12-2010

Features extraction using random matrix theory.

Viktoria Borisovna Rojkova
University of Louisville

FEATURES EXTRACTION USING RANDOM MATRIX THEORY

By

Viktoria Rojkova
M.S., University of Louisville, 2005
M.S., University of Illinois at Urbana-Champaign, 2004

A Dissertation
Submitted to the Faculty of the
Graduate School of the University of Louisville
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy

Department of Computer Engineering and Computer Science
University of Louisville
Louisville, Kentucky

December 2010

FEATURES EXTRACTION USING RANDOM MATRIX THEORY

By

Viktoria Rojkova
M.S., University of Louisville, 2005
M.S., University of Illinois at Urbana-Champaign, 2004

A Dissertation Approved on

11/08/2010
Date

By the following Dissertation Committee:

Mehmed Kantardzic (Dissertation Director)

Patricia Cerrito

Anup Kumar

Eric Rouchka

Adel Elmaghraby

ACKNOWLEDGEMENTS

I would never have been able to finish my dissertation without the guidance of my committee members, help from friends, and support from my husband and daughter.

I would like to express my deepest gratitude to my advisor, Dr. Mehmed Kantardzic, for his excellent guidance, caring, patience, and providing me with an excellent atmosphere for doing research. My warmest thank to Chair of my Department, Dr. Adel Elmaghraby, who as a good friend, was always willing to help and give his best advise.

In addition I would like to thank my committee members Dr. Patricia Cerrito, Dr. Eric Rouchka and Dr. Anup Kumar who corrected my understanding of different application domains and suggested robust real-life application testbeds for our novel methodology.

I would also like to thank my parents and my brother. They were always supporting me and encouraging me with their best wishes.

ABSTRACT

FEATURES EXTRACTION USING RANDOM MATRIX THEORY

Viktoria Rojkova

November 8, 2010

Representing the complex data in a concise and accurate way is a special stage in data mining methodology. Redundant and noisy data affects generalization power of any classification algorithm, undermines the results of any clustering algorithm and finally encumbers the monitoring of large dynamic systems. This work provides several efficient approaches to all aforementioned sides of the analysis. We established, that notable difference can be made, if the results from the theory of ensembles of random matrices are employed.

Particularly important result of our study is a discovered family of methods based on projecting the data set on different subsets of the correlation spectrum. Generally, we start with traditional correlation matrix of a given data set. We perform singular value decomposition, and establish boundaries between essential and unimportant eigen-components of the spectrum. Then, depending on the nature of the problem at hand we either use former or later part for the projection purpose.

Projecting the spectrum of interest is a common technique in linear and non-linear spectral methods such as Principal Component Analysis, Independent Component Analysis and Kernel Principal Component Analysis. Usually the part of the spectrum to project is defined by the amount of variance of overall data or feature space in non-linear case. The applicability of these spectral methods is limited by the assumption that larger variance has important dynamics, i.e. if the data has a high signal-to-noise ratio. If it is true, projection of principal components targets two

problems in data mining, reduction in the number of features and selection of more important features.

Our methodology does not make an assumption of high signal-to-noise ratio, instead, using the rigorous instruments of Random Matrix Theory (RMT) it identifies the presence of noise and establishes its boundaries. The knowledge of the structure of the spectrum gives us possibility to make more insightful projections. For instance, in the application to router network traffic, the reconstruction error procedure for anomaly detection is based on the projection of noisy part of the spectrum. Whereas, in bioinformatics application of clustering the different types of leukemia, implicit denoising of the correlation matrix is achieved by decomposing the spectrum to random and non-random parts.

For temporal high dimensional data, spectrum and eigenvectors of its correlation matrix is another representation of the data. Thus, eigenvalues, components of the eigenvectors, inverse participation ratio of eigenvector components and other operators of eigen analysis are spectral features of dynamic system. In our work we proposed to extract spectral features using the RMT. We demonstrated that with extracted spectral features we can monitor the changing dynamics of network traffic. Experimenting with the delayed correlation matrices of network traffic and extracting its spectral features, we visualized the delayed processes in the system.

We demonstrated in our work that broad range of applications in feature extraction can benefit from the novel RMT based approach to the spectral representation of the data.

TABLE OF CONTENTS

Page

ACKNOWLEDGEMENTS iii
ABSTRACT iv
LIST OF FIGURES ix

I INTRODUCTION 1

A Various aspects of feature extraction. 2

B Feature construction or preprocessing 2

1 Data understanding and feature selection. 5

2 Relevance index. Role of statistics in feature selection 6

C Filters view 9

1 Information-theoretic approach in filters 12

D Wrappers view 14

1 Wrapper framework of embedded methods 15

2 Wrapper framework of ensemble methods 16

E Search strategies for filters and wrappers 17

F Spectral feature extraction 19

1 Principal component analysis 19

G Nonlinear spectral feature extraction. 21

1 Kernel PCA. 21

2 Scaling methods 23

H Structure of the thesis 24

II NETWORKS AS COMPLEX SYSTEMS 27

A Traffic modeling considerations 27

B Spatio-temporal chaos and network communications 31

III RANDOM MATRIX THEORY 34

A Basics of the RMT 35

1 The foundations of the classical RMT 35
2 Universal eigen statistics: Summary of the RMT results . . . 38

IV FEATURE SELECTION WITH THE RMT 40

A Feature construction using covariance matrix of traffic time series . 41
B Eigen system of the matrix C and its interpretation 42
C How to test the eigen system against the RMT predictions 43
D Traffic count data 46
E Results of comparison with the RMT 47
F The eigenstatistics as visual analytics 50
1 Inverse participation ratio 51
2 Stability of inter-VLAN traffic interactions in time - overlap matrix 53
3 Meshgrid of eigenvector components and spatial-temporal representation of traffic load 56
4 Network topological representation of the traffic load 57
5 Experiments with traffic data set to detect anomalies of traffic interactions 58

V CLASSIFICATION WITH FEATURES EXTRACTED BY THE RMT 63

A Motivation for reconstruction error 64
B Reconstruction error used in PCA and RMT based algorithm . . . 65
C Insertion experiments 67
D Comparison of reconstruction errors computed with PCA and RMT 68
E Conclusion and recommendations 72

VI CLUSTERING WITH FEATURES EXTRACTED BY THE RMT 74

A Classification task and similarity measure 74
B Clustering in different contexts 76
C RMT based hierarchical clustering 77
D Clusters in leukemia dataset 80

E Summary of clustering results 83

VII FEATURE CONSTRUCTION WITH DYNAMIC DATASET 85

A Additional Motivation for Time-Lagged Correlation Matrices 86

B Time-lagged correlation matrix of network traffic time series 87

C Selecting eigenstatistics of time-lagged matrix as indicators of network behavior 87

D Eigenstatistics of time-lagged correlation matrix as visual analytics 88

1 Stroboscopic sequence for eigensystem 89

2 Frequency domain analysis 91

E Experiments with altering actual network traffic 92

1 Noise-like injections 93

2 Periodic in time injections 96

F Discussion of results in the context of traffic long range dependence and other applications 97

VIII CONCLUSION 101

A RMT based algorithms and methodology behind them 101

B Summary of the results 103

C Future work 105

REFERENCES 107

CURRICULUM VITAE 122

LIST OF FIGURES

FIGURE		Page
1	(a) PCA succeeds in finding low-dimensional space; (b) PCA finds erroneous linear subspaces in attempt to find "happy face" pattern.	4
2	Probability distributions of the relevance index for both relevant and irrelevant features, illustrating concepts of false positives and false negatives. In reality both distributions are unknown (adapted from Fig 2.1 (p. 67) of [7])	8
3	Empirical probability distribution function $P(\lambda)$ for the inter-VLAN traffic cross-correlations matrix C (histogram).	47
4	The empirical cumulative distribution of λ_i and unfolded eigenvalues $\xi_i \equiv F_{av}(\lambda)$.	48
5	Nearest-neighbor spacing distribution $P_{nn}(s)$ of unfolded eigenvalues ξ_i of cross-correlation matrix C.	49
6	Next-nearest-neighbor eigenvalue spacing distribution $P_{nnn}(s')$.	49
7	Number variance $\Sigma^2(l)$ calculated from the unfolded eigenvalues ξ_i of C.	50
8	Inverse participation ratio as a function of eigenvalue λ.	51
9	Dropped packets per second, (a) congested traffic and (b) uncongested traffic.	52
10	Inverse participation ratio as a function of eigenvalue λ.	53
11	(a) Eigenvalues distributions of traffic streams correlation matrix C for 1 hour, 3 hours and 6 hours time intervals. (b) Eigenvalues distributions for 24 hours, 48 hours and 72 hours	54
12	The gray scale of matrix $O(t,\tau)$ at $t = 60h$ and τ =0, 3, 12, 24, 36, 48h. .	55
13	Eigenvalues distribution, IPR and overlap matrix of deviating eigenvectors.	59

14 (a) The weights of components of u^{497} plotted for time period from 36 to 84 hours of uninterrupted traffic with 6 hours interval. (b) The weights of components of u^{497} plotted with respect to the same time period, with induced three hours correlation. (c) The weights of components of u^{496} plotted with respect to the same time period, with induced three hours correlation. 60

15 Left column - behavior of u^{497} during time period from 48h to 60h with 6h time window, induced correlation starts at 54h and lasts for 3h. Right column - behavior of u^{496} in same conditions. 61

16 (a) The weights of components of u^{497} plotted for time period from 36 to 84 hours with 6 hours interval, two different types of induced correlations. (b) The weights of components of u^{497} plotted with respect to the same time period, three different types of induced correlations. 62

17 Eigenvalues distribution, IPR and overlap matrix of deviating eigenvectors of inter-VLAN traffic cross-correlation matrix C. 62

18 ROC curves of reconstruction error at constant rate attack. Left column ROCs of PCA, right column ROCs of RMT. ROC dependence from number of involved nodes is on row 1, from length of attack is on row 2, from intensity of attack is on row 3. 70

19 ROC curves of reconstruction error at linear rate attack. Left column ROCs of PCA, right column ROCs of RMT. ROC dependence from number of involved nodes is on row 1, from length of attack is on row 2, from intensity of attack is on row 3. 71

20 ROC curves of reconstruction error at exponential rate attack. Left column ROCs of PCA, right column ROCs of RMT. ROC dependence from number of involved nodes is on row 1, from length of attack is on row 2, from intensity of attack is on row 3. 72

21 Polar dendrogram of denoised distance matrix, terminal nodes are labeled with respect to their ALL or AML correspondence."L" is ALL, "M" is AML. 81

22 Polar dendrogram of denoised distance matrix, terminal nodes are labeled with respect to their ALL B-/T-cells or AML correspondence."M" is AML,"L" is B-cell ALL and "T" is T-cell ALL. 81

23 Validating Clusters indices. Adjust Rand and Jaccard indices are so called "external" validators, they use the external available information about class assignment. Dunn and Silhouette indices are "internal", they evaluate individual quality of the cluster, such as its compactness and separability from another cluster. 82

24 Adjusted Rand Index, Silhouette Width, Dunn Index and stability (averages over 21 runs) for k-means, SOM, SOTA, average link and single link agglomerative clustering on the Leukemia test set, adapted from [107]. . . 83

25 (a) left, (b) random, and (c) right parts of the eigenvalue spectrum as obtained from actual data. Same graphs are presented in (d), (e), (f) respectively, after noise-like injections are made. 91

26 (a) $I(0)$ versus position in spectrum. Stroboscopic representation of IPRs corresponding to (b) first 10τ (c) second 10τ; (d)-(f) are the same representations upon noise-like injections. 92

27 Eigenvalues number (a) 2, (b) 257, and (c) 497, plotted with respect to time and their respective Fourier spectra ((d) through (f)). 93

28 IPRs for eigenvalues number (a) 2, (b) 257, and (c) 497, plotted with respect to time and their respective Fourier spectra ((d) through (f)). . 94

29 Eigenvalues number (a) 2, (b) 257, and (c) 497, plotted with respect to time and their respective Fourier spectra ((d) through (f)) after noise-like sample was injected. 95

30 IPRs for eigenvalues number (a) 2, (b) 257, and (c) 497, plotted with respect to time and their respective Fourier spectra ((d) through (f)) after noise-like sample was injected. 95

31 Eigenvalues number 2, 257, and 497: The results of the injections with $2.5\,min$ period. 98

32 Eigenvalues number 2, 257, and 497: The results of the injections with $15\,min$ period. 98

33 Eigenvalues number 2, 257, and 497: The results of the injections with 20 *min* period. 99
34 Eigenvalues number 2, 257, and 497: The results of the injections with 30 *min* period. 99

CHAPTER I

INTRODUCTION

It has been known for a long time, that data arrays, no matter how massive or problem specific, always carry unique attributes. The features incorporated into a bulk of data points play crucial role in approaching the biggest modern engineering challenge - efficiency in information storage and processing. They are also primary targets of modern learning machines: neural networks, tree classifiers, and Support Vector Machines (SVM)[1].

In this thesis we propose a particularly efficient method of feature extraction, largely independent of problem nature or size of the data set. Our method is based on the random matrix theory (RMT) and goes beyond standard spectral approaches such as principal component analysis (PCA) or its non-linear counterparts. The success of PCA and related spectral methods is limited by applicability of the assumption that larger variance has important dynamics. For example, they work effectively when data has a high signal-to-noise ratio. Our methodology is related to PCA in its spectral nature and goals to recover principal contribution. But it bypasses the assumption of high signal-to-noise ratio or similar to that of. Instead, using the rigorous instruments of Random Matrix Theory (RMT) it identifies the presence of noise and establishes its spectral boundaries. The knowledge of the structure of the spectrum yields insightful possibility for structure identification. Not to take anything form PCA, but the proof of our method superiority will be illustrated by comparison in anomaly detection section of this thesis. The detailed explanation of PCA maladies is also given in the following section.

A Various aspects of feature extraction.

Data mining and machine learning communities recognize that the feature extraction (FE) has several aspects, which interplay or follow each other depending on different goals [2]. More general view on FE considers *wrappers* and *filters* [11]. Where wrapper is aiming the enhancement of learning machine or predictor or classifier generalization, so it is incorporated or wrapped around a particular classifier. Filters, on the other hand are not involved into the learning process, their relevance criteria is calculated with relation to the class or label without being a part of classification and performance improvement. We will cover in the text some of the commonly used wrappers and filters. Both, filters and wrappers can make use of search strategy to explore the space of all possible feature combinations that is usually is very large to be explored exhaustively. Sometimes feature extraction is a hybrid of filter and wrapper.

Another view on FE decomposes it into feature construction and feature selection [12]. Feature construction essentially constitutes the preprocessing of the data, which includes standardization, normalization, signal enhancement, extraction of local features, linear and non-linear space embedding methods, non-linear expansion and feature discretization. Selection of informative and relevant features is the primary but not the only goal of feature selection. It includes as well general data reduction, to limit storage requirements, feature set reduction, to facilitate iterative algorithms, performance improvement, to gain predictive accuracy and finally, data understanding, to gain knowledge about the system that generated the data or simply visualize it. In following subsections we will bring the examples of all of the above mentioned methods. As far as we can see, the intricate combination of these methods, constitutes the feature extraction field.

B Feature construction or preprocessing

Whether a view on experimental results is taxonomic or ergonomic, the extraction of meaningful information requires data mining. Among many others, the primary concern in this process is level of supervision. The larger the output of the experiment, the lower the supervision level should be. For example, in processing

healthy and cancerous X-ray images, the human expertise, while being valuable for small size n of pattern vector $x = (x_1, x_2, ..., x_n)$, becomes powerless with increasing n. Here x characterizes the image in binary, categorical, or continuous way, and in general, can have vectors in place of its components x_i. The total number of features involved can be astronomically large, but actual pattern, distinguishing cancer and control patients can be reasonably small.

The first step in such a dimensional reduction is defining the features of interest, which in the context of this thesis is preprocessing of the original data. We will refer to this vector x' as a vector of transformed features. For simplicity we will think of it as a raw or column data of size $n' \leq n$, even though the following procedures can be generalized to higher dimensional data arrays.

The two simplest preprocessing stages needed for effective data mining are standardization and normalization. A typical example of bringing the data to the same scale is given by $x'_i = (x_i - \mu_i)/\sigma_i$, which uses mean of the respective feature μ_i and its standard deviation (STD) σ_i. Normalization, formally written as $x' = x/\|x\|$, depends on the definition of metric $\|...\|$, but otherwise is just as natural as use of percentage or fractions in place of absolute values. The most common choices for the metric include the Euclidean length, maximum component, sum of the components, their average or standard deviation. Both procedures aim at removing dependence on measurement units, nature of data points, and specifics of experimental setup.

In most experiments, data ends up being polluted by the noise and erroneous data. Consequently, preprocessing uses enhancement of signal-to-noise ratio, the procedure borrowed from signal processing, and naturally understood in the context of image recognition. The signal enhancement can be achieved through smoothing, sharpening, and de-noising techniques, background removal, as well as various filtering methods, employing Fourier or wavelet transforms. Despite highly developed methodology, this stage of data manipulation always faces a problem of handling "baby and bath water".

But the real dimensionality reduction demands more creative and sophisticated approaches, such as PCA[13] or Multidimensional scaling (MDS)[14]. These are termed the embedding methods, as they assume the existence of lower dimensional space

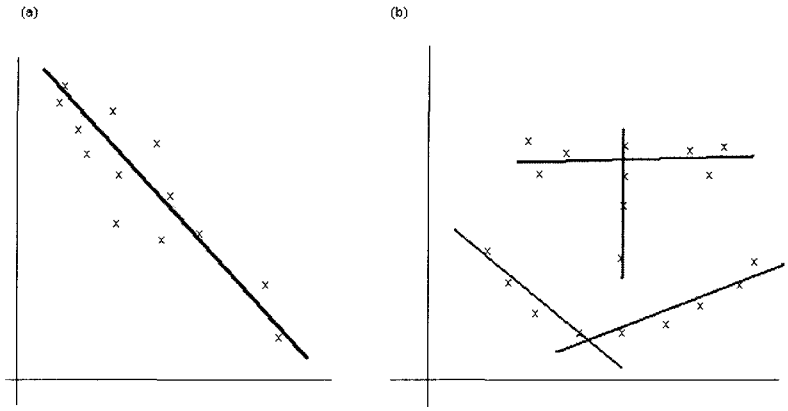

Figure 1. (a) PCA succeeds in finding low-dimensional space; (b) PCA finds erroneous linear subspaces in attempt to find "happy face" pattern.

absorbing most of the relevant information (Fig. 1a). The linear method, such as PCA is typically easy to implement, but is hard to rely on, whenever the underlying sub-space is non-linear. An exemplary illustration are face recognition and handwriting analysis. The printed letters are amenable to PCA, but their longhand replicas are not, due to non-linear functional representation of the latter. Even a "happy face" is too complex of an object for the conventional PCA (Fig. 1b). These and many more less transparent data mining tasks call for non-linear extensions of PCA or constructively different methods.

The non-linear embedding methods have their own share of downsides, including semi-empirical transformation kernels, tractability loss and storage volume increase. The majority of data sets studied within these approaches require appropriate polishing, which often undermines the idea of unsupervised learning. One of such cases described in [3] involves linear method [4] working perfectly as long as non-linear handle [5] on the data is used. Such pre-processing is equivalent to labeling data. And regardless of success level the applied method represents supervised learning.

1 Data understanding and feature selection.

The data mining community had developed a long variety of data interpretation techniques. The main challenges as of today are more theoretically plunged algorithms, better estimates of the computational burden, and improved performance assessment of feature selection. We will be having all these in mind when analyzing our data set. At the same time, several established frameworks of feature selection need to be outlined for the reasons of deeper understanding of main ideas behind it.

A great facilitator for data understanding is the learning machine, an algorithm that learns from data by searching for the most adequate model. A typical learning algorithm learns from a sequence of data, is made either of objects or objects and targets, which corresponds to unsupervised and supervised learning respectively. Given a task machine learns on experience using performance metric. The learning is going in the right direction provided that performance metric is improving.

As for the model building strategy, it is normally based on statistics and optimization. For example, the classification models divide the feature space into regions assigned to class label. A simplest illustration is the linear discriminant method. One constructs a linear function of the data $f(x) = w^T x + b$, by specifying widths vector w and a threshold $-b$. After that, the label assignment is determined by the sign of the function $f(x)$. For $f(x) > 0$ the corresponding object receives "+" label, otherwise it receives "-" label: $y_i = \{-1, 1\}$. To estimate w and b the linear regression follows. In other words, we minimize $\sum_i ||f(x_i) - y_i||^2$ upon centering of data vectors and labels [6].

A more advanced and versatile technique is widely popular Support Vector Machine, a method also, but more explicitly centered on the search for the hyperplane $f(x) = w^T x + b$. In essence, the construction of such hyperplane is tantamount to maximization problem applied to a distance between "training" data point and decision boundary. And the search process can be manipulated into an optimization problem which results in the decision function $f(x) = sgn(\sum_{i=1} \alpha_i x^T x'_i + b)$, where α are Lagrange multipliers of optimization problem. For non-linear version SVM, one replaces the inner product $x^T x'$ in the decision function with the kernel function $k(x, x') = \phi^T(x)\phi(x')$, where the explicit form of mapping $\phi(x)$ is usually unknown,

and what's more, is unnecessary. Such an elegance, comes at the expense of of mathematical rigor. The kernel comes in a variety of forms: polynomial, Gaussian, and hyperbolic tangent, none of which have any mathematical justification. Such kernels are also common in PCA, whenever the nature of the problem ceases to be described by a linear product xx'.

Such methods as SVM, as well as other machine learning tools, e.g. neural networks and decision trees are largely non-universal. According to Ref. [7] (I.1.4, p. 57) - "No algorithm is perfect or best suited for all the applications". Indeed, the idea behind certain choice of the method is based upon computational complexity, number of features, for instance. What is really crucial, for successful data interpretation is the method of assessment and validation. We thus focus on these two aspects in the following section.

2 Relevance index. Role of statistics in feature selection

Consider now another tool-set used in feature selection. Relevance index for a trial feature subset of data represents systematic measure of the degree of correspondence between the subset and the task it intends to accomplish. For example, such task can be in classification of data using decision tree, or in building a hyperplane in the framework of SVM. Even more instructive way to mathematically and visually define relevance index arises in shape search applications [8]. Such indices could be the number of non-null pixels on the image, the average of the lengths of all possible cords connecting two contour points, or the sum of the distances between the center of mass of the model and all visible points of the model [8].

Often, a relevance index is the distance to be minimized, or it is information gain. These choices are clearly not unique, and, furthermore, many "derivative" relevance measures exist. These indices, either real or categorical, do quantify the relationships between features and modeling parameters for regression, classification, and other approaches. Yet, the statistical treatment is needed in using a relevance index, as both variables characterizing candidate feature and model are frequently stochastic. Even if data source is deterministic, in most cases the data size is too large and relationship between variables are hardly predictable for any other treatment to be

used.

The relevance index is thus a realization of a random function of two variables - the modeling one and the one being modeled. Modeling variables split into relevant and irrelevant, naturally coming from different distributions [9, 10]. This fact allows to determine threshold of accepting or discarding variables by computing relevance index and random probes generation. In Fig. 2 we illustrate the feature selection process as decision-making guided by probabilities of false-positive and false-negative. In practice, the probability distributions for relevant and irrelevant features are not being known. But they can be recovered with the help of random probes and standard statistical techniques such as computing cumulative distribution function (CDF), Gaussian smoothing, etc. Upon selection of risk (of selecting a feature that is less relevant than a random one) level, a candidate feature is picked from ranked list, and its relevance index is computed, to be used in finding respective CDF. From latter, the probability of probe being more significant than selected features is found and algorithm is either terminated or carried on, depending on relation to the risk level. This is a typical feature selection procedure. One of such procedures, was described in detail in Ref. [10]. To summarize, the decision of feature selection or rejection is tied to the index going over or under the threshold of a false positive.

Additionally, statistical hypothesis testing apparatus truly comes into play due to the scarcity of available training data. For example, univariate tests of variable irrelevance assume Gaussian distribution of respective relevance index. The random probes replace analytical calculations of test distribution, which in turn allows "null model" considerations. The hypothesis H_0, of variable being irrelevant, can be tested provided we know distribution of relevance indices; and similarly, the hypothesis H_0 of expectation of relevant index exceeding the threshold can be verified as long as we know the threshold value.

When candidate feature is adequately described by the vector with components representing the values of the training (input) set, the relevance measure is simply an angle φ_i between vectors of feature and and the output. If the output is aligned with the input, it is explained through the latter. If, on the contrary, the angle is ninety degrees, the output has no correlation with an input. All the intermediate relevance

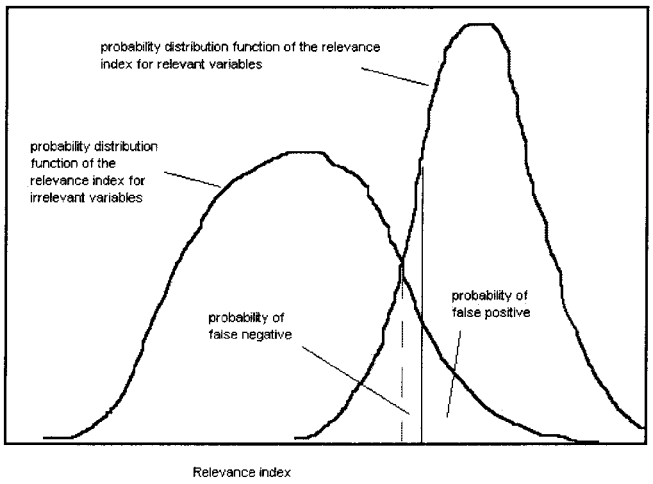

Figure 2. Probability distributions of the relevance index for both relevant and irrelevant features, illustrating concepts of false positives and false negatives. In reality both distributions are unknown (adapted from Fig 2.1 (p. 67) of [7])

index outcomes can be easily ranked. The angle φ_i is directly related to the Pearson coefficient: $\cos\varphi_i = (yx_i)/|y|\,|x_i|$, where y is a modeled quantity and x_i is i-th candidate feature. The use of random probes makes the selection procedure well grounded. Specifically, one can stop the search process after reaching a certain number of relevant features. It is also possible to have an estimate for the risk of retaining a candidate feature in favor of more relevant probe.

