. *Analytic number theory*, volume 53 of *American Mathematical Society Colloquium Publications*. American Mathematical Society, Providence, RI, 2004.
- [JR17] John W. Jones and David P. Roberts. Artin L -functions of small conductor. *Res. Number Theory*, 3:Paper No. 16, 33, 2017.
- [KV23] Matija Kazalicki and Domagoj Vlah. Ranks of elliptic curves and deep neural networks. *Res. Number Theory*, 9(3):Paper No. 53, 21, 2023.
- [LGS⁺14] Alexander Lex, Nils Gehlenborg, Hendrik Strobelt, Romain Vuillemot, and Hanspeter Pfister. UpSet: visualization of intersecting sets. *IEEE transactions on visualization and computer*

graphics, 20(12):1983–1992, 2014.

- [LMF24] The LMFDB Collaboration. The L-functions and modular forms database. <https://www.lmfdb.org>, 2024. [Online; accessed 27 November 2024].
- [LOP25] Kyu-Hwan Lee, Thomas Oliver, and Alexey Pozdnyakov. Murmurations of Dirichlet Characters. *Int. Math. Res. Not. IMRN*, (1):rne277, 2025.
- [Oli24] Thomas Oliver. Machine learning for number theory: unsupervised learning with L -functions. In *Mathematical software—ICMS 2024*, volume 14749 of *Lecture Notes in Comput. Sci.*, pages 196–203. Springer, Cham, [2024] ©2024.
- [Poz24] Alexey Pozdnyakov. Predicting root numbers with neural networks. <http://arxiv.org/abs/2403.14631v1>, 2024. arXiv:math.NT:2403.14631v1.
- [Sar23] Peter Sarnak. Letter to sutherland and zubrilina. <https://publications.ias.edu/sarnak/paper/2726>, 2023.
- [Sel92] Atle Selberg. Old and new conjectures and results about a class of Dirichlet series. In *Proceedings of the Amalfi Conference on Analytic Number Theory (Maiori, 1989)*, pages 367–385. Univ. Salerno, Salerno, 1992.
- [Sut22] Andrew Sutherland. Letter from sutherland to rubinstein and sarnak. <https://math.mit.edu/~drew/RubinsteinSarnakLetter.pdf>, August 2022.
- [Whi90] Elise Whitley. *Modular forms and elliptic curves over imaginary quadratic number fields*. PhD thesis, University of Exeter, 1990.