At the same time, the machine learning oriented modeling implies more than just estimating of the parameters in functions or governing equation, viewed as "true". Generally, we expect these parameters to be retrievable from the experiment, together with corresponding confidence intervals. But in machine learning problems, the model itself is in question. Consequently there are several regulatory principles in searching for a "true" model using machine learning.

As a rule, a data set is split into a training, validation and test parts, used for parameters estimates, model selection, and performance assessment respectively. For moderate sizes of data sets, even more subsets can be used. The strategy is to run training on as many data sets as possible, with subsequent validation on the last set; and then, to repeat the process the same amount of times as the number of subsets.

Such multiple cross validation ensures that the best model is chosen and overfitting is circumvented. Here relevance index is employed during the training stage.

Furthermore, series of very sophisticated variations of this technique are available, (see Chapter 2 of Ref. [7]). The main criteria used in validation and cross-validation is second order statistics, which keeps track of the consistency of performance and makes sure that estimated noise levels are matched. But without an appropriate relevance index the procedure would have been impossible.

And finally, the model selection process, e.g. proper kernel search in non-linear SVM and PCA, can integrate cross-validation and feature selection in the following algorithmic way. After separating the data into training and validation parts, one ranks all the features and creates their nested subsets. Then, model is trained and validation errors (risk variance) corresponding to different subsets of features are averaged. Next the optimal number of features is determined, in accord with minimum error, and the features are ranked again. As the final step, one chooses thus determined number of high-ranked features, trains the final model, and runs the test on independent data set.

C Filters view

One of the few popular feature selection algorithms is associated with removal of the unlikely candidates. This approach uses no predictors or modeling frameworks, but instead, calculates the metric from the data. The main advantage here is relatively cheap computational effort.

The filter is a functional tool, which returns relevance index for a given subset of features and ranks features according to their relevance[15]. The low-ranked features are considered useless, provided no mutual correlation exists. The filters are either local or global. The high-ranked features are considered useful and are retained in further data analysis, provided no mutual correlation exists. The filters are either local or global. Global feature assessment excludes any knowledge of physical context of the problem and targets the entire data. Meantime, local classification algorithms focus on specific data points and their neighborhood.

An essential part of any filter is establishing of a feature relevance [16]. The most straightforward definition behind both global and local filters maps the probe of

relevance onto comparison between conditional and unconditional probabilities of feature taking a certain value and that of a class. A feature can only be relevant if these two are different. Another factor, is, of course, statistical dependence, which make selected features redundant.

To cover both issues at the same time we consider a perfect example of feature relevance measurement, the Pearson correlation coefficient. It has an advantage of being a function of all the variables involved. It ranges between -1 (perfect linear anti-correlation) and 1 (perfect linear correlation); and it passes through zero whenever feature X with values x and target classes Y with values y and class are uncorrelated. It reads:

$$\rho = \frac{\sum_i (x_i - \overline{x}_i)(y_i - \overline{y}_i)}{\sqrt{\sum_i (x_i - \overline{x}_i)^2 (y_i - \overline{y}_i)^2}}. \tag{1}$$

Taking Pearson coefficient as a stochastic variable one can describe it with a probability density function

$$\mathcal{P}(X \sim Y) = \mathrm{erf}\left(|\rho(X,Y)|\sqrt{m/2}\right),$$

where erf stands for the error function and m is the number of samples. Note the sharp dependence on m which renders small correlation coefficients highly probable. This distribution could have provided a simple feature ranking criteria, but in some cases, $P(X \sim Y)$, for most of the features, is too close to 1 to be sorted in any order. Then, more elaborate statistical tests need to be used, such as, for example, Kolmogorov-Smirnov [17].

Another helpful concept in feature ranking is distance between probability distributions, which in its simplest form expresses the difference between joint and product distributions:

$$D_{pdf} = \sum_i \sum_{j=1}^{K} \left[P(y_j, x_i) - P(x_i) P(y_j)\right]. \tag{2}$$

A more robust versions of this relevance index are reviewed in [15]. Class and feature distributions can also be used in the context of information theory indices.

Information contents of the respective distributions are, by definition [18],

$$H(Y) = -\sum_i P(y_i) \log_2 P(y_i), \tag{3}$$

$$H(Y) = -\sum_i P(y_i) \log_2 P(y_i). \tag{4}$$

The relevance index is defined similarly:

$$H(Y,X) = -\sum_i \sum_j P(y_j, x_i) \log_2 P(y_j, x_i) \tag{5}$$

Low information values mean significance of certain features X in certain intervals of Y, creating an opportunity for the feature ranking and subsequent selection. Just like information gain utilized in decision tree, the information gain is highly effective as relevance index. Here it is defined as a difference between class information and "conditional" information, according to [18]

$$IG(Y,X) = H(Y) - H(Y|X) = H(Y) + \sum_{ij} P(y_j, x_i) \log_2 P(y_j|x_i) \tag{6}$$

Both feature and feature-class conditional probability densities are also useful for building the decision trees used in filtering. These are trees oriented on a specific feature as opposed to general decision trees. If the splits are binary, such trees are based upon single feature, and when algorithms are multi-splitting, the single level tree is built. Once again indices are computed through distances between distributions or through information content (see, for instance, [19]).

With all these different approaches in mind, particularly important questions rise when it comes to performance evaluation: How to spot good relevant indices? How to compare different indices? and how universal are the good ones? Turns out, the nature of the data is the most crucial ingredient for the index choice [20, 21]; bioinformatics, text recognition and medical diagnostics all have their own favorites. In most cases data set dimensions explain the data-miner's choice [22], but there are puzzling exceptions. An easy comparison can be made if monotonic functional dependence can be established between two different relevance indices.

In summary, filters are comparatively inexpensive way of feature selection. The existing great variety of techniques and quantifiers leads to a copious number of relevance indices. The search for efficient comparison framework and universality is an ongoing process, but large number of established methodologies in feature selection is already available. The statistical approach in filter construction combined with methods of information theory possesses an extensive track record, and creates a good

foundation for bringing in algorithms and tools from other areas. For that reason, we turn to information methodology in the next subsection.

1 Information-theoretic approach in filters

Shannon's way of quantifying information had long found its way to most of the areas of complex systems theory [23]. In natural sciences it is motivated by importance of information's antipode, the entropy. In studies of information networks it is self-explanatory. And, finally, in data mining and machine learning, knowledge of information content measurement can provide invaluable tool for feature extraction.

Specifically, both relevance criteria and search algorithm are formally connected to mutual information I. A variable gets selected on the account of maximizing function I, which spans over original variables, targets (class labels), and parameters. Assuming that training data is drawn from the same distribution as data in question, the basic idea behind building the reliable predictor is founded entirely on availability of risk estimates. That is where information theory becomes truly invaluable. We now proceed with its quick overview.

For a random variable X, characterizing experimental data, and discrete target variables Y, labeling classes, Shannon's entropy is defined through the average logarithm of probability density:

$$H(X) = E\left[\log_2 \frac{1}{p(\mathbf{x})}\right] \tag{7}$$

and

$$H(Y) = E\left[\log_2 \frac{1}{p(\mathbf{y})}\right] \tag{8}$$

respectively. From that, the so called, conditional entropy is readily written as follows:

$$H(Y|X) = \int p(x)\left(-\sum p(y|x)\log_2 p(y|x)\right) \tag{9}$$

It quantifies class entropy over all possible values of data. Next we write down the mutual information between X and Y as

$$I(Y,X) = H(y) - H(y|x) = \sum \int p(y,x)\log_2 \frac{p(y,x)}{p(y)\,p(x)} dx \tag{10}$$

Note, that we abandoned (hard-to-compute) conditional probability distribution function $p(y|x)$ in favor of joined probability distribution function $p(y,x) = p(y|x)\,p(x)$, and we used $p(y) = \int p(y,x)\,dx$.

Mutual information reflects dependence between Y and X variables. If joined probability distribution factorizes into $p(x)\,p(y)$, this dependence disappears; data and classes become independent.

Feature construction and selection both have transparent analogies in communication. Besides, since we will be discussing rate of information transmission it is worth to discuss its several general governing principles. According to Shannon [23], the rate R of transmitting X and getting Y on the other end is given by $R = H(X) - H(y|x)$, while channel's capacity is obtained as maximum R over all possible distributions of X. Channel input is defined through data set X, which serves as encoding to a "real source" Y. Unlike in communication problem, we fix X, but the channel output is a function $\Phi(X,\theta)$ of tunable parameters θ. The "channel" capacity is maximum rate R with respect to θ. Hence, to maximize information about "encoded" features Y one simply need to maximize mutual information between Y and Φ.

It also make sense to take a look at application of information-theoretical approach to optimal variable selection. Since feature selection targets redundancy and relevancy issues, we note that mutual information links those two. It also serves as a criterion in feature construction. Consider training data $\{\mathbf{x_i}, \mathbf{y_i}\}$. We want to find a function $\Phi(x,\theta)$, such that information $I(Y,\Phi)$ reaches its maximum. Here the knowledge of gradient $\partial I/\partial\theta$ plays crucial role in inverting relation between target function Φ and unknown parameters θ_i. No greedy optimization is required, and the problem can be handled with standard gradient descent numerical technique. See, for example [38].

Particularly useful in the context of feature selection are three methods described in recent works [28] and [29]. First two (of Ref. [28]) are fast-convergent and have high feature reduction ratio. They even apply for continuous and interdependent attributes. The authors perform data normalization, between 0 and 1, making sure that they have the identical scale for the continuous attributes). Then, they apply the

so-called Variance Gain metric to each attribute and extract relevance. This metric realizes a descending variance ordering in attribute ranking process. Next, algorithm selects the best attributes using threshold defined by the largest gap between two consecutive ranked attributes. And finally, the selected attributes are used for data mining induction.

Method of Ref. [29] builds upon difference image entropy concept and achieves remarkable improvement in pattern recognition accuracy. This work put forward a proposition that increase in number of frame indices can increase accuracy in pattern recognition. Specifically, provided the subject detected in a given image is judged as inadequate [29], the next frame image is inserted into the system. The process continues until detection of an adequate subject. The subject selection is based on differential image entropy (DIE). The selection module proposed in [29] calculates a differential image through pixel-level subtraction between pre-processed images and an average image. The DIE value is then compared to the threshold entropy value. Upon selection of current frame pre-processed image it is processed using traditional recognition algorithms (e.g. PCA, or linear discriminant analysis). In the event of the DIE exceeding the threshold value, the next frame image is loaded in. Further reviews of most recent practical applications can be found in [30].

All in all, information-theoretical approach proves to be very effective [15], as long as probability density can be recovered from the experimental data. Its firm analogue with transmission rate measure in communication problem, makes the mutual information ideal candidate for criterion in feature extraction. In addition, ties to transmission problems create a connection to network data we analyze in what follows. We will come back to this methodology later on when discussing the RMT and its network applications.

D Wrappers view

To conclude our overview of feature selection and extraction we go over wrapper approach as well. Selecting the features subset based on enhanced learning performance is the main difference of wrappers from filters. In the context of this new branch of methodologies, the feature selection becomes an extraction of a subset, which

provides the utmost in representation of the data. The subset has a priori specified dimension n and binary structure: $\sigma_i = 1$ if a given feature is included, and $\sigma_i = 0$ otherwise. For any vector σ, data set $\{\mathbf{x}, \mathbf{y}\}$ (y being the targets) and family of regression functions parametrized by a set α, one can write down a loss function L and corresponding risk functional according to

$$R(\alpha, \sigma) = \int L(\alpha, \sigma \odot \mathbf{x}, y)\, dP(x, y) \tag{11}$$

where $\odot$ stands for entry-wise matrix product, and dP is a measure on $\{\mathbf{x}, \mathbf{y}\}$. Then, the objective is to find a risk minimizing vector of indicator variables σ^* (see [31] and references therein).

1 Wrapper framework of embedded methods

Forward selection embedded methods start with a few features and iteratively accumulate more and more of them using specific criteria. Backward elimination methods perform the same operation in reverse. In addition, one can construct a nested procedure, combining feature addition and removal. The learner function, or classification algorithm is being carried along, which is the main distinctive property of the embedded methods [32]. In other words, the performance of a trained classifier for a given σ, uses information on learner and regression functions it acts upon.

Here is how the forward scheme works on archetypal least square example. One starts with a subset of n features, and builds matrix X_S out of them for m training points. To compute residuals, the target vector $Y = \{y, \ldots, y_m\}$ is multiplied with the projection operator $P_S = I - X_S^T \left(X_S X_S^T\right)^{-1} X_S$. Initially, $n = 0$, and once i-th component, which minimizes $\|P_i Y\|^2 = Y^T P_i Y$, and add it to the subset. Then residuals are recalculated and new component added to the set.

A closely related forward method, called Gram-Schmidt Orthogonalization [33] uses angle between feature and target as relevance index (see earlier comments). Algorithm maximizes the cosine of this angle and selects corresponding features. Each iteration uses previously chosen feature. Linear least-square predictor is "embedded", and, therefore, this technique falls into a category of embedded methods.

Alternatively one can start with all the features and backward-eliminate irrelevant part of them. Backward elimination methods are usually performed with the

aid of weight based analysis. Classifier assigns weights to the features., and the idea is to judge features by the effect caused by their removal. For example, Recursive Feature Elimination [27], uses greedy approach for iterative feature removal. It uses SVM classifier, upon finding parameters $\mathbf{w}$ and b, to determine feature with lowest weight, the one causing smallest margin of class separation. The process continues, until only a pre-selected number of features is left. The generalization to the non-linear case is fairly straightforward. The only nuance is that algorithm tries to remove features minimizing the functional $W = \sum_{k,l} \alpha_k \alpha_l k(x_k, x_l)$, where k is a selected kernel and αs are Lagrange multipliers.

An efficient extension of these feature selection algorithms is the Least Absolute Shrinkage and Selection Operator technique (LASSO) [24]. It is all about solving for w^s - minimizing parameters of the problem $\|\sum_k (w * x_k - y_k)^2\|$ subject to sparsity requirements, i.e. to keeping as little as possible of non-zero components. Some of the LASSO approaches produce the weights output interpretable as probabilities [24].

In summary, embedded methods provide a fast access to data understanding through the approximate solutions to optimization problem. Their chief characteristics include optimization over discrete binary set, greedy search procedures, and linear approaches, imposing sparsity of modeling parameters. The embedded methods have more capacity compared to filter methods, but a prone to interpretation. They also lack probabilistic interpretation, unlike the methods from previous subsections.

2 Wrapper framework of ensemble methods

Feature extraction in ensemble methods is closely related to the model selection. Since the base learner in ensemble is rather weak the feature subset of individual learner is unstable [34]. Thus, model-based feature selection would benefit from regularization effect provided by ensemble aggregation [35]. In parallel ensemble the features are selected at random at every bootstrap sample of the data. Errors introduced by every learner are canceled out [34, 35].

The feature selection in serial ensembles is more complex [36]. It consists of several stages: choice of single variable relative importance metrics, then iterative features subset selection with respect to the loss function minimization and finally,

Bayesian voting mechanism is performed on models with different dimensionality, so the final set of features is the set of most probable selected model [35].

Relative importance of the feature in ensemble method is multivariate-model based and thus different from the relevance measured by standard filter methods.

$$VI(x_i, T) = \sum_{t \in T} \triangle I(x_i, t),$$

where $\triangle I(x_i, t) = I(t) - p_L I(t_L) - p_R I(t_R)$ is the decrease in impurity due to an actual or potential split on variable x_i, and p_L, $p_R$are left and right proportions of data points at tree node. Tree is a most common weak learner in ensemble setup. For stochastic tree ensembles of M trees this importance measure is simply averaged over the trees [37]

$$M(x_i) = \frac{1}{M} \sum_{j=1}^{M} VI(x_i, T_j).$$

Note that ensemble approach is fruitful only in situations when different members of the ensemble bring up quantitatively different output. In our main approach we try to eliminate the need for such disagreement by systematically removing such disagreements. We listed main ideas of ensemble methodology to elucidate a valid plane of action in the same data-mining problems we explore.

E Search strategies for filters and wrappers

Given a feature evaluation technique we can start search for the optimal subset, at which point an effective search algorithm is desired. The order of subsets evaluation is referred to as search strategy. The brute force approach to a feature set of n different variables requires $2^n - 1$ subset to go through, which computationally unrealistic. Branch and bound algorithm gives only marginal improvement [25]. And just like in situation where analysis ceases to yield an exact solution, one can attempt to find an approximate solution to optimal feature selection.

Such a "suboptimal" approaches have a task of efficient search for reasonably good features subset, and are, by far, less computationally complex than exhaustive search. The best examples are sequential pruning and sequential growing. They start either with all the variables or from an empty set and move in opposite direction. The

pruning algorithms search for a variable whose removal results in the best evaluation of the return values. By contrast, the growing algorithm seeks the most substantial improvement with the addition of a new variable to the set.

These methods as well as their generalizations represent the so called "greedy" algorithms - they look for a best subset available in the direct vicinity of search, ignoring possibly better choice on distant branches. The alternatives, the beam, floating, and oscillatory methods are somewhat more efficient in terms of finding a global optimum, but still suffer from various method-specific drawbacks - computational load, in particular. Being deterministic in nature, all these approaches produce repeatable results. Yet, in applications, this property is rarely essential, and, as a result, stochastic search methods are preferred.

Stochastic algorithms use random choice at certain stages, overcoming the local minima problem by sampling the search space as efficiently as possible. There is little surprise, that two of the most powerful stochastic search techniques take their roots in natural phenomena: phase transition between liquid and solid, and evolutionary algorithms.

The search method called Simulated Annealing [26] has energy minimization as its underlining idea, and essentially represents a variant of Monte Carlo method. The method exploits the idea of a system coming to its lowest equilibrium with lowering its temperature, just like water does when cooling is imposed. Different states of thermodynamic system correspond to candidate optimization solutions. Temperature does not have a direct analogue, but is understood as a control parameter regulating the probability of the "next step". One starts with "high" value of this parameter, with a random subset of features. Then, this subset is randomly altered, and two "states" are compared. The lowest cost (energy) solution is an ultimate target, with probability of change is not symmetric. The adverse change is more likely at higher value of governing parameter, very much in accord with Boltzmann law in thermodynamics. Trapping in local minima is bypassed, due to non-zero probability of going back to less favorite optimization solution.

Another popular stochastic search method implements the so called Genetic Algorithm [26], which name implies the survival of the fittest. Unlike simulated

annealing, this procedure keeps the entire set of solution vectors (chromosomes) in memory. The set of chromosomes experiences random mutations, as new chromosomes are formed by flipping some of the vector components. Next, the new population is produced by randomly dissecting and gluing together chromosomes from the old population. The selection of "parents" is governed probabilistically, in agreement with renowned law of nature.

F Spectral feature extraction

1 Principal component analysis

Even though, not a feature selection technique, the most relevant tool in the context of present work, is the PCA [13]. Since we are proposing method that keeps all the most relevant features extracted from spectra of data sets, it is more than appropriate to spent some time on classical method that often does not (see earlier discussion in Section 1.1). The starting point for this method is normally a covariance matrix computed either in temporal or spatial domain. The set of features to be extracted through linear transformations is an eigen system - either eigenvectors, or eigenvalues or both. They can be easily ranked, based on their influence in the experimental data, which is again an incidental similarity to methodological framework in earlier exposition.

For example, in mechanical systems, one usually filters out most eigenvalues except for the largest one, which carries most of the dynamics, i.e. contains most of mechanical energy. In structural mechanics, it is sometimes important to be able to damp resonant frequencies, to prevent a building or a bridge from the damage under impulsive load. In biological data, such as gene micro arrays, the "cloud" of genes, projected onto the scatter plot, is stretched towards a few directions, identified by principal components [39]. The knowledge of eigen frequencies of a given structure does not suffice, however, the knowledge of principal component does. But, perhaps, the best illustration of the PCA in action, is its bioinformatics applications [40]. Advantages of the PCA become particularly obvious when one is confronted with massive amount of variables typical in micro-array experiments. Filtering out the

redundant components leads to dimensional reduction and helps in revealing intrinsic patterns in gene expression [39, 40].

Once a covariance matrix is known, one can establish the relations between different variables based on Pearson coefficient. This task is too ambitious, even without data being noisy, due to high-dimensionality of a generic problem. The main idea behind the next step in PCA, the Singular Value Decomposition (SVD) is precisely to move towards lower dimensionality. An n-dimensional covariance matrix C, and as a result, the original data are represented in a standard Euclidean basis, e.g. $\{1,0,0\}, \{0,1,0\}, \{0,0,1\}$, if the space is three-dimensional. Sparseness of the matrix signals alignment of the data along certain direction, but instead of "guessing", one can pass to a more suitable basis, in which matrix becomes diagonal. Such a basis does not necessarily has a simple $\{1,0,0\}, \{0,1,0\}, \{0,0,1\}$ form, but it guarantees the simplest possible representation of the data itself - the diagonal eigen-representation.

In this representation the original matrix C becomes: $C = V\Lambda V^T$, where V and Λ are orthogonal and diagonal matrices respectively; their elements are determined by the system of linear equations $Cv = \lambda v$. Real numbers λ_i, $i = 1, ..., N$ are eigenvalues (singular values), while columns v of V, are eigenvectors (the vectors of the simple-most basis for a given data-structure).

In our specific case of eigen-analysis of covariance matrix, eigenvalues are equal to the variance of the original data, for example time series. Indeed, time averaging implied in computing variance plays role of inner product for two orthogonal eigenvectors. In other instances, eigenvalues may not have such a concrete interpretation.

Furthermore, eigenvectors related to a particular eigenvalue, represent components of a new "preferred" basis with respect to the old, simple-looking one (see earlier example for three-dimensional situation). The eigenvectors of the covariance matrix C are also termed principal components of original data given by (generally) rectangular matrix X. In most applications, only a few components are necessary for adequate data description. Given X is zero-centered and normalized to a unit variance, the two matrices are related according to $C = X^T X/N$, where N is data space dimension.

G Nonlinear spectral feature extraction.

Sometimes, however, linear analysis does very little for the problem of learning. Hence, classical PCA, has to be generalized or abandoned [41]. As we demonstrate in this subsection, the alternative to feature selection, which in this case amount to dimensional reduction can still be achieved, although mathematical rigor and applicational universality is often inadvertently sacrificed. Below we discuss several unsupervised learning algorithms, which all fall into a class of spectral methods. They all employ eigen-analysis, but have more difficult task in mind. A non-linear extension of dimensionality reduction idea aims at finding a nonlinear manifold, which accumulates data in similar fashion the "new basis" (the eigen-basis) does in situations where conventional PCA is appropriate.

1 Kernel PCA.

Mapping of the original data into feature space lies in the foundation of kernel PCA, together with the idea of using distance between vectors of data variables, or their mutual angle or dot product - all three concepts being closely related, as was shown in earlier subsections.

In essence, matrix C is a collection of dot products; therefore, the mapping in question, does not have to be known explicitly, as long as dot product in feature space has known functional form. A new "covariance" matrix $\hat{C}$ reads:

$$\hat{C} = \frac{1}{N}\sum_{i=1}^{N} \Phi\left(x_i\right) \Phi\left(x_i\right)^T, \tag{12}$$

where Φ is unknown nonlinear mapping. Note, that even though $\hat{C}$ in non-linear in original data, the eigenvalue problem can still be formally defined:

$$\hat{C}V = \lambda V \tag{13}$$

or, alternatively,

$$\lambda\left(\Phi\left(x_i\right) \cdot V\right) = \Phi\left(x_i\right) \cdot \hat{C}V \tag{14}$$

Vectors V belong to this newly defined dot-product space, associated with matrix $\hat{C}$, hence can be expanded in terms of vectors $\Phi\left(x_i\right)$: $V = \sum_i \alpha_i \Phi\left(x_i\right)$, which

upon substitution into Eq. (14) produces

$$\lambda \sum_i \alpha_i \Phi(x_k) \cdot \Phi(x_i) = \frac{1}{N} \sum_{i=1} \alpha_i \Phi(x_k) \cdot \sum_j \Phi(x_j) (\Phi(x_j) \cdot \Phi(x_i)). \tag{15}$$

Introducing $K_{ij} = \Phi(x_i) \cdot \Phi(x_j)$ we arrive at

$$M\lambda\alpha = K\alpha, \tag{16}$$

which is of course another SVD equation. Solving for λs and αs we then define projections of data images $\Phi(x)$ onto eigenvectors V via

$$V^k \cdot \Phi(x) = \sum_i \alpha_i (\Phi(x_i) \cdot \Phi(x)), \tag{17}$$

and call them principal components. All we need to know for their explicit calculation is "kernel" matrix K.

Hence, one is able to bypass unknown functional dependence Φ and get an access to the principal components by using the same trick as in non-linear SVM (see, for example [41, 42]). The kernel is selected a priori; typically it is a function of a dot product of original vectors of data. Here is a list of popular choices:

$$K(x, y) = (x \cdot y)^d, \tag{18}$$

$$K(x, y) = \exp\left(-\frac{\|x - y\|}{2\sigma^2}\right), \tag{19}$$

$$K(x, y) = \tanh(\kappa(x \cdot y) + \theta), \tag{20}$$

where σ, κ, and θ are real parameters. Note, that linear PCA is recovered for $k = x \cdot y$.

As was already pointed out, despite its power in many practical data-mining situations [43], PCA has its limitations [44]. Particularly, PCA assumes monitored data as being static. Linearity assumption is also the key. But the most vulnerable is the embedding of statistical importance of mean and variance, or covariance. Once the data fails to be Gaussian or even multi-modal Gaussian, PCA brings up erroneous results in terms of basis axes. Another damaging assumption is the large variance relevance importance. This can only work on *a priori* knowledge of high signal-to-noise ratio.

Non-linear PCA, such as for example KPCA, has an advantage of capturing higher order statistics [45]and thus more detailed information on data set. Another

significant advantage of KPCA is that it works well with linear classifiers, for instance, with SVM. Non-linear kernels we listed above are not in the way. Hence the results of KPCA are generalizable to other classifiers despite non-linearity. Further pedagogical discussion of PCA methodology can be found in [46].

2 Scaling methods

To conclude our discussion of various existing data-mining options, we would also like to touch base on family of scaling methods [47]. To a large extent, scaling approach to data points relies on the inter-point distance as a measure of dissimilarity. In classical scaling such distances are identified with dissimilarities. In multidimensional scaling (MDS), which is the most widely used [47], the relationship is a bit more subtle [14].

Conceptually, MDS can be elucidated with classical Kruskal's example [49] of rail-road construction between certain number of stations. A table of inter-station distances, which might be of interest for such a project does not require to include actual distances (in fact there is no such a thing as actual distance). Instead, it uses a scale - a specific number of kilometers or miles per unit of scale. This table is typically extracted from some sort of photo-graphic map. A lot more involved problem is the object of MDS procedures. It the inverse problem of creating a map out of given table. This problem is further complexified with traditional problem of noise. Not to mention that dimension of map is not known either; hence the name - MDS [49].

Next, to outline, the procedure behind scaling methods, consider p-dimensional data space, containing n points $\mathbf{x}_i$, from which we construct a $n \times n$ matrix B: $B_{ij} = (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_j - \bar{\mathbf{x}})$, where $\bar{\mathbf{x}} = \sum_{i=1}^{n} \mathbf{x}_i/n$ is a vector ensuring zero centering. Eigen-decomposition $B = V\Lambda V^T$, can be further reduced if number of points n exceeds dimensionality of space p, because of the $n - p$ zero eigenvalues. We have $B = V_p \Lambda_p V_p^T$, where now $\Lambda_p = \text{diag}\{\lambda_1, \ldots, \lambda_p\}$ and V_p is $n \times p$ matrix with the eigenvectors, corresponding to these eigenvalues, as columns. And, of course, if, for some reason, we decide to keep k largest eigenvalues only, we reduce above described representation to $B = \hat{X}\hat{X}^T$, in terms of principal components $\hat{X} = V_k \Lambda_k^{1/2}$. The number of components k can be decided on based, for example, on variance participation ratio:

$\sum_{i=1}^{k} \lambda_i / \sum_{i=1}^{p} \lambda_i$. As can be seen from the construction, classical scaling is essentially another view on PCA (and *vice verse*).

This type of scaling targets a configuration of points $\hat{X}$, whose inter-point distances d_{ij} resemble the dissimilarities $\delta_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|$ of data most closely. We specify a functional form f, to which inter-point distances should be a match. This idea is very much in the spirit of kernel PCA [42].

Then, the analogue of SVM Lagrangian, the error (stress) function can be introduced, according to

$$S = \frac{\sum_{i,j} w_{ij} \left(d_{ij} - f\left(\delta_{ij}\right)\right)}{\sum_{i,j} d_{ij}^2}.$$

where w_{ij} are pre-selected weights, for instance $w_{ij} = 1/\delta_{ij}$. The main idea, just as in SVM and many other conditional extremum methods is to determine the points, which define the proffered d_{ij} configuration. And of course, a deck of S-minimization methods exists, with the most popular being gradient descent algorithm [50]. The only MDS assumption is presence of monotonic dependence of projected pairwise distances on the original ones.

The main strengths of MDS are computational efficiency, guaranteed asymptotic convergence, and global optimality. Yet, the MDS methods are highly dependent on the definition of dissimilarity. If we take DNA micro array experiments as an example, the type of features determines distance measure. The abundance of mRNA expressed via transcript levels composes entries of feature vectors, filled either across genes or samples. As in many other cases of correlation-based dissimilarity measures, B is not positive definite. To make any progress, it has to be converted into a positive definite metric in order for the metric MDS (or any other least-square-like method) to work. Any such conversion introduces bogus information into the metric, and, as a result, into the feature set. In addition, all conversion methods are susceptible to gross errors and lost or damaged data ([3] and references therein). Very recently, several non-metric alternatives to MDS have been introduced; see, for example, [3].

H Structure of the thesis

By and large, the strategic approaches to machine learning in feature selection are highly non-universal. Most of the successful applications result from meticulous

comparison and evaluation of different methodologies. In the next Chapter we discuss a family of methods, which are radically different from the discussed so far, in a sense of interaction between learning and feature selection.

In this thesis we propose a viable alternative to the methods discussed. We view our method as a construction upon them. In its technicalities, the algorithm we built is directed toward least amount of steps. It also leverages on general properties of many known complex systems to exhibit very specific singular value statistics. In order to appreciate the vigor of proposed lay-out it is necessary to look into all the spectrum of alternatives.

Having described feature extraction methodology in machine learning, we now turn to the work we have done on developing a new approach to a specific data mining example of network traffic. We start the next Chapter with general description of the existing network models. Even though we do not offer any network modeling ourselves, this step is meant to provide insights on results of our data analysis. Chapter II is also partially devoted to the review of complex systems. By building a link between hard-to-describe systems and complexity of network traffic, we introduce main methodology we use throughout this thesis.

In Chapter III we introduce and briefly state main relevant results of the theory of random matrices (RMT) [51, 52]. Spectral properties of random Wishhart matrices have direct connection to the corresponding properties of Pearson's coefficient matrices. This can be easily inferred from the very way these matrices are built. Hence, signatures of the RMT are anticipated in the eigen statistics of typical correlation matrices for time series or gene micro arrays. Financial time series had already backed up this hypothesis on many occasions [56, 57, 58].

Then, in Chapter IV we construct a feature selection algorithm and run series of statistical tests, to demonstrate and explain relevance of the RMT to the description of transient network dynamics. In order to fully expose presence of the RMT-like spectral behavior, we run comprehensive statistical tests on original network traffic data. In addition to that, we performed multitude of experiments making use of the knowledge of spectrum separation into the RMT and non-RMT parts.

We proceed with comparison of our method to the related feature selection and

clustering methods in Chapters V and VI. We analyze the robustness, computational complexity, accuracy and other performance characteristics of the respective approaches as well as their mutual correspondence. Specifically, in Chapter V we take on classification problem involving network traffic time series. We compare two linear methods, which we combined with our own reconstruction error technique. The primary goal is to demonstrate absence of data pre-processing and superiority towards computationally complex non-linear methods. An additional goal, was to prove positive role of the RMT methodology in feature selection. In Chapter VI we apply similar ideas to the clustering procedures run on biomedical data set. Here we use RMT de-noising algorithm to uncover existing patterns, which in turn facilitates proper clustering, and thus ensures accurate diagnostics.

Finally, we consider feature extraction in a time-lagged formulation of a network-traffic problem in Chapter VII. We discover an acute sensitivity of chosen features towards alteration of randomly selected time series. In the process, the RMT boundaries are used. Such a property makes these features eligible for the role of disruption of service detectors. To find the best anomaly indicators we carefully studied not only temporal dependence of eigen statistics, but also power spectra of all the features involved.

We summarize the obtained results and derived conclusions in Chapter VIII. This last Chapter also restates main principals behind methodology presented in this work. Several potential applications and possible direction of future research are laid out at the very end.

CHAPTER II

NETWORKS AS COMPLEX SYSTEMS

Understanding of natural, economic or social phenomena presume a definite model, Whether such construct is theoretical or numerical, the general goal is to have as little parameters as possible. Then, the model is viable, and, very often, a set of completely distinct phenomena can be approached with the same set of study tools. Here we certainly assume model's (at least partial) ability of capturing trends in experimental data - e.g. predicting its future or describing past history.

With networks, however, such a universality is hardly expected. The nature of a network plays an integral part in all aspects of modeling. Stock market, viewed as a collection of agents responding to the time series, the price evolution, and behaving in accord with such evolution, is a network. And so are networks of scientific or artistic collaborators, actors, which are systems of sociological nodes and links. The two can only be modeled simultaneously in few contexts, such as graph mining, for instance. Yet the meanings of traffic or evolution in such networks is so distant, that uniform approach to modeling make a little sense.

A Traffic modeling considerations

Our particular focus is on traffic in computer network. Even more specifically, on network of interconnected routers, therefore, we limit our introductory discussion to this subject only. Current general understanding of traffic patterns, instabilities and irregularities, comes from models based on fluid dynamics, directed percolation, random walk and cellular automata. We use some insights provided by this approaches, but generally we will stay away from micro-, meso, and macroscopic models [59] altogether, concentrating on the output time series analysis instead.

Here is an illustrative example of when and how the network traffic becomes

incomprehensibly complex: suppose a human user runs a Web browser from a desktop or laptop and clicks on a link with a location of an object such as a HTML (Hypertext Markup Language) document or another file accessible using Hyper Text Transfer Protocol (HTTP). This starts HTTP request that is passed down to Transmission Control Protocol in the protocol stack. Protocols in the operating system of a laptop or desktop are organized in a partial order represented as a protocol graph. Similar to express-mail services, sending pre-sealed envelopes inside their own envelopes, a protocol graph encapsulates the HTTP request by treating it as payload.

Since Internet is "leaky", i.e. it can loose the packet, TCP memorizes the packet information in the event it needs to resend it. TCP's packet is transferred to IP, a routing protocol. This protocol attempts to forward the packet as close as possible to its final destination. IP encapsulates the packet and hands it to the link layer. A popular link layer is Ethernet. It has an access to physical address at the next destination's IP address. (Network devices possess unique physical addresses.) Link layer attaches envelop with physical addresses and sends it down to the physical layer. The physical layer oversees the transmission of information containing the link layer packet over its communication medium. The physical layer at the receiving end decodes the transmission and does a hand-off to the appropriate link layer protocol above.

Assuming the receiver understands the IP "language", the link layer protocol decapsulates and hands off to the IP layer which determines whether additional forwarding is required to reach the final destination. If this is the case, the packet is encapsulated and passed down the protocol stack. This process is repeated at every IP-enabled device—called router—on the forwarding path until the destination IP device is reached. At that point, the IP layer passes its payload up to TCP which, in turn, hands off its payload to HTTP, and HTTP to its application. In this example, a web server that processes the HTTP request. This prompts HTTP response, which is passed down the protocol stack at the destination IP device and returned to the original sender, the client.

Several things can go wrong on an IP packet's journey. During the transmission, noise may corrupt packet, especially in wireless segments of the Internet. The corrupted packet can later be dropped. After arrival, the packet may face busy router.

Alternatively, there might not be any buffer space which again leads to the packet being discarded. Furthermore, router can malfunction erasing the packet in transition from its memory; IP has no provision for dealing with packet loss. Consequently, TCP running at the sender with the assistance of its counterpart, which runs at the receiver is the closest point for attempted recovery.

In this example we ignored the size of the packet and number of users. It turns out that size and number of users do matter. Even though traffic engineering attempts to flatten the traffic to accommodate predictability, Internet traffic shows the emergent chaotic behavior, which makes Complex System metaphor of Internet traffic more than suitable.

There are no established irreproachable network models to date. Hence it is almost impossible to build explanatory arguments on them. Our perception of what is going on inside network is somewhat close to that of Ref. [60]. We perceive network as a graph, which nodes contain, process and emit different amounts of information along randomly chosen links. Our goal is to find behavior patterns and extract relevant features from time series data, rather than simulate phase transition or any other network phenomena. Yet, seeing network architecture as numbered nodes on two-dimensional lattice, as it is done in Ref. [60] is very convenient. It creates natural site basis. The routers we consider in what follows, also form an interpretable system of exchanging information nodes, which can be studied both in time and in spectral domain. We also adopt probabilistic nature of routing, but do not assume any particular strategy.

Before we do that, it is instructive to look at computer networks from the point of view of deterministic behavior.

Suppose that matrix H describing the routing strategy is written in the site basis. Real-number entries of this matrix relate two indices - raw and column, that is two sites (routers). It tells us how much information is sent to one or the other node. It has little to do with connectivity matrix, and represents strength and speed of interactions in the system of routers. Only zero entries mean absence o communication - the same for both matrices.

The next important point in our development has to do with construction of

routing strategy matrix. It is reasonable to declare no knowledge of actual entries of such matrix. They can be assumed random, are partially random, for instance, distance nodes may be treated as non-interacting, and therefore, matrix H - as sparse. It is clearly symmetric: $H_{ij} = H_{ji}$, due to network reciprocity, but otherwise is quite generally defined.

To each such matrix, i.e. to each given network, we can associate a so called "propagators", a notion frequently used in many-particle systems in general and in hopping models of network traffic, in particular [59]. These are elements of matrix $\mathcal{G} = (E\mathbf{I} - H)^{-1}$, where $\mathbf{I}$ is an identity matrix, and E is a spectral variable. Propagators determine probability of a packet of information to end up at site j after being emitted at site i. In time domain, such probability is computed through the inverse Fourier transform. The probability of hopping from i to j at time t since routing from i is given by the following convoluted transformation into a time domain

$$P(i;j;t) = \int \mathcal{G}_{ij}(E+\Omega)\,\mathcal{G}^{*}_{ij}(E-\Omega)\exp\{-i\Omega t\}\,d\Omega, \tag{21}$$

which ensures real and positive definite value of P. The proper upper bound of probability is easy to achieve, as variables E and H_{ij} are rescalable.

Now, we can relate intrinsic network variables, such as rate of information transfer at a given node in a given direction to measurable quantity. This observable could be an aggregated average of the amount of information passing through a given node. In fact, the latter is actual data set available in practice. In our present study we run feature extraction on one such data collection.

Consider instantaneous number of bites δI_j passing thorough the node j; the aggregated average is simply this quantity accumulated over time Δt, divided by Δt. To obtain δI_j, we calculate the total probability of information going towards all other nodes, and subtract it from the probability of information to be acquired from all the other nodes:

$$\delta I_j = \sum_i P(i,j;t) - \sum_i P(j,i;t) \tag{22}$$

Combining Eqs. (21) and (22) we establish formal relation between internal properties, characterized by H_{ij} and router recordings $g_i(t)$, potentially available for most of the router networks, router-host combination structures and, in principle, for the Internet.

Our goal is, of course, not to get a better grip on such relations but rather elucidate several phenomenological facts. For one, randomness in routing strategy and capability, expressed by the coefficients H_{ij}, will reappear in stochasticity of time series $g_i(t)$. And second, the nature of interactions in the network is clearly non-linear. Although linear correlation relevance index can be illustrative, its direct interpretation is obscure and hardly reliable.

Indeed, the propagator $\mathcal{G}_{ij}$ represents a pulse traveling from site i to site j through all possible paths on two-dimensional lattice. It means, that certain averaging procedure has to be applied to calculation of $\mathcal{G}_{ij}$ to compensate for our absence of knowledge the exact path. Despite some routers may have consistently non-random strategy, realized, for example, by sending the same amounts of information to the same neighbor, the rest of the sites will necessarily destroy any deterministic trends in traffic between i and j, or any other pair of sites for that matter. And finally, since $\mathcal{G}_{ij}$ is an element of the inverse matrix $(E\mathbf{I} - H)$, it literally depends on all interaction constants H_{ij}. Thus, not only the statistical approach should go beyond linearity, but it should also address fundamental issues of understanding a complex system through data mining.

B Spatio-temporal chaos and network communications

Whereas a real dynamical system, is described by differential equations and continuous time, sometimes, for periodic or quasiperiodic motions, it makes sense to turn a real variable t into an integer and use iterative maps. In fact, variable t is always integer in practice due to discretization. But whenever dynamical system repeats itself over and over again, storing more than one period of information is waste of, literally time (and data-space). These is also true for nearly periodic processes.

With t being integer, the system evolves through a set of discrete time steps, becoming an iterative map of the form $x_{n+1} = f(x_n)$. Such maps capture changing traffic patterns in networks, and often used in modeling [59], as networks dynamics easily fit the description of dynamical systems, linear maps or flows. Indeed, on short time scale the router recordings look like superpositions of on-off processes.

If mapping function $f(x)$ is not linear, or the differential equation behind

dynamical system has non-linear terms, the closed form solution for a given initial condition does not exist, unless searched numerically. Such non-integrability displays itself through noise-like oscillatory behavior of the time series with a high degree of self-similarity. The uncanny appearance of map snapshots is just another testimony to the affluence of dynamics imposed by the non-linearity, non-integrability and non-determinism.

In network communications, function f is unknown or two complex to be expressed by formula. That is why cellular automata (CA) approach works so well for traffic problems [61]. First of all, many degrees of freedom are explicitly represented, the miscellaneous rules of behavior for every variable of the system are specified. And second, the CA capture visual representation of the system's evolution. Being an alternative to differential equations for the modeling of physical systems the CA inherit the concepts of space, divided into the cells, and influence of neighborhood. By studying the CA generated patterns, one can access the randomness, dynamic response and change in time of collective behavior of mutually interacting network sub parts. Further illustrative examples of the CA network use, include eco- and meteo- systems, financial markets, human neural networks, social and technological networks. There is however major alternative to both of the described approaches to complex graph structure. It comes from the field of chaology, the science of chaos, which we now quickly discuss.

The most recent account of the field of complex systems, complexity and chaos has been put forward by a collective effort in special edition of Science [62]. The subject is broad and almost philosophical in nature, still a few main premises for bringing in the chaology need to be stated.

Conventionally, a complex system is the one for which the number of independent components is large, or one in which multiple evolution pathways exist. A complex system is the one with multitude of interactions between constituents. And, furthermore, a complex system constantly unfolds and changes in time.

Within this definition, the complex system in general, and network in particular, may contain stochastic and deterministic components, or just one type of them. They all manifest themselves through static or dynamic output, e.g. time series collected at

various junctions of a network. Even a slightest proportion of chaos, which can be linked to non-integrability, causes unpredictable evolution. But the degrees of chaos vary.

The way chaologists define this degree relates to the way system explores its phase space - whether it explores this space in its entirety, or simply looses the place it once was occupying. From the point of view of machine learning, the proportion of chaotic and deterministic dynamics can be estimated with the use of the concepts of entropy or information (see Chapter I).

On the level, it is interested to us, any congestion and localization, any disruption of information flow are signatures of non-chaotic behavior. On the other hand, equilibration and absence of any special distance or time scale are features of spatio-temporal chaos. Coherent and synchronized patterns are viewed as signs of deterministic behavior, while unstable and unpredictable exchange between the nodes are considered to be reflections of ergodicity.

In the next Section we introduce our own criteria of distinguishing between chaotic and deterministic features. Our intentions are, of course, different from drawing a rigorous mathematical boundary, Altogether, the goals are to find statistically sound machine learning algorithm, which sets up feature extraction and enables anomaly detections in the regime of spatio temporal chaos. The method we discuss in the next Section has been successfully applied in the stock market studies, but received little to no attention in the context of network traffic.

CHAPTER III

RANDOM MATRIX THEORY

Large matrices appear frequently in analytical and numerical studies, which involve large experimental or synthetic data arrays. The covariance matrix, filled with Pearson coefficients could be a perfect example (Section 1.1.4.). The large dimensions of matrices, further result in voluminous eigen systems. This is a typical problem in machine learning anyway, however the theory of large matrices originated in a different field of science.

In physics, especially in nuclear, condensed matter and microwave experiments, the spectral data, i.e. experimentally measured eigenvalues is enormously large. Consequently, physicists first attempted to apply conventional methods of statistical mechanics, in which one system, e.g. nuclei, is replaced by an ensemble of similar systems, governed by the same matrix of interactions, the Hamiltonian. The ensemble provides a way of computing statistical averages, which, however, is too formal to be realized in any practical situation due to the lack of the knowledge of Hamiltonian.

In classical RMT [51, 52], such ignorance is turned into a basic premise. The system in question is considered on its own, but the unknown Hamiltonian is substituted with ensemble of large random matrices, which possesses the same physical symmetry as actual system [51]. The actual "degree" randomness of the matrix ensemble is unimportant and the underlying distribution of matrix elements, as well as probability measure for computing averages is normally chosen in accord with mathematical convenience.

Nowadays, the RMT represents highly developed area of mathematical physics, still build on the above sketched unorthodox statistical hypothesis made by Wigner [52] but with much more sophisticated machinery in hand [63]. The theoretically predicted and experimentally found [63] connection between universal spectral statistics and transport properties of various complex systems helped to create a link

to general structures with unknown or intricate interactions, such as computer and gene expression networks. Because of that, the RMT achieved a widespread success not only in several branches of physics of complex systems, but also in economical and social sciences. Many of the elegantly universal signatures of the RMT can be found in experimental and synthetic data sets analyzed for highly practical tasks of, for instance, portfolio risk assessment [64, 65], traffic regulation and human brain diagnostics. With this facts in mind, we review basic notions of the RMT assuming no additional knowledge of the reader, other than already discussed eigenvalues.

A Basics of the RMT

1 The foundations of the classical RMT

Applications of this purely mathematical phenomenology are growing faster and faster day by day. Findings by P. Sheba in [53] show that city center curb parking and starling flocks congregating on power lines to be two recent RMT trophies. Before that, came financial time series [64], Mexican public transport [54], human brain waves [55] and many others [63]. But we shall start the story where it officially began - in nuclear physics.

Compound nuclei, as well as large atoms and molecules is impossible to consistently describe in deterministic way due to large number of their constituents. Particles in these systems interact in some complex and often unknown way. As a result, even a theory based on classical statistical mechanics runs into a uncircumventable problem from very beginning. It is equally not possible to write down manageable system of governing equations and to explain the neutron-scattering resonance data obtained off of nuclei or atoms. The key difficulty is absence of any information apart from physical symmetries about the Hamiltonian matrix (analog of matrix H in earlier discussion) for such multi-particle systems. Thus, the RMT originated from an attempt to analyze resonance positions, directly linked to the eigenvalue spectra of compound atom or nuclei.

The way out of analytical and conceptual dead-end, was found by Wigner, Dyson and their collaborators in their classical formulation of the RMT, see historical

overview and recent developments in [66]. They decided to disregard all information about the system with the exception of the symmetries. The Hamilton matrix is replaced by the ensemble of matrices filled with random numbers and specific symmetries of the physical problem are taken into account in the way matrix is being filled. For instance, if the processes in atom will take their turn in exactly the same way as before, after the time starts flowing backwards, the matrix will be real and symmetric. Note, that generic Hamiltonians are rather sparse [67], while members of the ensemble are filled uniformly with statistically uncorrelated elements distributed according to:

$$P(H) \sim \exp\left\{-\frac{N}{4}\mathrm{Tr}\, H^2\right\}, \tag{23}$$

where Tr stands for trace of $N \times N$ matrix H (trace of an n-by-n square matrix A is defined to be the sum of the elements on the main diagonal), and V is an arbitrary even real polynomial in H. Here, $P(H)$ is short for probability distribution function of all of the matrix entries. The integration measure, used in calculation of statistical averages, is a product of all of the independent differentials $\mathrm{d}H_{ij}$.

This sequence of RMT assumptions and principles seems a little *ad hoc*, and to some degree it is. Yet, the entire machinery has a perceptible illustration, which originates in information theory and shows how assumptions affect methodology. Consider a toy problem, in which a node in a network sends packets of information in the direction of N different neighbors. Each neighbor receives N_i packets with I_i total information content in them. Suppose, we would like to know the probability distribution of packets according to their information content if the most efficient manner is the one minimizing the entropy $S = -\sum_i p_i \ln p_i$, where $p_i = N_i/N$. Such strategy is, of course, the same as we seen before in machine learning problems (Section I), i.e. the one maximizing information $I = -S$. If for some reason our goals are completely opposite, the ensuing logic still works.

As p_i stands for probability of finding N_i packets outgoing in ith direction, it should sum to one

$$\sum_i p_i = 1. \tag{24}$$

If in addition we want average information transfer to be at a certain pre-specified level

$\overline{I}$, the following is also true:

$$\sum_i I_i p_i = \overline{I}. \tag{25}$$

The conditional extremum conditions are given by

$$\frac{\partial}{\partial p_i}\left[S + \alpha\left(\sum_i p_i - 1\right) + \beta\left(\sum_i I_i p_i - \overline{I}\right)\right] = 0, \tag{26}$$

where α and β are Lagrange multipliers. which leads to Boltzmann distribution

$$p_i = \exp\left\{-\alpha - \beta I_i\right\}. \tag{27}$$

Now assume, that we have the same simplistic goal of deciding on distribution function for a continuous variable H, (ranging between minus and plus infinity,) guided by chosen average μ and variance $\mu^2 + \sigma^2$. Retracing the above described steps, we arrive at (this time we have three Lagrange multipliers)

$$p\left(x\right) = \exp\left\{-\alpha - \beta x - \gamma x^2\right\} = \frac{\mathrm{e}^{-(x-\mu)^2/\sigma^2}}{\left(2\pi\sigma^2\right)^{1/2}}, \tag{28}$$

which is nothing but Gaussian distribution. In other words, minimum information (maximum entropy) requirement, which is one of the basic premises for the RMT, leads to the Central Limit Theorem; at least in the case of one-dimensional matrix H. Generalization for any other dimension is more involved, but the result, maximum entropy matrix H can be guessed right away

$$p\left(H_{ij}\right) = \exp\left\{-\alpha - \beta \mathrm{Tr} H - \gamma \mathrm{Tr} H^2\right\}, \tag{29}$$

which after a immaterial shift along the real axis results in Eq. (23). To summarize, the choice made by the RMT founders was two-fold: minimum number of parameters and distribution corresponding to maximum entropy. Choosing trace Tr as a metric for matrix H can be derived rigorously. It can also be easily understood from another basic idea of the RMT: There is no preferred basis in the space of matrices H, hence their probability distribution must be invariant under rotational transformations.

Surprisingly, despite declaring complete ignorance towards actual details of system dynamics, the RMT is able to capture correctly spectral fluctuations of heavy nuclei [68], and of many other complex systems [69]. With the emergence of "quantum chaology" [69], the numerical and laboratory experiments, and in particular, the

Bohigas–Giannoni–Schmit conjecture [70] demonstrated that RMT can be used far beyond the statistical studies of nuclear scattering processes. The most impressive outcome is the universality of the RMT statistics in a sense, that it can be found in systems, which have literally nothing to do with each other, with the exception of their complexity, non-integrability or genuine randomness.

After this fact has been firmly established, random matrix ensembles became major theoretical tools for statistical estimates to be superimposed with experimental running averages. Since most physical systems are represented by real symmetric matrix, we only consider the so called Gaussian Orthogonal Ensemble (GOE) in what follows. The word "Gaussian" means that V is quadratic ($V(H) = H^2/v^2$) , i.e. all H_{ij} are taken from normal distribution with standard deviation equal to v. Any member H of such ensemble can be brought into specific diagonal form with the help of orthogonal matrices $U^T = U^{-1}$:

$$H = U^{-1}\mathrm{diag}\left\{\lambda_1, \lambda_2, \ldots, \lambda_N\right\} U,$$

which explains word "Orthogonal" in GOE. Real numbers λ are called eigenvalues, while columns of matrix U are called eigenvectors.

2 Universal eigen statistics: Summary of the RMT results

The eigen system of any matrix problem is defined according to

$$Hu = \lambda u, \tag{30}$$

where H is matrix of interest, u its eigenvector, and λ its eigenvalue. In other words, matrix H upon "acting" on some vector u simply changes its length, but not the direction. The eigenvalue λ is strain coefficient.

If we "sandwich" matrix H between two eigenvectors u_j and $u_k$corresponding to jth and kth eigenvalues, the following properties:

$$u_k^T H u_k = \lambda_k, \tag{31}$$

$$u_j^T H u_k = 0,\ j \neq k, \tag{32}$$

can be obtained from Eq. (30) and the ortonormality of eigenvectors ($u^T u = 1$).

As we turn to the RMT eigen statistics, several distinct features should be stressed. Although, matrices H taken from GOE have entries from random distribution, it does not lead to the arbitrarily distributed uncorrelated random eigenvalues $\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$ random numbers. Consequently, the nearest neighbor spacings (NNS) of two consecutive eigenvalues: $s_i = \lambda_{i+1} - \lambda_i$ do not follow Poisson law

$$P(s) = \exp(-s). \tag{33}$$

The later is characteristic for memoryless process, as if chimpanzee were throwing darts at the stock market quotes, and later on we would project the holes in the newspaper onto a straight line. The eigenvalues of random matrices instead, obey

$$P(s) = \frac{\pi}{2} s \exp\left(-\frac{\pi}{2} s^2\right), \tag{34}$$

the distribution called "Wigner surmise" [68, 69], which shows faster (compared to Poisson spectra) decay, and vanishes as s goes to zero. It means that eigenvalues "know" about their location in spectrum, they do not stay neither too close to each other nor too far. In other words, eigenvalues of GOE are correlated - the darts are landing trying to avoid already taken positions on the projection line. Or, perhaps, chimpanzee throws the darts in conscious, although slightly random manner.

Another prominent feature of GOE system is rather simple eigenvector statistics. The eigenvector components are all Gaussian uncorrelated random numbers [68, 69]. It once again, signifies absence of any favored basis in the space of random matrices [68, 69]. We discuss the rest of the known signature RMT correlations later on when we run statistical tests on our data.

CHAPTER IV

FEATURE SELECTION WITH THE RMT

The inclusion of the RMT into feature selection process is a powerful way for distinguishing system-specific, non-random properties embedded in complex systems from random noise. Recent surge of RMT financial applications, particularly in portfolio optimization and stock-stock correlation problems [64], is a primary affirmation.

The bioinformatics application could be the RMT next success. Defining co-expression networks without ambiguity based on genome-wide micro array data is particularly problematic since not much existing biological knowledge can be exploited on the basis of threshold between true correlation and noise. Hence, searching for universal predictions of the RMT in biological systems may bypass the correlation threshold problem. The minimal set of features can be determined by "cleansing" the correlation matrix of micro array profiles by "subtracting" known RMT spectral correlations from those pertinent to the system. This sort of procedure was done by brute force in Ref. [71] where the authors attempted to study the gene co-expression networks and predicted functions of unknown genes using Wigner surmise and Poissonian NNS as relevance indices. Unfortunately, level repulsion can be found in many spectra, even those without any ergodic behavior. In other words a single test is inconclusive, as well as the procedure grounded on it. Additional tests are discussed at length in the next two sections, on the specific example of time series data. Just as in financial applications and in Ref. [71], the object of interest is covariance matrix.

Hence, we begin by investigating the network traffic with the goal of finding traces of the RMT spectral correlations. The methodology we built in this work provided the unique possibility to accomplish several tasks of traffic analysis. We attempt to verify the uncongested state of the network, by establishing the profile of random interactions. Then, we worked out the way of revealing the system-specific

large-scale correlations, and establishing the profile of stable in time non-random interactions. Trough the analysis of eigenvalues and eigenvectors statistics, which goes far beyond regular PCA or its kernel variants, we were able to detect and allocate in time and space the anomalies of network traffic interactions.

After establishing the boundaries of noisy and meaningful interactions in spectra of a complex data set, we observe changes in time of meaningful interactions using the time-lagged correlation matrices of the system. The choice of relevance indices is not unique, but the logistics of feature selection is largely the same, guided by the RMT/non-RMT distinction.

A Feature construction using covariance matrix of traffic time series

The infrastructure, applications and protocols of the system of communicating computers and networks evolves constantly. Despite simple outcomes, the traffic data, generated on minute-by-minute basis within multi-layered structure by different applications and according to different protocols is a reliable suspect for RMT-like spectrum.

The first main approach to traffic analysis focuses on protocols, traffic and routing matrices - every aspect of packets propagation. The second one treats infrastructure, between the points of a complex, and essentially random graph, as a "black box" [72, 73]. Measuring interactions between logically and architecturally equivalent substructures of the system is a natural extension of the latter approach.

Certain amount of work in this direction has already been done. Studies on statistical traffic flow properties revealed the "congested", "fluid" and "transitional" regimes of the flow at a large scale [74, 75]. The observed collective behavior suggests the existence of the large-scale network-wide correlations between the network sub parts. Indeed, the work of Ref. [76], where the RMT is first applied to routing data, showed the large-scale cross-correlations between different connections of the Renater scientific network. Furthermore, the analysis of correlations across simultaneous network-wide traffic, done in [77], and [78] for network distributed attacks and traffic anomalies detection respectfully, suggests, that such correlations can be given a meaning.

Among numerous types of traffic monitoring variables, we choose time series of traffic counts and construct covariance matrix C out of them. The procedure is fairly standard in quantitative finance literature. First of all, we take N traffic counts time series of L time points, and calculate traffic rate increments of every time series T_i $i = 1, \ldots, N$, over a time scale Δt,

$$G_i(t) \equiv \ln T_i(t + \Delta t) - \ln T_i(t). \tag{35}$$

This measure is independent from the volume of traffic exchange and is sensitive to the slightest changes in the traffic rate [76]. Next we normalize traffic rate change is

$$g_i(t) \equiv \frac{G_i(t) - \langle G_i(t) \rangle}{\sigma_i}, \tag{36}$$

where $\sigma_i \equiv \sqrt{\langle G_i^2 \rangle - \langle G_i \rangle^2}$ is the standard deviation of G_i. The equal-time cross-correlation matrix C can be computed as follows

$$C_{ij} \equiv \langle g_i(t)\, g_j(t) \rangle. \tag{37}$$

In other words our initial targets are linear correlations, that is we choose the Pearson coefficient C_{ij}, as our first relevance index, along the lines of methodology described in Chapter I.

In matrix notation, the interaction matrix C can be expressed as

$$C = \frac{1}{L} G G^T, \tag{38}$$

where G is $N \times L$ matrix with elements
$\{g_{im} \equiv g_i(m \triangle t); i = 1, \ldots, N;\ m = 0, \ldots, L-1\}$, and G^T denotes the transpose of G.

B Eigen system of the matrix C and its interpretation

At any time the readings from the network nodes give an instantaneous traffic load pattern. This pattern can be viewed as an expansion in terms of eigenvectors of matrix C in the following sense. An eigenvector u_k is a set of different intensities of network-wide traffic load satisfying

$$C u_k = \lambda_k u_k.$$

Among possible configurations of network-wide traffic load u_k^i is an amount of traffic load on a particular node. Then, ratio u_k^i/G_k is equal to the number of nodes involved in the mutual interaction. For a variance of a traffic load at a given node we get:

$$\sigma_k^2 = \left\langle \left(\sum_{i=1}^{M} \frac{u_k^i}{G_i} \delta G_i \right)^2 \right\rangle = \sum_{i,j=1}^{M} u_k^i u_k^j C_{ij} = u_k^T C u_k. \tag{39}$$

At this point we can employ the result of Eq. (31) to realize, that the variance of the traffic load at a given node is specified by the corresponding eigenvalue: $\sigma_k^2 = \lambda_k$. Once again, this is true for a network-wide traffic described by the u_k. By contrast, there is no correlation between two network-wide traffic loads attributed to two eigenvectors u_k and u_l:

$$\left\langle \left(\sum_{i=1}^{M} \frac{u_k^i}{G_i} \delta G_i \right) \left(\sum_{j=1}^{M} \frac{u_l^j}{G_j} \delta G_j \right) \right\rangle = u_k^T C u_l = 0, \; b \neq l.$$

With this in mind we proceed with the thorough analysis of the real traffic pattern.

C How to test the eigen system against the RMT predictions

Just as was done in [65], we consider a random correlation matrix

$$R = \frac{1}{L} A A^T, \tag{40}$$

where A is $N \times L$ matrix containing N time series of L random elements $a_{i\,m}$ with zero mean and unit variance, which are mutually uncorrelated as a null hypothesis.

Statistical properties of the random matrices R have been known from earlier works [79, 80]. In particular, it was shown analytically [80] that, under the restriction of $N \to \infty$, $L \to \infty$ and providing that $Q \equiv L/N (> 1)$ is fixed, the probability density function $P_{rm}(\lambda)$ of eigenvalues λ of the random matrix R is given by

$$P_{rm}(\lambda) = \frac{Q}{2\pi} \frac{\sqrt{(\lambda_+ - \lambda)(\lambda - \lambda_-)}}{\lambda} \tag{41}$$

where λ_+ and λ_- are maximum and minimum eigenvalues of R, respectively and $\lambda_- \leq \lambda_i \leq \lambda_+$. λ_+ and $\lambda_-$are given by

$$\lambda_\pm = 1 + \frac{1}{Q} \pm 2\sqrt{\frac{1}{Q}}.$$

All the pedagogical details are presented in [79].

The first step in testing the data against the random matrices is to find a transformation called "unfolding", which maps the eigenvalues λ_i to new variables, "unfolded eigenvalues" ξ_i, whose distribution is uniform [51, 68, 69]. Unfolding ensures that the distances between eigenvalues are expressed in units of *local* mean eigenvalues spacing [51], and thus facilitates the comparison with analytical results.

We define the cumulative distribution function of eigenvalues, which counts the number of eigenvalues in the interval $\lambda_i \leq \lambda$,

$$F(\lambda) = N \int_{-\infty}^{\lambda} P(x)\, dx, \tag{42}$$

where $P(x)$ denotes the probability density of eigenvalues and N is the total number of eigenvalues. The function $F(\lambda)$ can be decomposed into an average and a fluctuating part,

$$F(\lambda) = F_{av}(\lambda) + F_{fluc}(\lambda), \tag{43}$$

Since $P_{fluc} \equiv dF_{fluc}(\lambda)/d\lambda = 0$ on average,

$$P_{rm}(\lambda) \equiv \frac{dF_{av}(\lambda)}{d\lambda}, \tag{44}$$

is automatically average eigenvalues density. The unfolded eigenvalues are then given by

$$\xi_i \equiv F_{av}(\lambda_i). \tag{45}$$

Eigen system for matrix C is not expected to obey properties of GOE. The eigen statistics of C is contrasted with the eigen statistics of a matrix taken from the so called "chiral" Gaussian Orthogonal Ensemble [65]. Three known universal properties of "chiral" GOE matrices are [79, 80]: (i) the distribution of nearest-neighbor eigenvalues spacing given by

$$P_{GOE}(s) = \frac{\pi s}{2} exp\left(-\frac{\pi}{4}s^2\right), \tag{46}$$

(ii) the distribution of next-nearest-neighbor eigenvalues spacing, being identical to the distribution of nearest-neighbor spacing of Gaussian symplectic ensemble (GSE) [69],

$$P_{GSE}(s) = \frac{2^{18}}{3^6\pi^3} s^4 exp\left(-\frac{64}{9\pi}s^2\right) \tag{47}$$

and finally (iii) the specific behavior of "number variance" statistics Σ^2.

The latter is defined as the variance of the number of unfolded eigenvalues in the intervals of length l, around each ξ_i [69, 51, 68].

$$\Sigma^2(l) = \left\langle [n(\xi, l) - l]^2 \right\rangle_\xi, \tag{48}$$

where $n(\xi, l)$ is the number of the unfolded eigenvalues in the interval $\left[\xi - \frac{l}{2}, \xi + \frac{l}{2}\right]$. The number variance is expressed according to

$$\Sigma^2(l) = l - 2\int_0^l (l - x)\, Y(x)\, dx, \tag{49}$$

with $Y(x)$ for the GOE case is given by [51]

$$Y(x) = s^2(x) + \frac{ds}{dx}\int_x^\infty s(x')\, dx', \tag{50}$$

and

$$s(x) = \frac{sin(\pi x)}{\pi x}. \tag{51}$$

Just as was stressed in Subsection 1.2.3 (see also [65, 81, 63]) the overall time of observation is crucial for explaining the empirical cross-correlation coefficients. On one hand, the longer we observe the traffic the more information about the correlations we obtain and less "noise" we introduce. On the other hand, the correlations are not stationary, i.e. they can change with time.

To differentiate the "random" contribution to empirical correlation coefficients from "genuine" contribution, we contrast the eigenvalues statistics of C with the eigenvalues statistics of a correlation matrix taken from the so called "chiral" Gaussian Orthogonal Ensemble [65].

A *random* cross-correlation matrix, which is a matrix filled with uncorrelated Gaussian random numbers, is supposed to represent transient uncorrelated in time network activity, that is, a completely noisy environment. In case the cross-correlation matrix C obeys the same eigen statistical properties as the RMT-matrix, the network traffic is equilibrated and deemed universal in a sense that every single connection interacts with the rest in a completely chaotic manner. It also means a complete absence of congestions and anomalies.

Meantime, any stable in time deviations from the *universal* predictions of RMT signify system-specific, nonrandom properties of the system, providing the clues about the nature of the underlying interactions. That allows us to establish the profile of system-specific correlations.

D Traffic count data

In this Section we unveil the technical details of the data used to construct correlation matrix C. We assembled averaged traffic count data from all router-router and router-VLAN subnet connections of the University of Louisville backbone routers system. This system consists of nine interconnected multi-gigabyte backbone routers, over 200 Ethernet segments and over 300 VLAN subnets. We collected the traffic count data for 3 months, for the period from September 21, 2006 to December 20, 2006 from 7 routers, since two routers are reserved for server farms. The overall data amounted to approximately 18 GB.

The traffic count data is provided by Multi Router Traffic Grapher (MRTG) tool that reads the SNMP traffic counters. The MRTG log file never grows in size due to the data consolidation algorithm. It contains records of average incoming, outgoing, max and min transfer rate in bytes per second with time intervals 300 seconds, 30 minutes, 1 day and 1 month. We extracted 300 seconds interval data for seven days. Then, we separated the incoming and outgoing traffic counts time series and considered them as independent. For 352 connections we formed $L = 2015$ records of $N = 704$ time series with 300 seconds interval.

Next, we watched changes in the traffic rate, excluding from consideration the connections, in which channel was open but the traffic was not established or there was a constant rate and equal low amount test traffic. Additional reason for excluding the "empty" traffic time series was making the time series cross-correlation matrix unnecessary sparse. After the exclusions the number of the traffic time series became $N = 497$.

To calculate the traffic rate change $G_i(t)$ we used the logarithm of the ratio of two successive counts. As it is stated earlier, *log*-transformation makes the ratio independent from the traffic volume. We added 1 byte to all data points, to avoid

manipulations with $log\,(0)$, in cases where traffic count is equal to zero bytes. This measure did not affect the changes in the traffic rate.

E Results of comparison with the RMT

Once the cross-correlation matrix C of traffic time series is constructed, we compare its eigen statistics with the predictions of the RMT, to establish the boundaries of the random, noisy interactions and extract the meaningful features of the system. We constructed inter-VLAN traffic cross-correlation matrix C with number of time series $N = 497$ and number of observations per series $L = 2015$, $(Q = 4.0625)$ so that, $\lambda_+ = 2.23843$ and $\lambda_- = 0.253876$. Our first goal is to compare the eigenvalue distribution $P(\lambda)$ of C with $P_{rm}(\lambda)$ [64].

We compute eigenvalues of C using standard *MATLAB* function. The empirical probability distribution $P(\lambda)$ is given by the histogram $P(\lambda)$ displayed in Figure 2.1 and compare it to the probability distribution $P_{rm}(\lambda)$ taken from Eq. (41) calculated for the same value of traffic time series parameters $(Q = 4.0625)$. The solid curve demonstrates $P_{rm}(\lambda)$ of Eq.(41). The largest eigenvalue shown in inset has the value $\lambda_{497} = 8.99$. We zoom in the deviations from the RMT predictions on the inset to Figure 2.1.

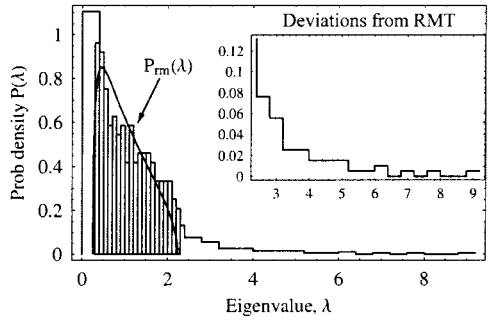

Figure 3. Empirical probability distribution function $P(\lambda)$ for the inter-VLAN traffic cross-correlations matrix C (histogram).

We note the presence of "bulk" (RMT-like) eigenvalues which fall within the bounds $[\lambda_-,\lambda_+]$ for $P_{rm}(\lambda)$, and presence of the eigenvalues which lie outside of the

"bulk", representing deviations from the RMT predictions. In particular, largest eigenvalue $\lambda_{497} = 8.99$ for seven days period is approximately four times larger than the RMT upper bound λ_+.

The histogram for well-defined bulk agrees with $P_{rm}(\lambda)$ suggesting that the cross-correlations of matrix C are mostly random. We observe that inter-VLAN traffic time series interact mostly in a random fashion.

Nevertheless, the agreement of empirical probability distribution $P(\lambda)$ of the bulk with $P_{rm}(\lambda)$ is not sufficient to claim that the bulk of eigenvalue spectrum is random. Therefore, further RMT tests are needed [65].

To do that, we obtained the unfolded eigenvalues ξ_i by following the phenomenological procedure referred to as Gaussian broadening [82], (see [82, 83, 65, 81]). The empirical cumulative distribution function of eigenvalues $F(\lambda)$ agrees well with the $F_{av}(\lambda)$ (see Figure 2.2), where ξ_i obtained with Gaussian broadening procedure with the broadening parameter $a = 8$. The first independent

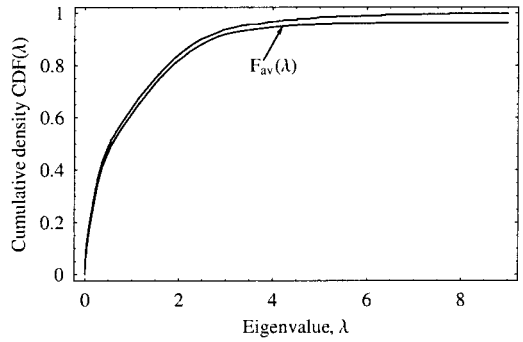

Figure 4. The empirical cumulative distribution of λ_i and unfolded eigenvalues $\xi_i \equiv F_{av}(\lambda)$.

RMT test is the comparison of the distribution of the nearest-neighbor unfolded eigenvalue spacing $P_{nn}(s)$, where $s \equiv \xi_{k+1} - \xi_k$ with $P_{GOE}(s)$ [69, 51, 68]. The empirical probability distribution of nearest-neighbor unfolded eigenvalues spacing $P_{nn}(s)$ and $P_{GOE}(s)$ are presented in Figure 2.3. The Gaussian decay of $P_{GOE}(s)$ for large s suggests that $P_{GOE}(s)$ "probes" scales only of the order of one eigenvalue spacing. The agreement between empirical probability distribution $P_{nn}(s)$ and the distribution of nearest-neighbor eigenvalues spacing of the GOE matrices $P_{GOE}(s)$

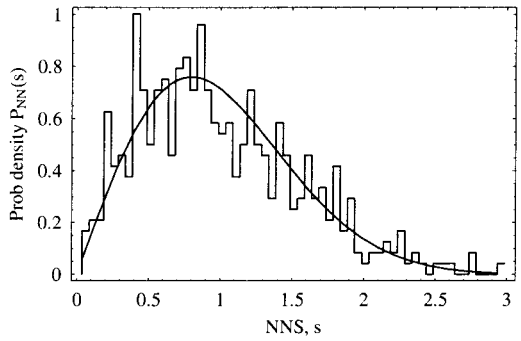

Figure 5. Nearest-neighbor spacing distribution $P_{nn}(s)$ of unfolded eigenvalues ξ_i of cross-correlation matrix C.

suggests, that the positions of two adjacent empirical unfolded eigenvalues at the distance s are correlated just as the eigenvalues of the RMT matrices.

Next, we took on the distribution $P_{nnn}(s')$ of next-nearest-neighbor spacings $s' \equiv \xi_{k+2} - \xi_k$ between the unfolded eigenvalues. According to [84] this distribution should fit to the distribution of nearest-neighbor spacing of the GSE. We demonstrate this correspondence in Figure 2.4. The solid line shows $P_{GSE}(s)$. Finally, the

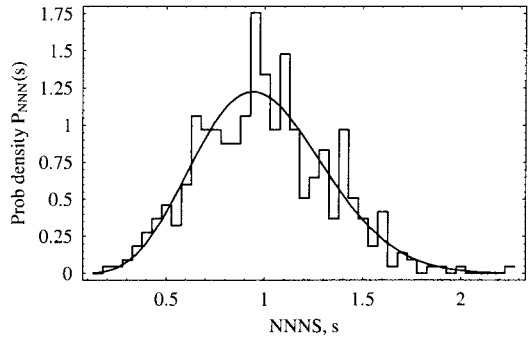

Figure 6. Next-nearest-neighbor eigenvalue spacing distribution $P_{nnn}(s')$.

long-range two-point eigenvalue correlations were tested. It is known [69, 51, 68], that if eigenvalues are uncorrelated we expect the number variance to scale linearly with l, ($\Sigma^2 \sim l$). Meanwhile, when the unfolded eigenvalues of C are correlated, Σ^2 approaches constant value, revealing "spectral rigidity" property of an RMT spectrum [69, 51, 68]. In Figure 2.5, we contrasted Poissonian number variance with the one we observed, and came to the conclusion that eigenvalues belonging to the "bulk" clearly exhibit

universal RMT properties. The broadening parameter $a = 8$ was used in Gaussian broadening procedure to unfold the eigenvalues λ_i [82, 83, 65, 81]. The dashed line corresponds to the case of uncorrelated eigenvalues. These findings show that the

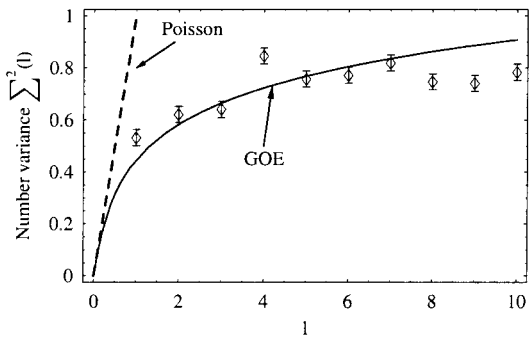

Figure 7. Number variance $\Sigma^2(l)$ calculated from the unfolded eigenvalues ξ_i of C.

system of inter-VLAN traffic has a *universal* part of eigenvalues spectral correlations, shared by broad class of systems, including chaotic and disordered systems, nuclei, atoms and molecules. It can be further concluded, that the bulk eigenvalue statistics of the inter-VLAN traffic cross-correlation matrix C are consistent with those of real symmetric random matrix R, given by Eq. (40) [80]. Meantime, the deviations from the RMT contain the information about the system-specific correlations. The next Section is entirely devoted to the analysis of the eigenvalues and eigenvectors deviating from the RMT, which signifies the meaningful inter-VLAN traffic interactions.

F The eigenstatistics as visual analytics

In this section, we demonstrate the use of the RMT based statistics and general eigenstatistics as visual analytics in congestion control, network monitoring and traffic anomaly detection. First, to establish the visualization techniques we will give the formulation of the statistics, their interpretation and visual examples. Then, we conduct the experiments on the real traffic time series to demonstrate the usefulness of these statistics for visual analysis.

1 Inverse participation ratio

We turn our attention to eigenvectors of inter-VLAN traffic cross-correlation matrix C, determined by $Cu^k = \lambda_k u^k$, where λ_k is k-th eigenvalue. Particularly important characteristics of eigenvectors is its inverse participation ratio (IP) (see, for example, Ref. [69]). The predictions are that all components participate in the eigenvectors of random interactions, while the number of significant contributors in eigenvectors of meaningful interactions is few. The IPR quantifies the reciprocal of the number of significant components of the eigenvector. For the eigenvector u^k it is defined as

$$I^k \equiv \sum_{l=1}^{N} \left[u_l^k\right]^4, \tag{52}$$

where u_l^k, $l = 1, \ldots, 497$ are components of the eigenvector u^k.

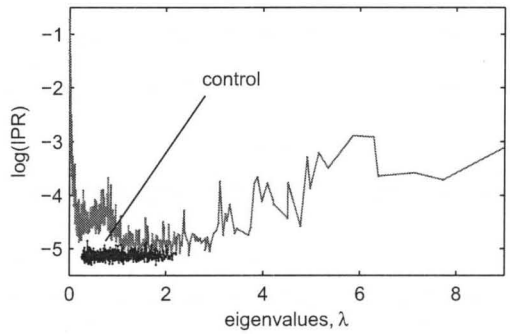

Figure 8. Inverse participation ratio as a function of eigenvalue λ.

The IPR is quite indicative in terms of signaling the number of significant u_k^l, i.e. "contributors" to the eigenvector of interest. For example, if we have reasons to expect absence of correlations between routers input into the experimental data, $I_k(0)$ should have its value around $1/\sqrt{N}$. Indeed, the eigenvector is normalized, thus $\sum_{l=1}^{N} \left[u_k^l\right]^2 = 1$. It has N components, and they are all roughly the same in magnitude (otherwise correlations must be present). Therefore, $u_k^l \simeq 1/\sqrt{N}$, and $I_k(0) \simeq 1/N$. Note, that since N is typically much greater than 1, any finite value of IPR signals *localization* (decrease in the number of eigenvector contributors) in inter-VLAN traffic.

In Figure 2.6 we plot the IPR of eigenvectors of cross-correlation matrix C as a function of spectral variable λ. The control plot is the IPR of eigenvectors of random cross-correlation matrix R of Eq. 40. As we can see, the eigenvectors corresponding to

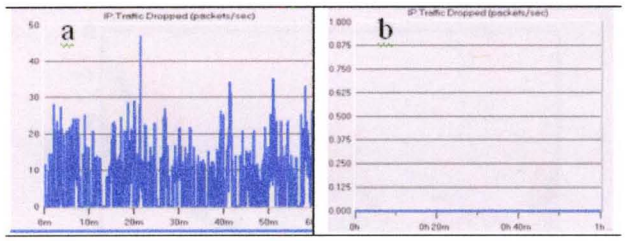

Figure 9. Dropped packets per second, (a) congested traffic and (b) uncongested traffic.

eigenvalues falling between 0.25 to 3.5, (which is within the RMT boundaries), have IPR close to 0. This means that almost all components of eigenvectors in the bulk interact in a random fashion (number of components, $1/0 \approx \infty$).

Another observation which we derive from Figure 2.6 is that the number of significant participants is considerably smaller at both edges of the eigenvalue spectrum. In other words, the IPR of eigenvectors, which signify the traffic collective event is high, meaning that there are few eigenvector contributors. Thus, the IPR plot can be used as visual tool to monitor the number of nodes involved into the collective traffic event.

The more illustrative example of IPR as a visual analytics for congestion control is presented further.

With the help of OPNET modeler simulation tool, we simulated the network layout with the same number of backbone routers and subnets. We have placed the nodes with high traffic loads in the simulated layout and insured the loss of utilities with the performance statistics provided by OPNET. The congestion of the traffic is defined as the loss of utility to a network user due to high traffic loads [85].The packet loss ratio of simulated congested and uncongested traffic are presented in Figure 2.7a and 2.7b, respectively.

The IPR of cross-correlation matrix C versus the position of eigenvalue λ in spectrum for simulated congested, simulated uncongested, real traffic and control (random matrix) are presented in Figure 2.8. The control green line is the IPR of eigenvectors of random cross-correlation matrix R of Eq. 40. Blue line is IPR of real traffic. Yellow and red lines are IPR of simulated uncongested and congested traffic correspondingly. As we can see, eigenvectors of real and simulated uncongested traffic (blue and yellow lines) are closer to the control IPR. The IPR of congested traffic (red

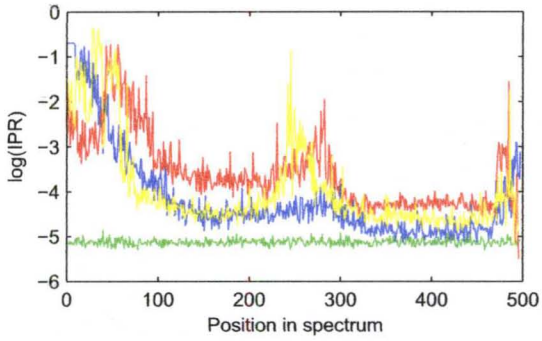

Figure 10. Inverse participation ratio as a function of eigenvalue λ.

line) shows the higher *localization* level. The localization signifies the restrictions in communication or correlation pattern formation. Even though there are still islands of freely communicating network nodes, the number of nodes involved in such communication decreases. The system attempts to keep the balance by dropping the packets, which is testified by packets loss ratio measurement.

2 Stability of inter-VLAN traffic interactions in time - overlap matrix

We assume that the health of inter-VLAN traffic is expressed by stability of its interactions in time. Meanwhile, the temporal critical events or anomalies will cause the temporal instabilities. The "deviating" eigenvalues and eigenvectors provide us with stable in time snapshots of interactions representative of the entire network. Therefore, these eigenvectors judged on the basis of their IPR can serve as monitoring parameters of the system stability.

We expect to observe the stability of inter-VLAN traffic interactions in the period of time used to compute traffic cross-correlation matrix C. The eigenvalues distribution at different time periods provides the information about the system stabilization, i.e. about the time after which the fluctuations of eigenvalues are not significant. Time periods of 1 hour, 3 hours and 6 hours are not sufficient to gain the knowledge about the system, which is demonstrated in Figure 2.9a. After 1hour the system-specific eigenvalues are very high and sketchy and differ from eigenvalues after 3 hours period and after 6 hours period. In Figure 2.9b the system stabilizes after 1 day period. To observe the time stability of inter-VLAN meaningful interactions we

compute the "overlap matrix" of the deviating eigenvectors for the time period t and deviating eigenvectors for the time period $t+\tau$, where
$t = 60h, \tau = \{0h, 3h, 12h, 24h, 36h, 48h\}$.

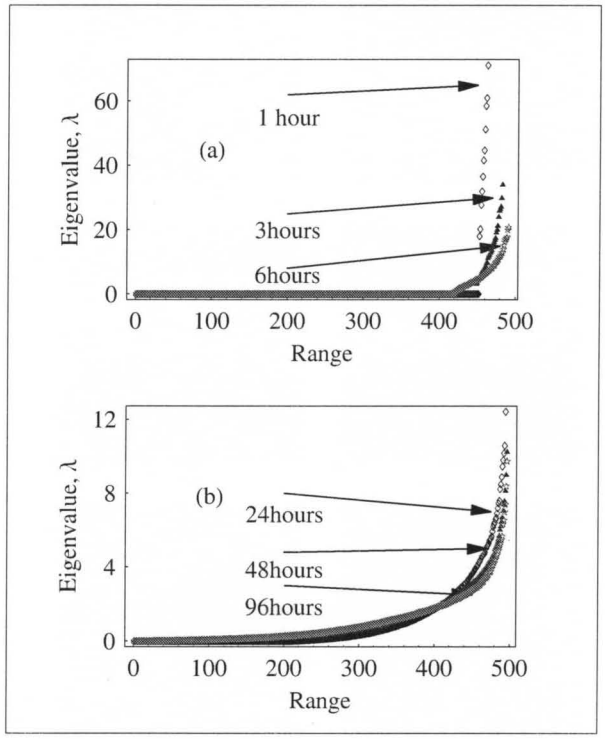

Figure 11. (a) Eigenvalues distributions of traffic streams correlation matrix C for 1 hour, 3 hours and 6 hours time intervals. (b) Eigenvalues distributions for 24 hours, 48 hours and 72 hours

For "overlap matrix", first, we obtain matrix D from $p = 57$ eigenvectors, which correspond to p eigenvalues outside of the RMT upper bound λ_+. Then we compute "overlap matrix" $O(t, \tau)$ from $D_A D_B^T$, where O_{ij} is a scalar product of the eigenvector u^i of period A (starting at time $t = t$) with u^j of period B at the time $t = t + \tau$,

$$O_{ij}(t, \tau) \equiv \sum_{k=1}^{N} D_{ik}(t) D_{ik}(t + \tau) \tag{53}$$

The values of $O_{ij}(t, \tau)$ elements at $i = j$, i.e. of diagonal elements of matrix O will be 1, if the matrix $D(t + \tau)$ is identical to the matrix $D(t)$. Clearly, the diagonal of the "overlap matrix" O can serve as an indicator of time stability of p eigenvectors outside of the RMT upper bound λ_+. The gray scale color-map of the "overlap matrices" $O(t = 60h, \tau = \{0h, 3h, 12h, 24h, 36h, 48h\})$ is presented in Figure 2.10. Black color of

gray scale represents $O_{ij} = 1$, white color represents $O_{ij} = 0$. At lag $\tau = 3$ hours the

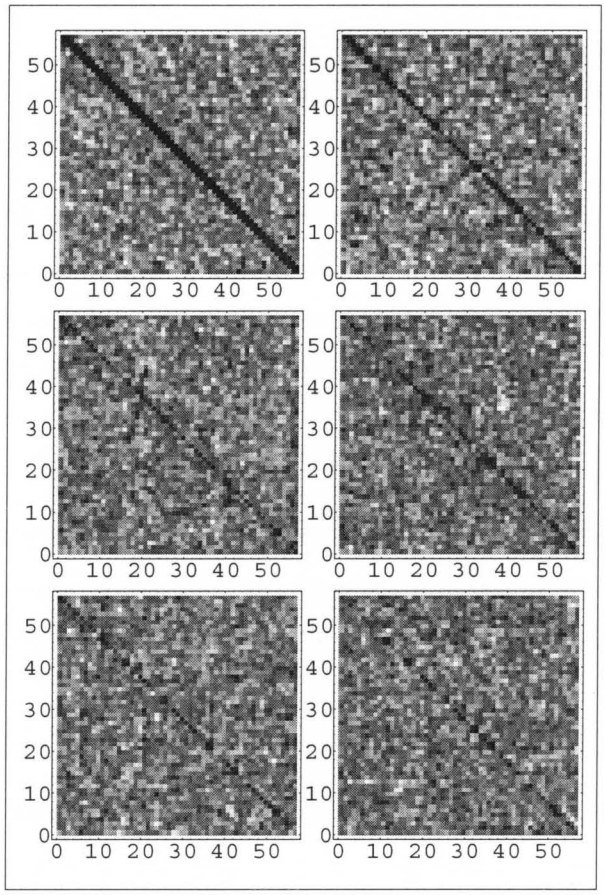

Figure 12. The gray scale of matrix $O(t, \tau)$ at $t = 60h$ and τ =0, 3, 12, 24, 36, 48h.

inter-VLAN interactions show the highest degree of stability. For further lags the overall stability decays. As the analysis of deviating eigenvectors content showed, the highly interacting traffic time series are time series of service based VLANs, intended for routing. In fact, we found three types of connections groupings. One group contains connections, which are interlinked on the router. We recognize them as, VLAN_X-router incoming traffic connection, VLAN_X-router_firewall connection and VLAN_X-router outgoing traffic connection. The connections, which are listed as VLAN_Y-router1, VLAN_Y-router2, VLAN_Y-router3, etc..., are reserved for the same service on every router and comprise another group. Final group of VLAN-router connections constituted of connections, which interact due to the routing. Particular network services are evoked at the same time and active for the same period of time,

which explains the stability and consequent decay of deviating eigenvectors of traffic interactions.

3 Meshgrid of eigenvector components and spatial-temporal representation of traffic load

At any time the readings from the network nodes give an instantaneous traffic load pattern. This pattern can be viewed as an expansion in terms of eigenvectors of matrix C in the following sense. An eigenvector u_k is a set of different intensities of network-wide traffic load satisfying

$$Cu_k = \lambda_k u_k.$$

Among possible configurations of network-wide traffic load u_k^i is an amount of traffic load on a particular node. Then, ratio u_k^i/G_k is equal to the number of nodes involved in the mutual interaction. For a variance of a traffic load at a given node we get:

$$\sigma_k^2 = \left\langle \left(\sum_{i=1}^{M} \frac{u_k^i}{G_i} \delta G_i \right)^2 \right\rangle = \sum_{i,j=1}^{M} u_k^i u_k^j C_{ij} = u_k^T C u_k. \tag{54}$$

At this point we can employ the result of Eq. (31) to realize, that the variance of the traffic load at a given node is specified by the corresponding eigenvalue: $\sigma_k^2 = \lambda_k$. Once again, this is true for a network-wide traffic described by the u_k. By contrast, there is no correlation between two network-wide traffic loads attributed to two eigenvectors u_k and u_l:

$$\left\langle \left(\sum_{i=1}^{M} \frac{u_k^i}{G_i} \delta G_i \right) \left(\sum_{j=1}^{M} \frac{u_l^j}{G_j} \delta G_j \right) \right\rangle = u_k^T C u_l = 0,\ b \neq l.$$

With this in mind, if we mesh-grid the eigenvector components against time we will obtain the dynamics of particular network-wide traffic load in space, due to precise location of significant components, and time. The mesh-grid of last eigenvector u^{497} components for time period $t + \tau$, where $t = 36$ hours and $\tau = 6n$, $n \in \{0, 1, \ldots, 7\}$ is shown on Figure 2.12.

Most recent research on network traffic analysis focuses on observing temporal dynamics of traffic and effects from user and protocol behavior [86]. In such analysis, detailed Internet Protocol (IP) packet traces on individual links reveal the

characteristics of network traffic at multiple time scales, e.g., rich scaling dynamics arising over small time scales, and self-similarity and long-range dependence at large time scales [87]. Recently, graph wavelets have been proposed for spatial [88] traffic analysis with knowledge of aggregate traffic measurements over all links [89]. This method can provide a highly summarized view of traffic load throughout an entire network. Despite these advances, spatial and temporal traffic analysis still presents difficult challenges, not only because large-scale distributed networks exhibit high-dimensional traffic data, but also because current analytical methods require examination of large amounts of data, which can strain memory and computation resources in even the most advanced generation of desktop computers. Despite these inherent difficulties, investigation of spatial-temporal dynamics in large-scale networks is an important problem because modern society grows increasingly reliant on the Internet, a network of global reach that supports many services and clients. Lacking means to predict, monitor, and adjust spatial-temporal dynamics, Internet Service Providers (ISPs) typically overprovision network capacity, which leads to under-utilized resources on average with overloaded hot-spots arising from time to time. Further, the Internet appears increasingly vulnerable to attacks and failures [90, 91]. These factors suggest a crucial requirement to devise and develop promising tools that can monitor network traffic in space and time to identify shifting traffic patterns. Such tools can aid in operating and engineering large-scale networks, such as the Internet. While useful network management tools might focus on either offline or online monitoring and analysis, the task of network-wide on-line monitoring presents more stringent requirements for transferring and handling traffic data in a timely fashion.

4 Network topological representation of the traffic load

Another visualization example is inspired by popular among network practitioners technique - network topological representation as a graph. The network-wide traffic load, expressed by the components of the eigenvector of interest, in our case eigenvectors outside of the RMT boundaries, can be visualized as an indirect graph with active and inactive edges. Active edge corresponds to the traffic time series, which is a significant participant in a given eigenvector (traffic load). The

illustration of this technique is presented in Figures 11-15.

5 Experiments with traffic data set to detect anomalies of traffic interactions

Among the essential anomalous events of VLAN infrastructure we can list violations in VLAN membership assignment, in address resolution protocol, in VLAN trunking protocol, router misconfiguration. The violation of membership assignment and router misconfiguration will cause the changes in the picture of random and non-random interactions of inter-VLAN traffic. To shed more light on the possibilities of anomaly detection we conducted the experiments to establish spatial-temporal traces of instabilities caused by artificial and temporal increase of the correlation in normal non-congested inter-VLAN traffic. We explored the possibility to distinguish different types of increased temporal correlations. Finally, we observed the consequences of breaking the interactions between time series, by injecting traffic counts obtained from sample of random distribution.

Experiment 1

We selected the traffic counts time series representing the components of the eigenvector which lies within the RMT bounds and temporarily increased the correlation between these series for three hour period. The proposed monitoring parameters show the dependence of system stability on the number of temporarily correlated time series (see Figure 2.11). Presented in Figure 2.11.a, left to right are eigenvalue distribution, IPR of eigenvectors and the overlap matrix of deviating eigenvectors. The same parameters with induced temporal correlation between ten and twenty time series are shown on Figures 2.11.b and 2.11.c correspondingly. One can conclude that increased temporal correlation between ten time series does not affect system stability. Meanwhile, when the number of temporarily correlated time series reaches the number of significant participants of eigenvector of largest u^{497} of largest eigenvalues ($\tilde{}$ 22), the system becomes visibly unstable. The largest eigenvalue changes from 10 to 12, the tail of inverse participation ratio plot is extended and the diagonal of "overlap matrix" disappears

In addition, we visualize in Figure 2.12 the system instability during temporal

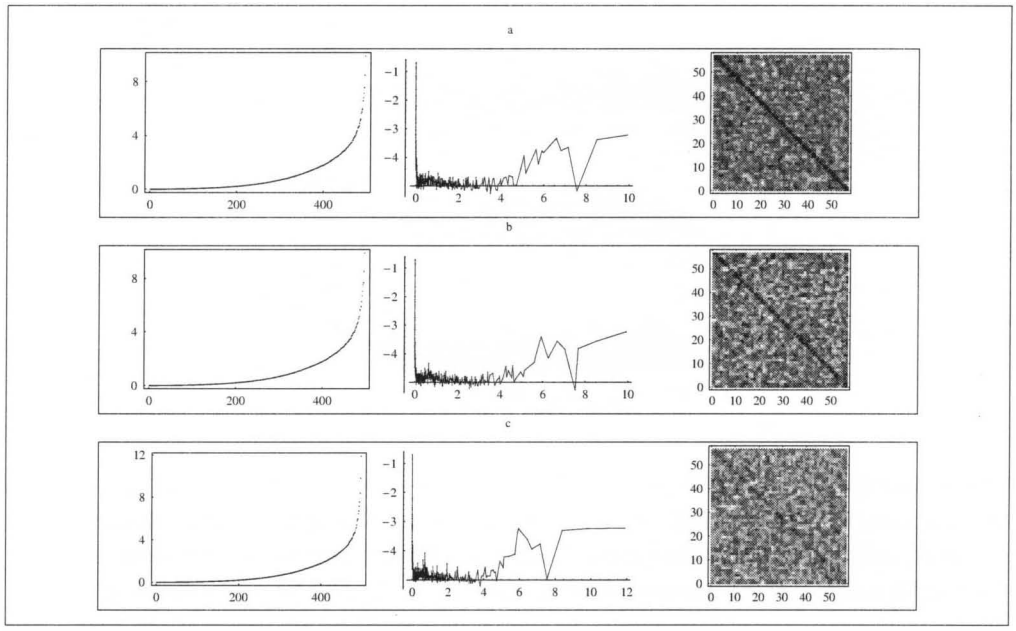

Figure 13. Eigenvalues distribution, IPR and overlap matrix of deviating eigenvectors.

increase of correlation between twenty time series with spatial-temporal representation of eigenvector u^{497}. In Figure 2.12a the spatial-temporal pattern of u^{497} captures precise locations of system-specific interactions of uninterrupted traffic for 84 hours of observation. The abrupt change of this pattern in Figure 2.12b indicates the starting point of induced correlation between twenty traffic time series usually interacting in a random fashion. As we can see the stable pattern of eigenvector u^{497} moves to eigenvector u^{496}, the weights and locations of significant components of eigenvector u^{496} are suppressed and replaced by the weights and locations of significant components of eigenvector u^{497} when the interruption ends. Thus, we are able to observe the end point of the induced correlations in Figure 2.12c, which represents weights of components of eigenvector u^{496} plotted with respect to the same time intervals. With this setup we are able to locate the anomaly in time and space.

Translated to network topological representation, the behavior of eigenvectors $u^{497}$and u^{496} during our manipulations with inter-VLAN traffic may be monitored with the following graphs (see Figure 2.13).

Experiment 2

In the previous experiment we injected just one type of increased correlation

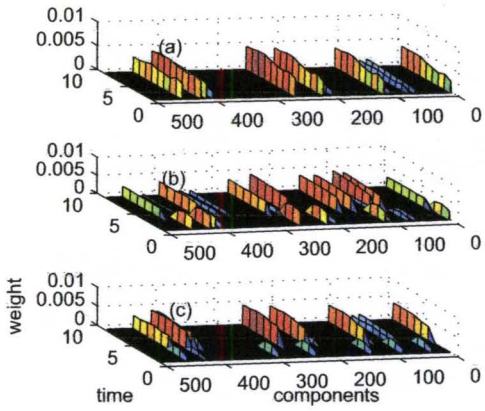

Figure 14. (a) The weights of components of u^{497} plotted for time period from 36 to 84 hours of uninterrupted traffic with 6 hours interval. (b) The weights of components of u^{497} plotted with respect to the same time period, with induced three hours correlation. (c) The weights of components of u^{496} plotted with respect to the same time period, with induced three hours correlation.

among time series. Now we make two and three different types of induced correlations produce different spatial-temporal patterns on eigenvector u^{497} components (see Figure 2.14). Time series for temporal increase of correlation are obtained in the same way as in Experiment 1. We temporarily increased the correlation between series by inducing elements from distributions of sine function and quadratic function, respectively for three hours. In Figure 2.14a, one type of three hours correlation is induced among ten traffic time series and another type of correlation among other ten time series. Three different types of three hours correlations are induced among twenty traffic time series in Figure 2.14b. The sorted in decreasing order content of significant components shows that time series tend to group according to the type of correlation they are involved in.

Experiment 3

Next we turn our attention to disruption of normal picture of inter-VLAN traffic interactions. This can be done by injecting the traffic from random distribution to non-randomly interacting time series for three hours. We demonstrate it by examining the eigenvalue distribution, the IPR and the deviating eigenvectors overlap matrix plotted in Figure 2.15. After 60 hours of uninterrupted traffic, we injected elements from random distribution to significant participants of u^{497} for three hours. The largest eigenvalue increases, from 10 to 12. Extended IPR tail shows the larger number of

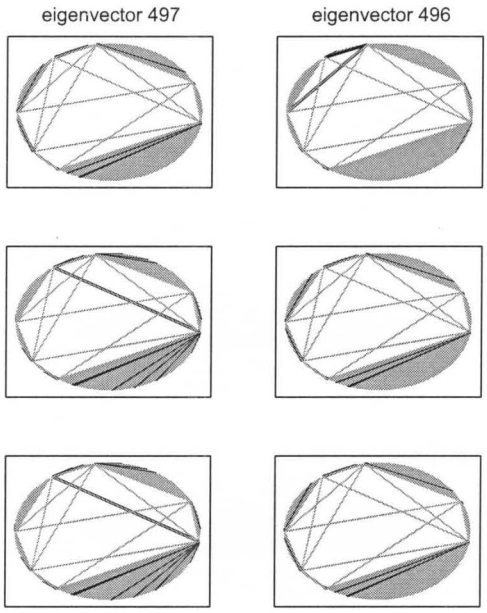

Figure 15. Left column - behavior of u^{497} during time period from 48h to 60h with 6h time window, induced correlation starts at 54h and lasts for 3h. Right column - behavior of u^{496} in same conditions.

localized eigenvectors and we observe the dramatic break in deviating eigenvectors stability.

Very often, however, one has to come up with more specific information about location of anomaly in the network. Such a detection is highly challenging task given stochastic environment. One of the traditional approaches in discerning the intrusion or malfunction in traffic from the normal flow of a network is PCA. More recently the non-linear extension of PCA, called Kernel PCA (KPCA), similar to the SVM, has gained increased popularity. The relevance index in both PCA and KPCA, the reconstruction error [[13]], which helps quantifying the anomalies, is the subject of the next Chapter.

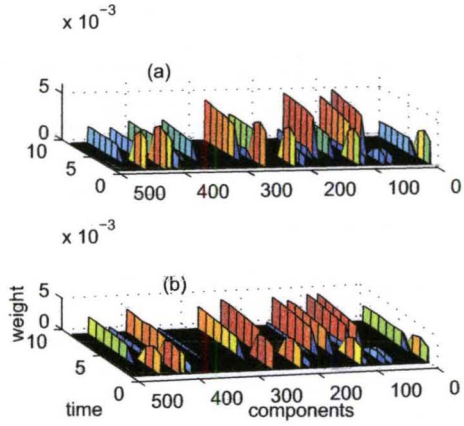

Figure 16. (a) The weights of components of u^{497} plotted for time period from 36 to 84 hours with 6 hours interval, two different types of induced correlations. (b) The weights of components of u^{497} plotted with respect to the same time period, three different types of induced correlations.

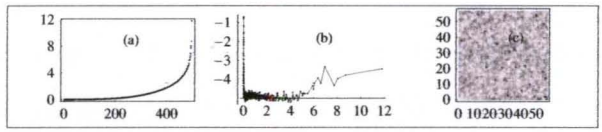

Figure 17. Eigenvalues distribution, IPR and overlap matrix of deviating eigenvectors of inter-VLAN traffic cross-correlation matrix C.

CHAPTER V

CLASSIFICATION WITH FEATURES EXTRACTED BY THE RMT

In this chapter we explain the relationship between linear method for features reduction - PCA, non-linear method - KPCA and our algorithm based on the RMT. We make our explanations more illustrative with classification task of traffic anomaly detection.

Provided that, the underlying assumption about features (dimensions) interaction in the data set is linear, the most efficient way of uncovering structure of the data set, is to explore spectral properties of its correlation matrix. PCA turns the data set with a number of linearly related features into a smaller number of uncorrelated features called principal components or eigenvectors of corresponding eigenvalues of the spectra[13]. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible. Thus, effectively number of new features of the data set is reduced to the amount of variability researcher would like to preserve in the "new" data set. In new dataset there is no direct connection between original variables and new variables. In particular, every new variable is a weighted sum of original variables.

The wide popularity of PCA is backed up by frequent success in many engineering and data mining problems. It is really hard to cite all studies and works where PCA is successfully used. Enough to say that it is a regular component in any standard data mining or multivariate statistical package. The number of failures are plenty as well. Even though the method operates on linearity assumption, in most cases data is non-linearly polished [4, 5] to provide the starting point for PCA usage. Polishing includes outliers removal, and other procedures which create bias in data and change the natural distribution of data set variables.

assumption about linear relationship between data set variables. Similar to Support Vector Machine, KPCA is using a kernel function to map the data into a different space with higher dimensionality. In this space, KPCA extracts the principal components of the data distribution.

For the case of two classes in the data set, SVM as tool to find classes separating hyperplane in a high dimensional space is proven to be very successful [41, 42]. However, when one of the classes is almost not-existent, i.e. the class is a rare event, the problem is considered to be a one-class classification or novelty/anomaly detection. The classifier learns only one class; the learning is actually a process of reconstructing data set from principal components of mapped high-dimensional kernel. Reconstruction error in a new feature space is then used as a novelty or anomaly measure. Reconstruction error computed with the help of KPCA as a measure of novelty showed a promising result in early diagnosis of cancer [92].

It is not very hard to consider detection of unusual traffic loads and time series patterns of network traffic in one-class classification framework. There would be some desired properties of the mechanism underlying the reconstruction error computation. We would like to make no assumptions about the nature of relationships between the features of the traffic data set as well as to skip the non-linear polishing of traffic data. Also, we want to be able to explain the contribution of original features to anomalies. In case of KPCA, even with high reconstruction error which will signify the high separability of background and anomalous data, we would not be able to reveal the culprit for network practitioners. The latter have to know what node, or router or server of the network has to be monitored closely.

We propose an algorithm where reconstruction error is computed based on spectral intelligence obtained with the help of the RMT.

A Motivation for reconstruction error

Reconstruction error in new feature space used in KPCA is geometrically motivated and aims at giving lower classification errors than the one-class SVM [92]. The reconstruction error recovers circular shapes, including numeral eight, struggles with images, like Olympic rings, and has limited success with general longhand. Yet,

the overall success with recognition of "novelties" is convincing, and these robustness with respect to different noise levels makes construction error into an appropriate tool for our purposes.

Algorithmically this recognition process is broken down in three parts. In first, data is split into train subset, which contains only one class and test set with novel class. Training part is mapped to non-linear kernel. In second, one reconstructs the manifold of test data from spectral decomposition of train data. And finally, the reconstruction error is computed.

The most popular example of the latter is based on the concept of orthogonal vectors having zero projection onto each other. For two n-dimensional data vectors, for which we assume inner product $(X \cdot Y)$ to have usual Cartesian form, $(X \cdot Y)\, X$ is, up to a constant, projection of Y on X, and $(Y \cdot X)\, Y$ is, up to a constant, projection of X on Y. Hence, reconstruction error could be defined, via

$$p = X \cdot X - VX \cdot VX, \tag{55}$$

where V is a matrix of eigenvector of correlation matrix $C = X^T X$. In kernel methods, the same is true for vectors in feature space, and kernel build upon them.

Below we analyzed a specific problem of feature selection in the case of forceful modifications of time series data. We considered intrusions into the traffic, for which the above described difficulty is quite real, but there is a possibility of finding the way out. The hope was, that upon comparison of features extracted from the time series, before and after the insertion of foreign segment, there would be a unique relevance index, quantifying the impact associated with each insertion. And indeed we were able to find such a relevance index. The idea behind it is similar to the idea behind the so called reconstruction error used in non-linear spectral methods.

B Reconstruction error used in PCA and RMT based algorithm

In our analysis we used similar construct. Except, first, we had done feature selection in accordance with two deferent methodologies. Specifically, for PCA, we restricted set of features used in subsequent analysis to eigenvectors contributing to 90% of net variance (trace of correlation matrix). For our algorithm, we look at the

spectrum from a different point of view. The eigenvalues are split into three groups, according their role in the dynamics of the network: bulk eigenvalues (those within RMT bounds), left deviating eigenvalues, and right deviating eigenvalues. Either of the three could be a good group of features for the purposes of finding the "novelty", i.e. intrusion.

Hence we defined an eigen-kernel $W_{tr} = X_{tr}\hat{V}$, with X_{tr}, being our train time series data - a chunk of the original data, selected before any intrusion, and $\hat{V}$ is a set of principal components in case of the PCA and bulk eigenvectors in case of our algorithm. The group of bulk eigenvalues had been selected because of the regular, non-random nature of the intrusions. We expected salient features of random background to remain unaltered by the types of network disruptions we analyzed.

If train data of N time series is L_1-long, then, for l selected features, eigen-kernel W_{tr} is $L_1 \times l$ matrix of eigen-signals. Similarly we can construct two more eigen-kernels for X_{test} and X_{alt}, the test and altered traffic time series data. We have two $L_2 \times l$ matrices ($L_2 = L - L_1$) $W_{test} = X_{test}\hat{V}$ and $W_{alt} = X_{alt}\hat{V}$. For each node of the network we then define, an impact factors $f_{tr} = W_{tr}^T X_{tr}$, $f_{test} = W_{test}^T X_{test}$, and $f_{alt} = W_{alt}^T X_{alt}$, all $l \times N$ matrices.

In order to reduce vectors representing each node in our linearly transformed data we define a reconstruction errors through inner products according to

$$\rho_i^{test} = \frac{1}{N^2}\left[\left(f_{test}^T\right)_i (f_{test})_i - \left(f_{train}^T\right)_i (f_{train})_i\right], \tag{56}$$

$$\rho_i^{alt} = \frac{1}{N^2}\left[\left(f_{test}^T\right)_i (f_{test})_i - \left(f_{alt}^T\right)_i (f_{alt})_i\right], \; i = 1, \ldots, N, \tag{57}$$

for test and altered traffic respectively. The normalization factor is somewhat arbitrary and is used for visualization convenience.

The impact factors can be interpreted as matrix elements of selected feature basis in data space, and, therefore, reconstruction errors help to identify those nodes that changed their decomposition, and hence, experienced intrusion. Once again, the only difference in PCA and RMT algorithms was the way matrix $\hat{V}$ had been constructed.

C Insertion experiments

The main difficulties in diagnosing the attack or disruption of service are related to speed of detection and precision in locating the origin of abnormal traffic. Since we take statistical approach, solution to both problems can be reduced to analyzing time series correlations, computed for a selected interval. The length of the interval, and most importantly, corresponding diagnostic accuracy are key factors in judging of a particular method.

We ignored topological differences, and assumed that any router of our network could be the target. The sources of disruption were chosen at random. The primary characteristics of the abnormality aside from rate of its change, were number of affected routers, affected time interval, and relative power. By varying these parameters we tested robustness and sensitivity of both methods, the one based on traditional PCA and the one involving the RMT.

The form of the insertions models several known attack strategies. In particular, we attempted to recognize abrupt and gradual rates of change of the disruptions. The latter had taken place simultaneously, at several locations, whose number, as we already mentioned, are part of the parameter space. We used three forms of insertions, and run our analysis on the resulting correlation matrices. For each insertion we fixed two parameters, while changing the third one.

The first type of increasing rate attack had rectangular shape in logarithmic difference scale of time series. The height of the rectangle describes the power, and the length measures the extent of the attack. This was an example of distributed disruption of service (DDOS) [93], which involved sudden change in temporal pattern of the traffic.

Then, we considered attacks with gradually increasing rate of change in traffic volume. In our simulation, several time series had their actual values removed and replaced with the segment, characterized by the linear increase on logarithmic scale. The shape of the disruption is, in general, a trapezoid, fully specified by its mean base - the power, and its height - the span.

And finally, we simulated a less trivial transient behavior of affected network nodes. The log-scaled insertion segments had a form of exponential. Despite

complicated functional dependence, $g_{insertion} \sim \exp\{\alpha t + \beta\}$, with α and β being randomly chosen real numbers, and t - integer increments of Δt, such a DDOS can again be defined in terms of two main parameters. We could still defined both Length of the attack and certain average power.

Our analysis of correlations in network with DDOS is fully tantamount to macroscopic monitoring of the traffic and the following real time algorithm. For a given time interval L, and number of nodes N, we slice traffic data into "train" and "test" subparts. The time-lag needed for averaging in calculation of matrix C, could be, for example, $L/2$. We then, compute reconstruction error, using the most appropriate method (determined below). Depending on the outcome, the train part of the data set can be either extended, kept the same or replaced with the test, provided that the latter is free from DDOS. Thus, in the next subsection, we focused on demonstrating the superiority of one of our chosen methods over the other.

As far as general spatial-temporal correlations approach to the attack diagnostics, is concerned, it had been developed over the years (see, for example,). The abnormalities we considered here are similar to the ones brought up in reference [93]. Apart from practical importance of these examples, this work also discusses technical expenses of methods targeting spatial-temporal correlations.

D Comparison of reconstruction errors computed with PCA and RMT

The aim of this subsection is to provide a clear support for the linear methods in the face of DDOS diagnostics [93]. In particular, we intended to show, that the RMT eases feature extraction, when the reconstruction error is used in novelty recognition.

Both methods we examined provide significant dimensional reduction in selection of the features. Except, in case of the PCA, we have a rather nonrestrictive variance criteria, while an algorithm involving the RMT only imposes certain boundaries on values L and N, but other than that, is free from the externally specified conditions.

Once the orthonormal eigen-basis is found for a given C, computed either for real traffic or a modified one, we used the RMT boundaries to select the eigenvectors corresponding to the bulk. These are then, used to construct eigen-kernels, impact

factors, and corresponding reconstruction errors. The procedure was similar for the PCA except, eigenvectors we used were selected based on the ninety percent contribution to variance criteria. After that, Eqs. (56) and (57) produced all the necessary information on abnormalities in test traffic.

The classification performance on traffic data was evaluated using ROC curves. A ROC curve plots the fraction of test patterns correctly classified as anomalous (true positives) versus the fraction of patterns incorrectly classified as anomalous (false positives) to illustrate the performance over all possible decision thresholds. To compute such a curve, first, the reconstruction error ρ_i was evaluated for all test patterns i. Second, the set $\{\rho_i\}$ was sorted according to the p-values. Finally, by counting how many novel and ordinary samples are above a decision threshold taken between two neighboring p-values, the fractions of false and true positives are readily available. Thus, for each ρ_i, there is a point on the ROC curve. Together, these points cover the full range of false positives: from 0 to 1.

In Figures 18.a and 18.b we presented ROC curves for several values of parameter p, the fraction of disturbance affected nodes. The insertion is a simple rectangle with the height of the same order as mean variance of the log-scaled time series. The length of the rectangle is about one tenth of the total length of the time series. For small values of p both methods show low level of success. The situation improves when the percentage of sites with altered traffic is higher than ten. The progress is considerably bigger for the RMT based method.

As we fix p and alter parameter l, the ratio of the length of the rectangular insertion the net length of time series, the ROC curves become even more eloquent. Standard PCA fails along the full range of the parameter. By contrast, the RMT feature selection works perfectly, as it can be seen in Figures 18.c and 18.d. Furthermore, the superiority of the RMT selection is transparent, when both p and l are unchanged, while power of the attack, governed by h, the ratio of the height of the rectangle to mean variance of the affected series. The ROC curves are presented in Figures 18.e and 18.f.

Next we had begun to introduce linearly varying DDOS. Once again we varied p first, keeping the rest of the parameters constant. Figures 19.a and 19.b illustrate the

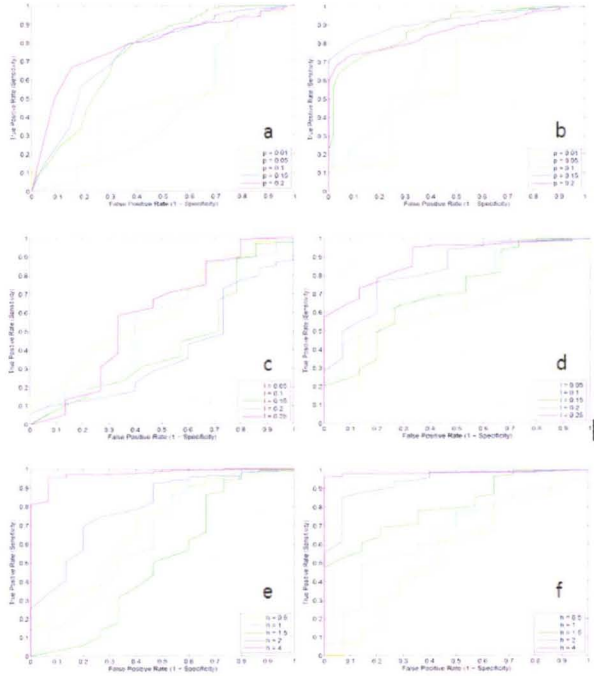

Figure 18. ROC curves of reconstruction error at constant rate attack. Left column ROCs of PCA, right column ROCs of RMT. ROC dependence from number of involved nodes is on row 1, from length of attack is on row 2, from intensity of attack is on row 3.

fact, that PCA detector is very inconsistent, inaccurate for most of the considered values. However, with the help of the RMT, the diagnostics procedure becomes almost flawless, once at least one tenth of the network nodes is compromised.

Increasing the span of the attack l creates even greater separation between two feature selection methods. In fact, as we see from comparison of Figures 19.c and 19.d, the PCA uniformly fails throughout the range of l. Meantime, its rival, does remarkably well. Then, we repeated the procedures for the case of increasing parameter h, characterizing the DDOS power. This time, the PCA approach to construction of feature based kernel is consistently off (Fig. 19.e). The RMT-governed diagnostics, on the other hand, works perfectly, even for relatively small linearly grown, in log-scale, disturbances (Fig. 19.f).

Finally, we turned to the injections with more rapid rates of traffic growth, i.e. the disturbances characterized by the exponential time dependence on a log-scale we used throughout this work. In Figures 20.a and 20.b we displayed the outcome for

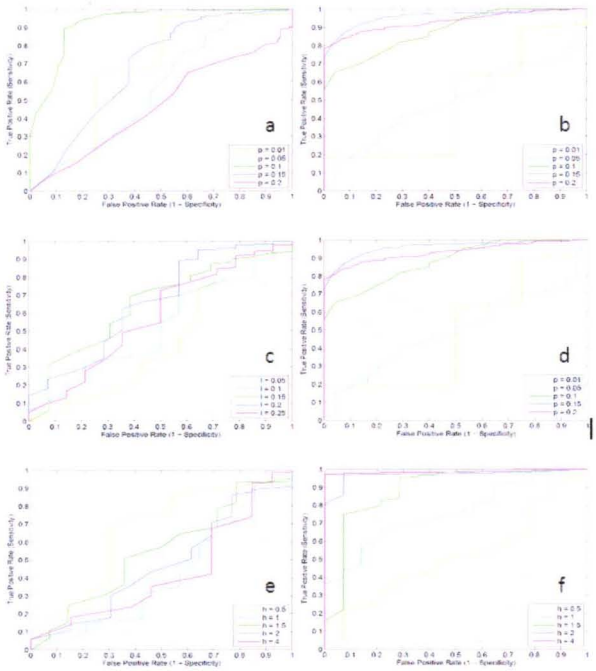

Figure 19. ROC curves of reconstruction error at linear rate attack. Left column ROCs of PCA, right column ROCs of RMT. ROC dependence from number of involved nodes is on row 1, from length of attack is on row 2, from intensity of attack is on row 3.

different values of participation p. The PCA backed approach exhibited occasional success together with simultaneous bogus results. Our second method had also struggled a bit with large number of affected routers. Yet, it had been quite dependable across the entire range of p.

As far as the experiments, with the span of the injections l, are concerned, the picture is largely the same. Our method struggled with extremely short DDOS, but had been foolproof in the rest of the instances, as can be witnessed in Figure 20.d. But its PCA counterpart in Figure 20.c, demonstrated consistent fallacy.

And, at last, we varied the power governing parameter h. The results are close to the ideal for both methods. However, the RMT procedure has a slight, but unquestionable edge (cf. Figures 20.e and 20.f).

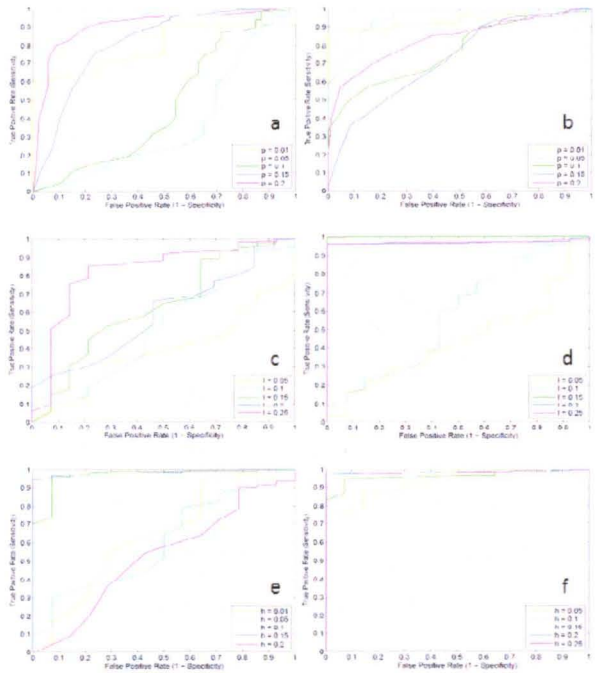

Figure 20. ROC curves of reconstruction error at exponential rate attack. Left column ROCs of PCA, right column ROCs of RMT. ROC dependence from number of involved nodes is on row 1, from length of attack is on row 2, from intensity of attack is on row 3.

E Conclusion and recommendations

We proposed a new algorithm for anomaly or time series pattern detection in network traffic. The detection power based on the reconstruction error of test data set. Test data set is reconstructed from partial spectral space of train set, where part of the spectrum is defined by RMT boundaries.

The benefits of our approach over PCA are the following: we do not make assumptions about the nature of relationships between the features of the traffic data set in our case features are nodes of the network. We avoid non-linear polishing of the data, in particular we do not perform any transformation of the variables except for logarithmic, correspondingly we are not concerned with outlier detection. The reconstruction error is susceptible to the value of variance allowed in the dataset, thus in case of PCA variance becomes a parameter of reconstruction error. RMT, on the other hand, gives precise solution for defining the boundaries of the spectrum, which is going to be reconstructed.

With high separation power of KPCA reconstruction error comes the high computational price. Mapping large dataset to high dimensional kernel, and storing the kernel are computationally intensive. Our method is no more expensive than simple PCA. Another advantage is that we are able to interpret the detected novelty or anomaly by mean of original features, in our case nodes of the network. For network specialists, the kernel feature space is not very helpful environment to monitor the network.

Based on the results of our simulations, we would like to pin point several observations: larger the proportion of nodes involved into the anomalous activity, longer the intrusion, higher the intensity of intrusion, easier its detection. Our algorithm allows to detect the constant rate intrusion when 15% of the network is involved, linear rate intrusion when as low as 10% of the network is involved and exponential rate intrusion when only 5% of the network is affected. At constant and linear rate intrusion our algorithm needs from 20% to 25% of time window, at exponential rate intrusion it detects the attack as early as 10% of the time of the attack. The intensity of the traffic during the attack at constant and linear rate has to be two times higher than intensity of normal traffic. At exponential rate of the attack algorithm is sensitive even to the half of the intensity of normal traffic.

CHAPTER VI

CLUSTERING WITH FEATURES EXTRACTED BY THE RMT

It turns out, the possibility of improving traditional methods with the insights given by the RMT, goes beyond the framework of time series analysis. In this Chapter we took on another data set, of completely different nature, which is equally important in applications. We demonstrate, that hierarchical clustering algorithm, aided by the RMT, has a strong potential to broaden our scope from the discovery of network disruption, to the discovery of cancer.

Measure of similarity is most important step in any type of clustering, since definition of the cluster is subset of similar data points. We propose a new algorithm, which uses techniques of the RMT to identify and remove noise from similarity/distance matrix. After noise undressing the distances, we use an average linkage hierarchical clustering. New algorithm is tested on benchmark dataset of two leukemia types introduced by Golub *et all* in 1999 [94]. Obtained clusters are validated externally employing apriori knowledge of classes and internally, evaluating compactness of individual cluster and disparity between different clusters. Proposed algorithm clearly produces two clusters for two types of leukemia, moreover it identifies T and B cells subclusters of one type of leukemia. Our algorithm is scalable due to RMT assumptions of infinite length of data points. Compared to other clustering techniques for high dimensional data, it allows us to avoid the dimensionality reduction problem.

A Classification task and similarity measure

Clustering is an important data exploratory tool. It is defined to recognize subsets of data which are similar in a certain way. Besides revealing the natural

ground for supervised learning. Thus, it can be exploited in classification task.

Clustering techniques vary by the type of dataset partitioning and assembling the clusters. In hierarchical algorithms successive clusters are assembled from previously established clusters in a bottom-up or top-down fashion. Each data point in bottom-up algorithm is a separate cluster, which can be merged with other clusters into a higher level larger cluster. Divisive or top-down clustering considers the whole dataset and divides it into successive clusters. Partitional algorithms typically determine all clusters at once, which implies apriori knowledge of number of possible clusters in the dataset. Density-based and self-organizing map types of clustering are defined to determine dense regions in the dataset and form arbitrary-shaped clusters.

Regardless of the clustering methodology there is a general criteria of good clusters, that is the inter-cluster relationship between data points have to be stronger than intra-cluster relationship, i.e. points in one cluster have to be more similar than points in two different clusters. Hence, the most essential step in any clustering is to establish the clear relationship between data points and to transform this relationship into appropriate distance or similarity measure.

The main challenge in clustering the real life applications such as image recognition, robotics, gene micro arrays, document categorization, networks dynamics, is dimensionality of the data [99]. There are two general approaches to the problem, reduction of data dimensionality or data subspace search [100, 101]. Graph partitioning, manifold learning, Support Vector Machine are few among various methods of subspace searching, which are usually computationally involved. Popular approach to dimensionality reduction is spectral decomposition of data matrix and PCA [102]. Latter is computationally less demanding, number of selected components is based on certain variance threshold accepted by community without rigorous statistical confirmation. In PCA, there is no one-to-one correspondence between reduced and original variables, since new variable or principal component is a linear combination of all old variables.

In this Chapter, we propose the clustering algorithm which assumes infinite dimensionality of the data. In fact higher dimensionality of the data matrix will improve the clustering results. Pairwise relationships between data points are usually

presented by a correlation matrix. We employ the RMT techniques to identify and remove noise from correlation matrix. Once correlation matrix is noise undressed it is transformed into the distance/similarity matrix. Consequently, distances are fed into the hierarchical clustering algorithm. The choice of clustering algorithm after RMT steps is dictated by the application. We apply our algorithm to gene expression data, for which the hierarchical clustering is proven to be a successful method. Results of this type of clustering are naturally explained, since data points in bio-domain are very often hierarchically related.

B Clustering in different contexts

One of commonly used measurement of linear relationship strength between two variables is Pearson correlation coefficient. For multiple variables, pairwise variables relationships are represented by symmetric correlation matrix. Similarly, correlation matrix may be constructed from pairwise relationships between data points. In different applications different relationships are considered either between variables or between data points. Sometimes, both types of relationships are used to distinguish substructures or meaningful groups and clusters. Financial applications, for instance, are mostly focused on correlations between stock returns, since successful portfolio diversification benefits from identification of similar stocks. Biological applications, particularly gene expression analysis, make use of two-way clustering, where, correlations between different data points are examined after the correlations between different genes. Latter may reveal functional genes clustering and sometimes substantially reduces the number of variables in data points. Unfortunately, the correspondence between genes and new variables becomes non trivial in this case.

The further analysis of variables or data points relationship is sensitive to the presence of random noise in correlation matrix. With noise undressed correlation matrix of financial time series Giada and Marsili uncovered the clusters of companies belonging to the same economic sectors [103]. Series of studies on filtering eigenvalue spectra of correlation matrix of stock returns [65] revealed their clustering structure and demonstrated time stability of these clusters. In 2005, Kim and Jeong proposed systematic decomposition of correlation matrix into random and non-random parts

[86]. They showed that unfiltered correlation matrices fail to identify groups of related stocks. Removing the noise from correlation matrix of genes, Luo *et all* [104], constructed gene-co-expression networks and proposed predicting function for unknown genes.

Motivated by these results and our own studies on noise extraction from correlation matrices of complex traffic networks [109], we develop clustering algorithm which uses techniques of noise identification with the help of RMT.

C RMT based hierarchical clustering

Given the data matrix A of size $N \times L$, where N is the number of data points and L is the number of variables, our algorithm starts with computing the correlation matrix C of size $N \times N$. Next, singular value decomposition of C is performed, thus eigenvalues λ and eigenvectors V are obtained. Maximum eigenvalue is an overall variance of linear combination of N components. In financial data it corresponds to so called global market variance, while in gene expression data it represents the variance of dominating organism/class or family of related genes. Algorithm removes the influence of largest eigenvalue by finding projection $\hat{X}_{max} = XV_{max}$ and linear fit of every data point to this projection. The difference between linear fit and X is a new dataset $X_{residual}$ for which new correlation matrix $C_{residual}$ and corresponding eigenvalues λ and eigenvectors V are calculated. Algorithm proceeds with RMT tests if size of data matrix satisfies the following requirements: $\frac{L}{N} = Q > 1$, since boundaries of eigenvalue spectra where the RMT tests are applicable lie within $[\lambda_-, \lambda_+]$ [80], where

$$\lambda_\pm = 1 + \frac{1}{Q} \pm 2\sqrt{\frac{1}{Q}}.$$

Once, tests confirm the presence of random matrix within the corresponding spectral boundaries, correlation matrix $C_{residual}$ is decomposed to two parts [86]:

$$C_{residual} = C_{rand} + C_{non-rand}, \tag{58}$$

$$C_{rand} = \sum_i \lambda_i V_i V_i^T, \tag{59}$$

$$C_{non-rand} = \sum_j \lambda_j V_j V_j^T, \tag{60}$$

where i and j run over RMT and non-RMT eigenvalues respectively, and eigenvector notation is kept as V, even though they are calculated for a residual matrix. Algorithm keeps only second term in Eq. (58) and disregards the first. The goal is to remove from correlation matrix the influence of random eigenvalues, i.e. the noisy relationship between data points. Pearson's correlation coefficients is not a particularly good metric in data space [105]. It is not positive definite, and arbitrary shift does not make it into one. Furthermore, the triangular inequality as well as zero distance axiom require additional reformulation for C_{ij}. The way out was found by Mantegna [105], who studied taxonomy of financial data with the aid minimum spanning tree of portfolios. He proposed the following alternative to the matrix C:

$$D = (1 - C)^{1/2}. \tag{61}$$

Here, the elements of distance matrix D possess all the necessary properties required by the metric axioms, and thus, can serve as a proximity measure during clustering.

The crucial step in algorithm is in returning to the idea of distance matrix, which now no longer reflects properties of sample to sample correlations. Indeed, $D = (1 - C_{non-rand})^{1/2}$. For example, if for strongly correlated samples ($C_{ij} \approx 1$), in the original matrix, we had vanishingly small distance $D_{ij} \approx 0$, now, diagonal elements of $C_{non-rand}$ are not equal to unity. As a result, the diagonal of D can be safely ignored. However, the values of other matrix elements $C_{non-rand}$ reflect the pattern we are trying to recover, in spite of the lack of interpretation as Pearson's coefficients. This makes D a perfect input for clustering algorithm.

In hierarchical clustering part of the algorithm, two clusters are merged such that, after merger, the average pairwise distance within the newly formed cluster, is minimum. Clusters having minimal distance iteratively merged, such that the new cluster, on average, possesses minimum pairwise distances between the points in it.

External validation of obtained clusters is possible if some portion of data is labeled [106]. Jaccard and Rand indices show the level of agreement between a set of class labels $\mathcal{C}$ and clustering result $\mathcal{K}$. The Jaccard index is determined by the number of point pairs assigned to the same cluster in two partitions:

$$J(\mathcal{C}, \mathcal{K}) = \frac{a}{a + b + c}. \tag{62}$$

Here a stands for the number of pairs of points with the same label in $\mathcal{C}$ and assigned to the same cluster in configuration $\mathcal{K}$, b denotes the number of pairs with the same label, but in different clusters, and c is the number of pairs in the same cluster, but with different class labels. The Jaccard index produces a result in the range of 0 and 1. The value of 1.0 indicates that C and K are identical. Rand index is normalized to unity and positive:

$$R(C,K) = \frac{a+d}{a+b+c+d}, \tag{63}$$

A new variable d stands for the number of pairs with a different label in C, assigned to a different cluster in $\mathcal{K}$. Note that high value for this index generally indicates high degree of agreement between clustering and the annotated natural classes.

Internal validation checks that clusters are compact and clearly separated. For any partition of clusters, let c_i represent the i-th cluster of such partition. The Dunn's validation index d is computed according to:

$$d = \min_{1\leq i\leq n}\left[\min_{1\leq j\leq n, i\neq j}\left\{\frac{\delta(c_i,c_j)}{\max_{1\leq k\leq n}\{(\delta'(c_k))\}}\right\}\right], \tag{64}$$

where $\delta(c_i,c_j)$ - distance between clusters c_i and c_j, $\delta'(c_k)$ - intracluster distance of cluster c_k, and n is a number of clusters. The main goal of the measure is to minimize the intracluster distances and maximize the intercluster distances. Note, that the number of clusters maximizing d, has to be optimal.

The Silhouette index is another popular measure of validity. As described in [106], the Silhouette validation technique calculates the silhouette width for each sample, averages silhouette width for each cluster and overall average silhouette width for the entire data set. Using this approach each cluster could be represented by the silhouette, based on the comparison of cluster's separation and tightness. The average silhouette width is then applied to evaluation of clustering validity and decision on goodness of the number of selected clusters. In order to construct the silhouettes $S(i)$, the following formula is used:

$$S(i) = \frac{(b(i)-a(i))}{\max(a(i),b(i))}, \tag{65}$$

Algorithm 1 RMT based Hierarchical Clustering Algorithm

1. For data matrix A of size $N \times L$, where N is number of data points and L is number Of variables, compute correlation matrix C of size $N \times N$
2. Compute correlation matrix $C_{residual}$ removing the influence of largest eigenvalue λ_{max}
3. Compute eigenvalues λ and eigenvectors V of $C_{residual}$
4. Identify boundaries $[\lambda_-, \lambda_+]$, run RMT tests.
5. Using results of step 3, run decomposition: $C_{residual} = C_{rand} + C_{non-rand}$.
6. Transform matrix $C_{non-rand}$ to distance matrix D, according to $D = (1 - C_{non-rand})^{1/2}$.
7. Build hierarchical tree of distances.
8. Assign clusters based on cut off value.
9. Validate clusters internally and if labels are available externally.

where $a(i)$ is an average dissimilarity of i-th object from all other objects in the same cluster, and $b(i)$ is a minimum of average dissimilarity of i-th object from all objects in another (closest) cluster.

As we see from Eq. (65) that $-1 \leq S(i) \leq 1$. If it is close to 1, the sample is thought to be 'well-clustered' and is assigned to an appropriate cluster. If silhouette value is close to zero, it means that the sample has to be assigned to another closest cluster as well, and the sample lies equally far away from both clusters. Now, if silhouette value is close to -1, the sample is taken for 'misclassified' and is thought to lie somewhere in between the clusters. The overall average silhouette width is average $S(i)$, computed for all objects of the entire dataset. Note, that the largest overall average silhouette indicates the best number of clusters.

D Clusters in leukemia dataset

Early diagnostics is crucial for successful treatment. Motivated to identify and predict two close types of leukemia Golub *et all* have introduced the dataset with 72 bone marrow samples of acute myeloid leukemia (AML) and acute lymphoblastic

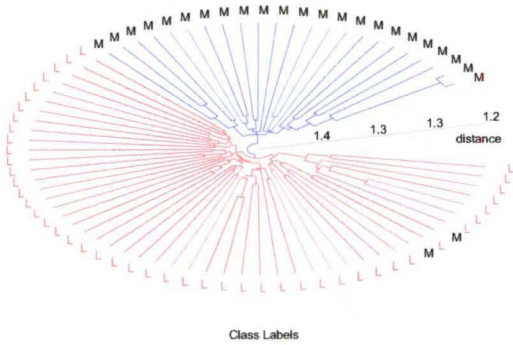

Figure 21. Polar dendrogram of denoised distance matrix, terminal nodes are labeled with respect to their ALL or AML correspondence."L" is ALL, "M" is AML.

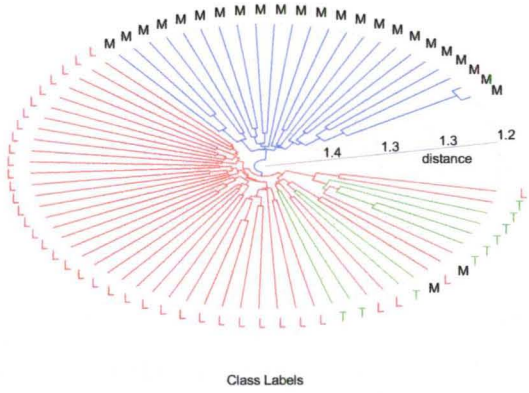

Figure 22. Polar dendrogram of denoised distance matrix, terminal nodes are labeled with respect to their ALL B-/T-cells or AML correspondence."M" is AML,"L" is B-cell ALL and "T" is T-cell ALL.

leukemia (ALL) [94]. Dataset became a clustering benchmark since it is real, not-simulated, typical gene expressions array with very large number of genes and small number of samples.

Data matrix A with $L = 7129$ genes and $N = 72$ bone marrow samples is publicly available at: http://www.broad.mit.edu/cgi-bin/cancer/data sets.cgi. The results of walking the data through the RMT based clustering algorithm visualized in Figure 21. At the cut off distance 1.5, hierarchical three splits into two branches, and into three branches at the cut off distance 1.46, which is shown in Figure 22.

Optimal number of clusters identified by RMT based clustering is three, which

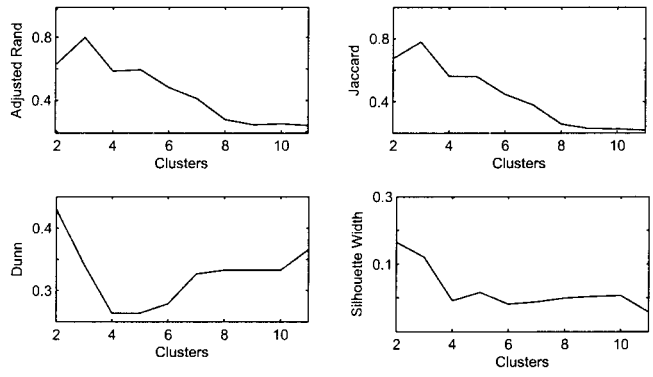

Figure 23. Validating Clusters indices. Adjust Rand and Jaccard indices are so called "external" validators, they use the external available information about class assignment. Dunn and Silhouette indices are "internal", they evaluate individual quality of the cluster, such as its compactness and separability from another cluster.

validated by highest Rand and Jaccard indices in Figure 23. It turns out that ALL samples are of two types: from T-cell and B-cell lymphocytes. Several algorithms developed for micro array clustering and tested on Golub's data are able to recognize it [107].

Judged not by available labels but rather by quality of clusters themselves, RMT based clustering produces two most compact and easily separable clusters corresponding to AML and ALL types.

The comparison of partitional and hierarchical clustering algorithms on this dataset is presented in Figure 24. The data are subjected to a series of standard pre-processing steps: lower and upper threshold values (raw expression values of 100 and 16 000, respectively) are applied, the 100 genes with the largest variation across samples are selected, and the remaining expression values are log-transformed. The resulting dataset of size 38 x 100 is subjected to a cluster analysis under Euclidean distance. Altogether, evidence accumulation over the set of employed validation techniques indicates a high quality of the three-cluster solution discovered by k-means, SOM, SOTA and average link. The evaluation under the adjusted Rand Index (comparing to the known class labels) shows that average link, k-means, SOTA and SOM perform robustly on these data. They identify the three main clusters (AML, B-lineage ALL and T-lineage ALL), and assign most of the samples correctly.

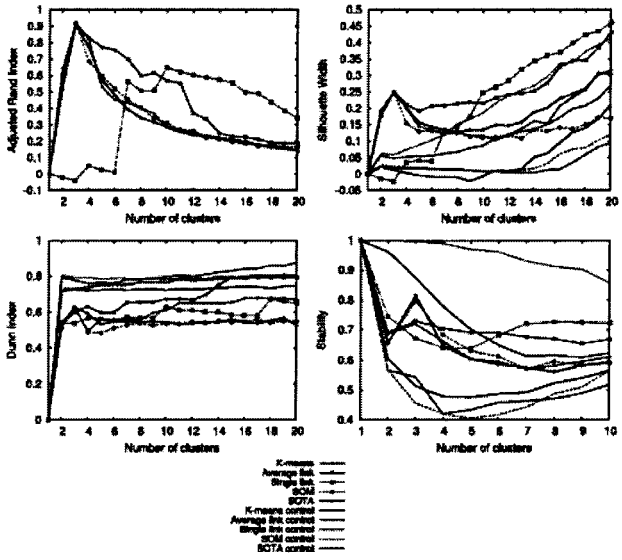

Figure 24. Adjusted Rand Index, Silhouette Width, Dunn Index and stability (averages over 21 runs) for k-means, SOM, SOTA, average link and single link agglomerative clustering on the Leukemia test set, adapted from [107].

Naturally, this is knowledge that would not be available in a real-life cluster analysis, and it is therefore interesting to see whether the results under the internal validation measures would have led to the same conclusion. The performance curves under the Silhouette Width clearly indicate the high quality of the three-cluster solution. The stability-based technique is less consistent: for k-means and SOM, the performance peak at k=3 is well pronounced, but it is much weaker for SOTA and average link. Both the Silhouette Width and the stability-based method indicate the lack of structure in the single link solutions. The application of the Dunn Index is somewhat less successful: it fails to predict the insufficiency of single link, and it mis-estimates the number of clusters for average link.

E Summary of clustering results

The luxury of having labeled sample in the dataset is not always present. Thus proper clustering among other unsupervised techniques is the only source of reliable information for further modeling or classification activities.

Various clustering techniques have been used on the popular benchmark dataset of two close types of cancer with extremely large number of variables and statistically

low number of samples. All of them correctly identify cancer types and even cancer subtypes at the expense of significant variables reduction [107]. Our experiments with removing the noise from correlation matrix of bone marrow samples, showed that clear and correct clusters may be obtained from refined distance/similarity matrix without reducing the number of genes. Our algorithm is scalable, in fact it benefits from higher dimensionality of the data. It does not rely on any dimensionality reduction technique and its quality. Since the number of genes in obtained cluster is intact, further genes clustering may help to identify clusters of genes within particular type of cancer.

With RMT tests our algorithm identifies presence of random noise in the correlation matrix of data points. Keen decomposing procedure undresses the correlation matrix from noise. After astute transformation, elements of de-noised correlation matrix possess all the necessary properties required by the metric axioms, and thus, can serve as a distance measure during clustering. With such distance matrix correct classes of dataset may be obtained even with the simple top-down hierarchical clustering.

CHAPTER VII

FEATURE CONSTRUCTION WITH DYNAMIC DATASET

Having discussed feature extraction for time series analysis with the help of equal time correlations, we then turned our attention to time-lagged covariances. The primary idea behind that was to bring the time back into time series analysis. For equal time covariance, we established the distinction and roles of different groups of eigenvalues and eigenvectors. By analogy with financial applications, where lead-lag relationships between various stocks are of great interest, we could try to show that such relationships exist between different network elements. However, the most interesting aspect for us, was once again, finding a possibility for detecting anomalous patterns.

Temporal evolution of network traffic is a non-linear dynamics problem. We can expect relationships between variables to be highly unstable. Yet, whatever pattern formation we detect, can be put to service for feature extraction. Chaos in our system could either be low- or high-dimensional. In simple terms, low-dimensional chaos implies some level of short-term predictability, while high-dimensional ceases to yield any sort of prediction.

In a given network, the exact nature of chaotic behavior is unknown up-front. Part of our consideration, is getting a better idea about degree of chaos in our network. What is even more important for our present discussion, is that when studying time-lagged correlation of network traffic we cannot expect to have any proportionality between cause and effect, e.g. between external disturbance and system's reaction.

Spectral analysis, we use in this Chapter, is established tool for finding hidden periodicities. Our objective is to find any inherent periodicities in the system's power spectra, i.e. frequency content of various correlators, bi-linear in time series. Then, we would want to look at the new, if any, periodicities, induced by altered time series. Note, that we do not expect discernible few frequencies in completely chaotic power

spectra. That is why we also need the RMT methodology. Spectral analysis on its own is insufficient for feature extraction, as empirical identification of chaotic behavior is not clear cut.

A Additional Motivation for Time-Lagged Correlation Matrices

Long-range dependent (LRD) processes, which show significant correlations across large time scales were first discovered in network traffic over a decade ago [95]. Since then, LRD was found and studied intensively in various aspects of network behavior. Such dependence is a manifestation of self-similarity of the process, an important property, simplifying modeling large networks. Its basic meaning is scale-invariance of the process in space and time.

The first rigorous statistical analysis of self-similar characteristics in Local Area Network (LAN) traffic was done by Leland *et al*[96]. They showed that the aggregated Ethernet traffic is not smoothing out in accord with Poisson model, and is time scale invariant. In this framework the traditional Poisson or memory-less models of network traffic became inadequate. Since high variability across different time scales produces high level of congestion, the impact of self-similarity on network performance is proven to be considerable [97].

In our work, we employed the time-lagged correlations of the network traffic system for slightly different purposes. Knowing the boundaries of random and group eigenvalues, we attempted to trace the appearance of meaningful interactions in time series and their evolution in time. Our hypothesis was, that eigenstatistics of non-random interactions present in traffic, would scale with time, i.e would signal the LRD. In addition, we expected drastic changes in the inverse participation ratio, compared to that of equal-time correlations. As we demonstrated already, the IPR presents a concisely convenient visualization of traffic load intensity. Yet, we expected time-lagged IPR to be even more illustrative in terms of changing in time intensity of traffic load patterns.

B Time-lagged correlation matrix of network traffic time series

The starting point of our discussion is again, averaged traffic count data collected from router-router and router-VLAN subnet connections of the University of Louisville backbone routers system. To construct a lagged correlation matrix, we had taken $L = 2015$ records of $N = 497$ time series averaged over 300 seconds, where incoming and outgoing traffic generate independent time series. We defined normalized traffic rate change $g_i(t)$ according to Eqs. (35 and 36) and build the time-lagged correlation matrix $D(\tau)$ [98] according to

$$D_{ij}(\tau) \equiv \mathrm{Sym}\langle g_i(t) g_j(t+\tau)\rangle = \frac{1}{2L}\sum_{t=0}^{t=L}\left(g_i(t) g_j(t+\tau) + g_j(t) g_i(t+\tau)\right) \qquad (66)$$

The sole purpose of symmetrization is the restriction of the eigenvalues and eigenvectors to real values. This is undoubtedly a significant simplification of subsequent analysis. In principle, the numerical experiments we ran below can be repeated for the eigensystem of non-symmetric correlation matrix. Studies of the spectral properties of such matrices are already in progress (see, for example, [108]). The analysis of this work takes place in a different setting, and the goals are somewhat opposite to what we had put before us. Financial time-lagged correlation matrices help to reveal "networks" of stocks, and we already have a network.

C Selecting eigenstatistics of time-lagged matrix as indicators of network behavior

Our original motivation of extracting the most concise indicators of network's behavior, remains unchanged. More specifically, we are after efficient indicators to help defining "normal" state of the system, and predict structural reaction to the external or internal disruption. We followed our general direction - finding reduced set of features sufficient to represent network's behavior. The candidates are those eigenvalues, that would be most receptive to a particular probe.

First of all, we defined an eigen problem for each time delay increment τ, thus making our cross-correlation matrix is time dependent

$$D(\tau) u_k(\tau) = \lambda_k(\tau) u_k(\tau), \qquad (67)$$

Here λ_k is k-th eigenvalue, corresponding to kth eigenvector u_k.

As opposed to same-time eigensystem $\{\lambda_k(0), u_k(0)\}$, our eigensystem does not characterize presence or lack of organization in the system at a given time. Instead it serves as a measure of back (or forward) in time covariance within network structure. Some of the network nodes might be driving others; some of the nodes can be genuine resistant toward the most intense traffic through the others. Non of these or similar existing scenarios could be inferred from the connectivity matrix only. Just as in our earlier approaches, we had searched the possibility of locating those few features, that bear the most exhaustive information on the network dynamics.

The main difference is of course, absence of usual RMT picture of eigenvalues spectrum being split into three parts. By three parts we understand the central - RMT part, which is responsible of universal behavior, and its side - "left" and "right" neighbors which exhibit non-universal features [109]. Although, for very small τ this subdivision is clearly still accurate, we expected, transient behavior of $\{\lambda_k(\tau), u_k(\tau)\}$ to reveal new, otherwise undetectable correlations within the network. Hence, we found it convenient to keep track of quantitative and qualitative changes in eigensystem using left, random, and right terminological distinction.

In addition, we decided to look at the evolution of corresponding IPRs. Given the eigenvector $u_k(\tau)$ the IPR is computed according to

$$I_k(\tau) \equiv \sum_{l=1}^{N} \left[u_k^l(\tau)\right]^4, \tag{68}$$

with u_k^l, $l = 1, \ldots, 497$ being components of the kth eigenvector [65]. For non-zero values of τ, IPRs acquire more general meaning in a sense that routers which interact heavily at time t may loose their "bond" at time $t + \tau$, while those not knowing about one another at time t may acquire significant level of interaction at time $t + \tau$. Other more complex possibilities can be perceived via $I_k(\tau)$ as well. For example, if normal state of the traffic becomes altered by DDOS.

D Eigenstatistics of time-lagged correlation matrix as visual analytics

The usefulness of eigenstatistics of time-lagged correlation matrices became apparent when turned to visual analysis of network monitoring and congestion control.

We visualize the time dependence of different parts of eigenvalues spectra. We also did the same for the respective IPRs. Once we discovered the oscillatory patterns for parts of the eigenvalue spectra we immediately switched to analyzing frequency content of transient behavior. We employed the simplest tool of frequency domain analysis - the Fast Fourier Transform (FFT). Knowing the content of frequencies is the same as possessing the characteristic time scales of a given traffic load formation, which is of great use for modeling the "normal" traffic load, monitoring the congestion level, and traffic anomaly detection.

1 Stroboscopic sequence for eigensystem

Upon building the cross-correlation matrix $D(\tau)$ with the help of Eq. (66) we performed eigen-decomposition (Eq. (67)) numerically, using standard *MATLAB* routine. We then looked at the resulting eigenvalues evaluated at all times τ. A noticeable spike for very small values of delay time is expected, notwithstanding the position in spectrum. However, our increments in τ ($= 300\,sec$) may not be small enough to observe it. For the remainder of observation the result has to uncover the way system constituents communicate with themselves and their neighbors on a long run.

In Figures 25 (a)-(c) we illustrated how left, random, and right groups of the spectrum evolve with time delay value τ. As it turned out, "randomness" and "regularity" found their new interpretations in the context of system reminiscing itself. With an exception of a few located at the right and left edges of the spectrum, most eigenvalues are very close to each other numerically. To make the evolution picture more transparent, we plotted their τ-dependence using different offset values (these values are the same within each part). Only ten eigenvalues are offseted in each case and plotted versus time. The lowest eigenvalue was excluded from consideration here and throughout the paper due to its secular - grossly linear - behavior in τ.

At a glance, non-edge eigenvalues Figure 25 (b) safe for an expected spike at small τ does not seem to represent any process. Such a lack of forward-in-time correlation is not completely surprising, as the eigenvalues from middle part of the spectrum have their origin in RMT part of the spectrum. We termed them RMT-like

from that point on.

It follows, that these random interactions between traffic time series are time delay invariant. In other words, random spectrum of eigenvalues is an appropriate indicator of traffic's self-similarity [116]. Meantime, the eigenvalues at the edges (Figures 25 (a) and (c)) represent a quasiperiodic process, distinguishing themselves from their RMT-like peers. These eigenvalues having their ancestors located in regular part of the spectrum for $\tau = 0$ clearly display long time dependence [116]. Therefore, it makes sense to look further into the properties of edge eigenvalues, especially into the properties of those with relatively high absolute values. The actual values can be used as a measure of delayed time correlations, as they are related to traffic variances [79]. Having observed these sharply distinct trends in eigenvalues delayed correlation matrix $D(\tau)$, we took a closer look at a 'derivative' spectral characteristics, the IPR.

A remarkable property of IPRs for equal time cross-correlation matrix (Figure 26 (a)) was its consistently low, order $1/N$, value for the major part of the spectrum. This segment in Fig. 26 (a) is known to obey the RMT [109]. To the left and to the right from this segment there is a strong evidence of regular, non-random eigen statistical behavior. When the first 20 instances are considered as in Figures 26 (b) and (c), with IPRs offseted by an arbitrary amount for transparency, and plotted versus the eigenvalue position, we see drastic difference. The peak, located close to the center of the spectrum, signifies presence of previously unrevealed correlations, and the lead-lag relationships amongst time series.

Close examination of Figures 26 (b) and (c) shows, that initially, the high IPR had changing support in the spectrum. Peak value told us, that about four time series drive the entire correlation pattern. Later on, the peak "settles down" and establishes itself around median eigenvalue position (Figure 26 (b)). The meaning of this and other two peaks differs from that of the IPR peaks in Figure 26 (a). The increase in IPR computed from the time delayed matrix $D(\tau)$ indicated correlations between system's behavior at a given time and system's stroboscopic image after τ elapsed, rather than correlations within the spectrum. In addition, it provides reasonable way of tracking down the sources of lead-lag behavior. Thus, the observed features make IPRs into good candidates as indicators of network's congestion state. Note also, a

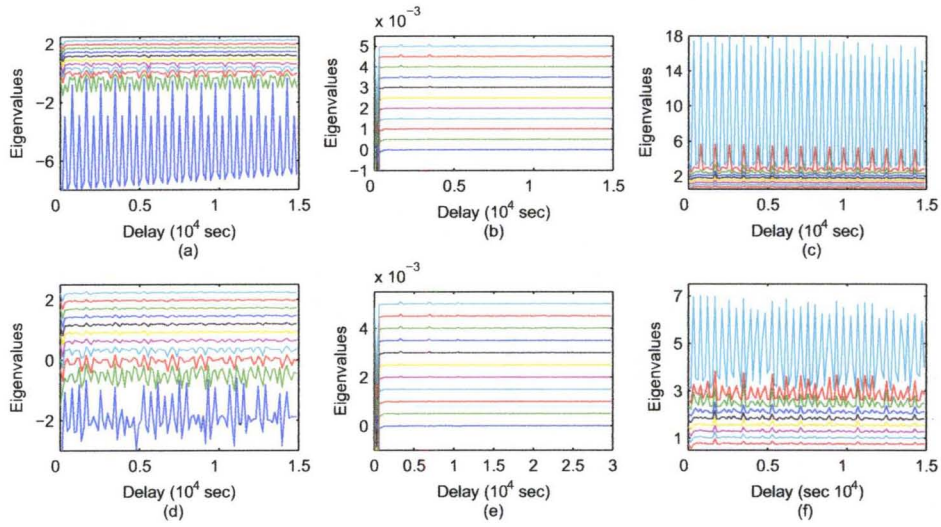

Figure 25. (a) left, (b) random, and (c) right parts of the eigenvalue spectrum as obtained from actual data. Same graphs are presented in (d), (e), (f) respectively, after noise-like injections are made.

significant change in height of the central peak.

2 Frequency domain analysis

Thus, we prepared to the next step in getting more quantitative on the subject of long memory processes in network traffic. We proceeded to analyze transient behavior of eigenvalues and IPRs of matrix $D(\tau)$ in further detail. Since quasiperiodic behavior is present in the majority of quantities of interest we focused on their precise frequency content. The standard way of analysis is to transform $\lambda_i(\tau)$ into frequency domain using fast Fourier transform. In a sense, we constructed a spectrum of the spectrum. The same operation was performed on respective IPRs.

We took fast Fourier transforms for all the functions at hand, and then, took the square of their absolute value. The result can be called power spectrum. There should be no confusion, as graphs of power always accompany the corresponding time-domain quantity.

In Fig. 27 we display representative eigenvalue dynamics. Once again, RMT-like λ_i (Fig. 27 (b)) did not exhibit anything remarkable, compared to its regular counterparts. The eigenvalues, taken from left and right parts of the spectrum, resembled each other, reflecting a symmetry of the spectra, induced by the

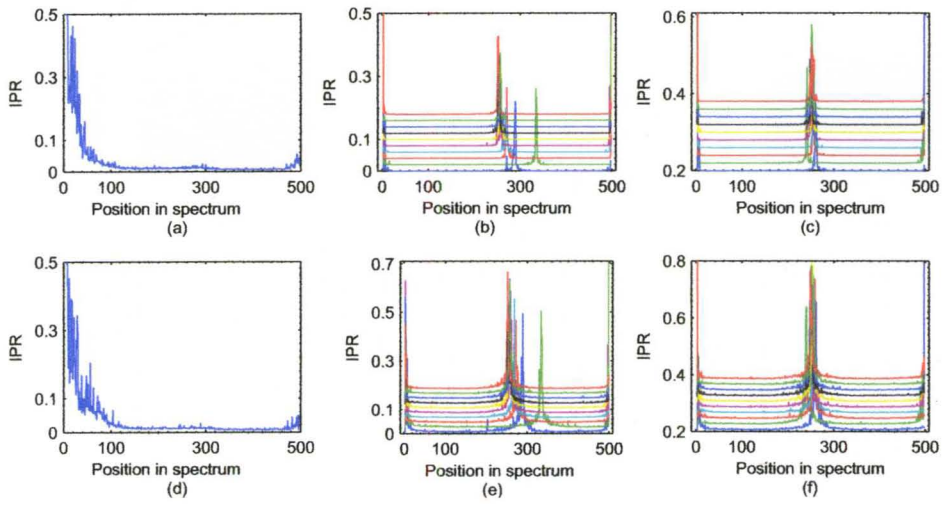

Figure 26. (a) $I(0)$ versus position in spectrum. Stroboscopic representation of IPRs corresponding to (b) first 10τ (c) second 10τ; (d)-(f) are the same representations upon noise-like injections.

symmetrizing procedure (Eq. (66)). From that point on, we considered them in parallel.

Aside from a substantial low and high frequency contribution, which was expected from Figs. 27 (a) and (c), we discovered two strong contributions from frequencies, corresponding to oscillations with time periods 15 and 30 minutes respectively (cf. Figs. 27 (d) and (f)). This is in evident contrast to the situation with power of a random eigenvalue. Such an eigenvalue has equal (and negligibly small) contribution from the entire range of frequencies. The existence of these two characteristic frequencies suggests a natural way assessing the current state of the inter-domain network traffic. This fact makes it possible to use these as the LRD quantifiers [116] in the future.

E Experiments with altering actual network traffic

To demonstrate the use of our visual indicators in network behavior anomaly detection, we conducted two types of experiments. We altered the original traffic data by introducing the noise-like and periodic injections to the traffic. The former consisted of inserting the traffic counts originated by random number generator into

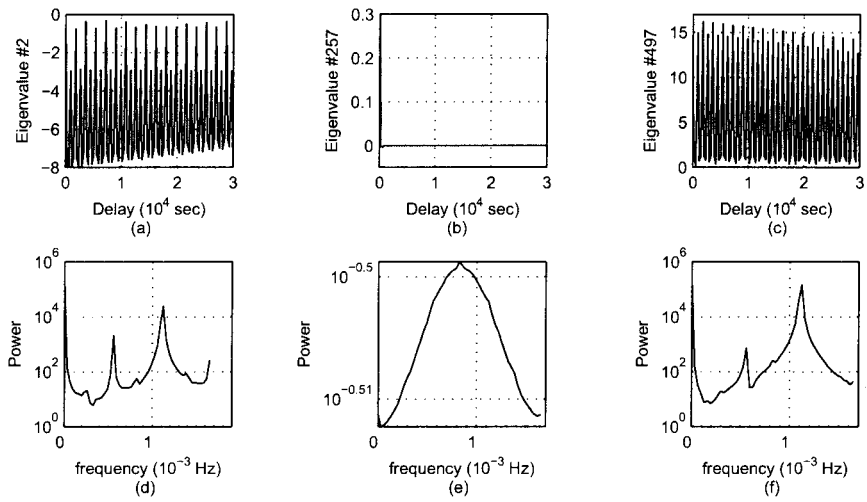

Figure 27. Eigenvalues number (a) 2, (b) 257, and (c) 497, plotted with respect to time and their respective Fourier spectra ((d) through (f)).

the traffic time series. The latter was achieved by replacements of actual network traffic with known functions of time delay τ.

1 Noise-like injections

We investigated consequences of modifying the time-lagged correlations between time series. We have already known the time series contributing the most to the correlation pattern [109]. All of them can be linked to eigenvalues, which fall into the right segment of eigenvalue spectrum. In these series we replaced the original traffic with counts obtained by random number generator for a certain period of time. Then, we constructed matrix $D(\tau)$ for all hundred increments and repeated manipulations described above. The results are shown in Figure 25 (d) through (f).

The eigenvalues, belonging to the middle of the spectrum, remained completely unaffected, i.e. they are still time delay invariant. Clearly, our manipulations with the traffic had not disturbed self-similar nature of delayed correlations. However, edge eigenvalues lost the time scales, present in their original transient behavior (see Figure 25 (d) through (f)). In other words, the LRD got destroyed.

The effect on IPR (Figures 26 (d) and (f).) was less noticeable but was still there, while for the random segment it was absent. The result of random counts

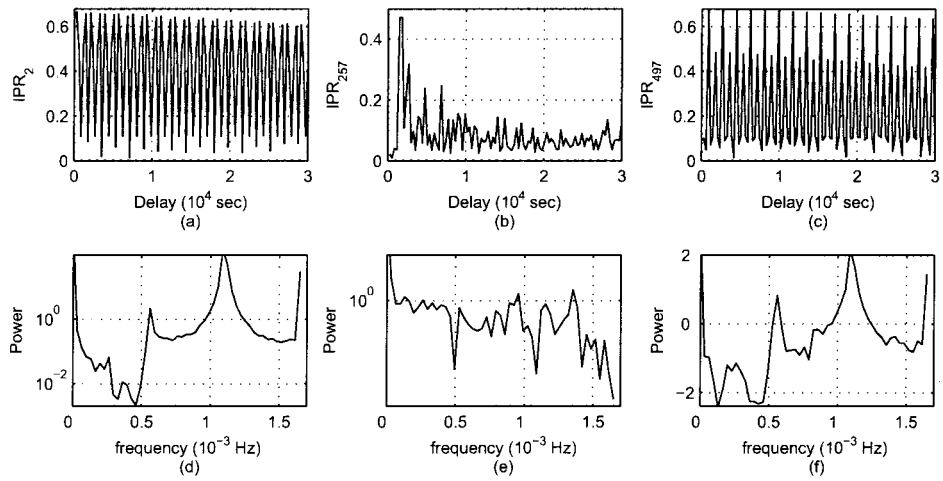

Figure 28. IPRs for eigenvalues number (a) 2, (b) 257, and (c) 497, plotted with respect to time and their respective Fourier spectra ((d) through (f)).

injections can be summarized as a presence of randomly positioned small peaks, superimposed on the original IPR picture. Indeed, in Figures 26 (a) and (c) small peaks are very infrequent and unstable in time, unlike peaks in Figures 26 (d) and (f).

The above outcome calls for a more close look into eigenvalues and IPRs of a system, experiencing noisy injections into its time series. We presented three eigenvalues as functions of time delay, together with their respected power spectra. As can be inferred from Figure 29 (a) and (c), the time dependence looses its LRD structure. It is backed up by the fact that a lot more frequencies contributed to power spectra upon random injection. Middle part of the spectra also undergoes certain transformation, but is still scale-free Figure 29 (b), as actual values of power are small relative to the power corresponding to edge eigenvalues. The quantitative changes are also in place for both edge eigenvalues. The effect can be judged based on comparison of the tallest peaks in Figures 29 (d) and (f) to their counterparts in Figures 27 (d) and (f).

Similar conclusions could be derived for the IPR, as we had taken a look at Figure 30 (d) and (f) and compared the outcome of our experiment with the graphs in Figure 28 (d) and (f).

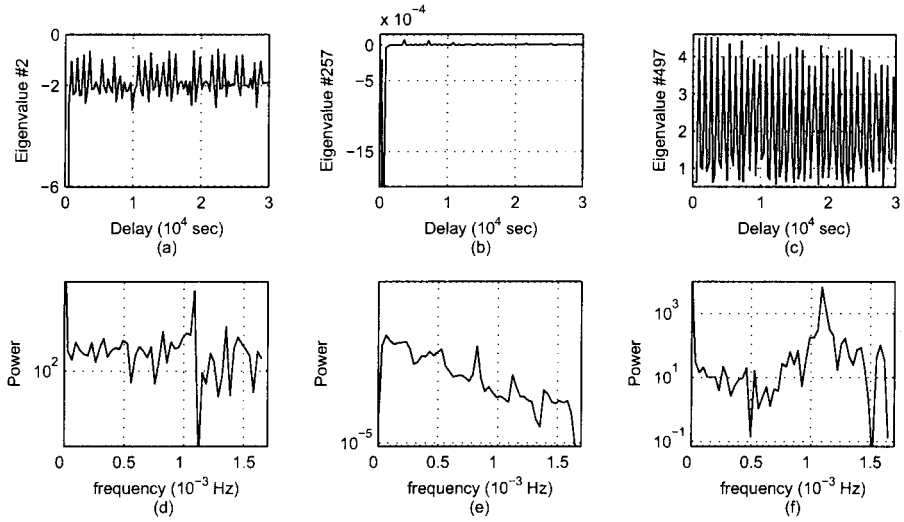

Figure 29. Eigenvalues number (a) 2, (b) 257, and (c) 497, plotted with respect to time and their respective Fourier spectra ((d) through (f)) after noise-like sample was injected.

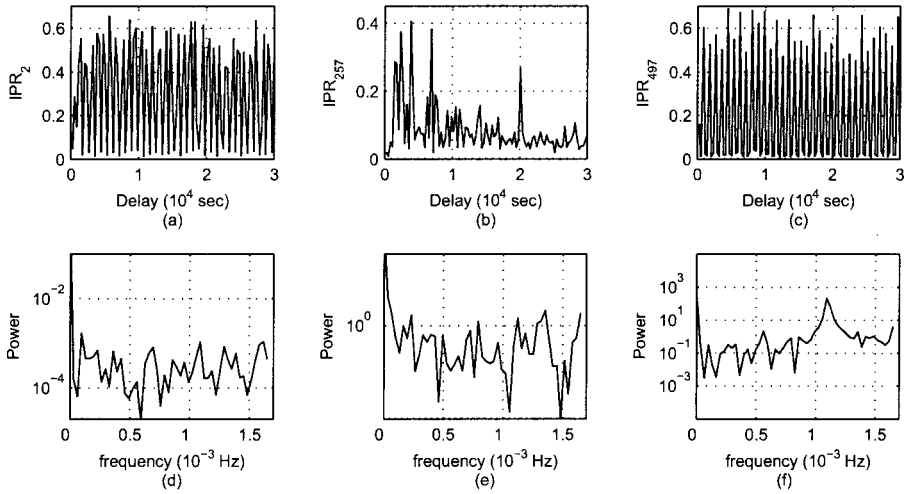

Figure 30. IPRs for eigenvalues number (a) 2, (b) 257, and (c) 497, plotted with respect to time and their respective Fourier spectra ((d) through (f)) after noise-like sample was injected.

2 Periodic in time injections

A continuation of the above experiment was the injection of an artificial traffic counts which possessed regularity into actual experimental data. This time, however, we performed the replacements for the time series, which could be traced back to the eigenvalues falling into the random segment. Time series for this replacement were chosen at random. Other possibilities could have also been considered, but since random segment was much less sensitive to the previous experiment, the above choice appeared natural.

We chose four injections to be co sinusoidal, having periods of 2.5; 15; 20 and $30\,min$ and repeated the same manipulations as in the first experiment, we just discussed. The results turned fairly sound. Even though the random part of the eigenvalue spectrum was again unaltered, the "reaction" of left and right parts was observable - both qualitatively and quantitatively. For a cosinusoidal sample, with period much smaller, than both characteristic periods (15 and $30\,min$), the resulting power spectra in Figures 31 (d) and (f) are not significantly changed. The two characteristic periods were still present, and yet certain narrow frequency range got suppressed - note the anti-peak between the two main peaks.

We also noticed slight asymmetry in the way smallest and largest eigenvalues had reacted to the injection. We should add, that observed picture is essentially the same for the injections with periods of 5 and 10 minutes. This was no accident - both time scales although not matching, were commensurate with the characteristic periods.

After that, we turned to the result displayed in Figure 32, where the cosinusoidal replacement with period $15\,min$ of actual traffic counts lead to the dramatic change in appearance of power spectrum. We observed enhancement of the peak corresponding to period of $15\,min$, which can qualify as a resonance phenomenon (Figures 32 (d) and (f)). The very same plots showed the suppression of peaks, corresponding to the other characteristic period of $30\,min$. Similar resonant effect was achieved. when the period of injection had been changed to $20\,min$ (see Figure 33). This time, both peaks were gone, while the new characteristic period appeared in Figure 33 (f). The period had approximately matched the period of injection. And finally, for the experiment, in which period of the injection was chosen to be $30\,min$, i.e. matching to another

characteristic period, we obtained yet another result supporting previous conclusions.

In this case, however, the resonance phenomenon was slightly more difficult to establish. From the results displayed in Figure 34 we saw, that relative contribution to power spectrum is now changed for two main peaks. Before the experiment was performed, the higher harmonic (smaller period) dominated, overshooting its counterpart by a few orders of magnitude. After running the experiment, this had still been the case for the spectrum of largest eigenvalue, but the difference became marginal (see Figure 34 (d)).

At the same time, for the left most eigenvalue, we had determined, that lower harmonic (period, matching the period of injection) contributed the most to the power spectrum (Figure 34 (d)). Two power spectra for the edge eigenvalues were no longer symmetric, and contributions from certain ranges of frequencies were again strongly suppressed. As for the random eigenvalue considered in Figures 34 (b) and (e), no impact had been recorded, just as in all other cases.

The found resonance effect adds more strength to the proposed indicators of network behavior. If the anomalous traffic event occurs at the time interval, which is characteristic for some system-specific traffic load, it causes the most dramatic change in visual representation of our indicators. Furthermore, when the period of injection coincides with one of the characteristic time scales of the network (i.e. oscillation periods of edge eigenvalues) the corresponding spectral peak gets enhanced. The Fourier transform peak, corresponding to the other scale gets suppressed and sometimes even annihilated. Finally, injection with the period much less than both scales has little effect on Fourier spectra, while period of the same order in magnitude rearranges the original spectra completely.

F Discussion of results in the context of traffic long range dependence and other applications

Network traffic analysis had undergone the evolution from considering the network traffic time series as an outcome of Poisson and memory-less processes to recognizing the long range dependencies and self-similarity of the traffic. We found, that statistics of eigenvalue spectrum and IPRs of eigenvectors of time-lagged

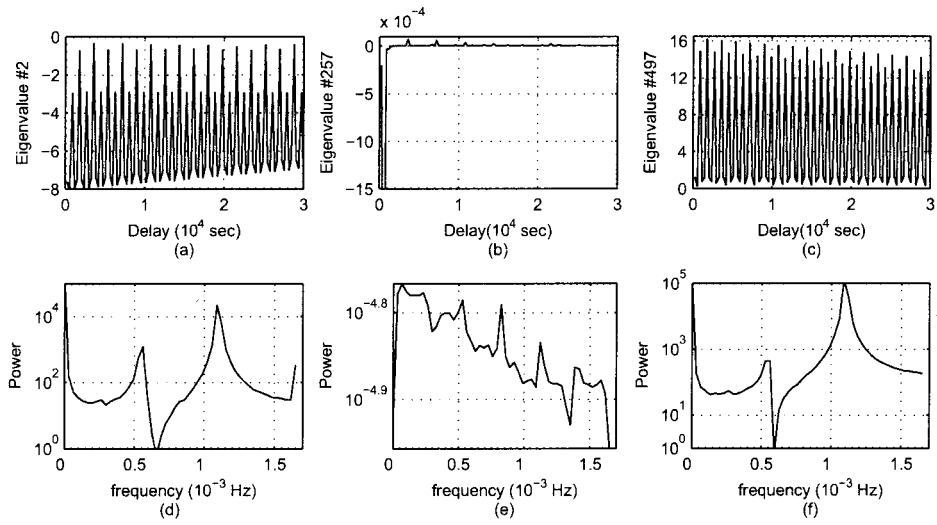

Figure 31. Eigenvalues number 2, 257, and 497: The results of the injections with 2.5 *min* period.

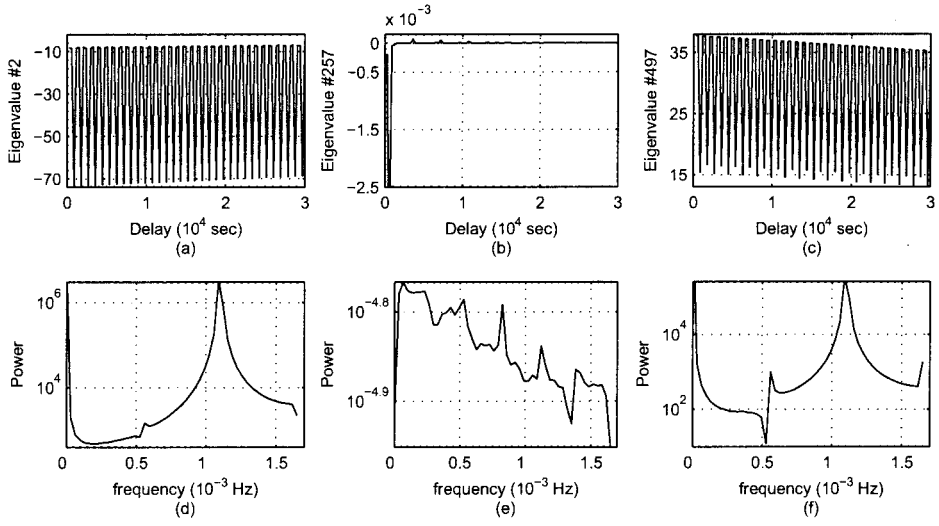

Figure 32. Eigenvalues number 2, 257, and 497: The results of the injections with 15 *min* period.

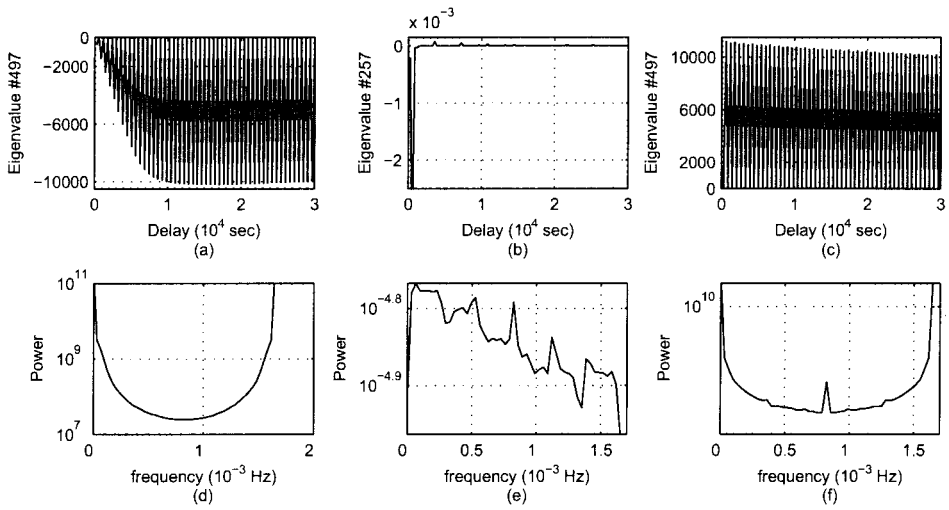

Figure 33. Eigenvalues number 2, 257, and 497: The results of the injections with $20\,min$ period.

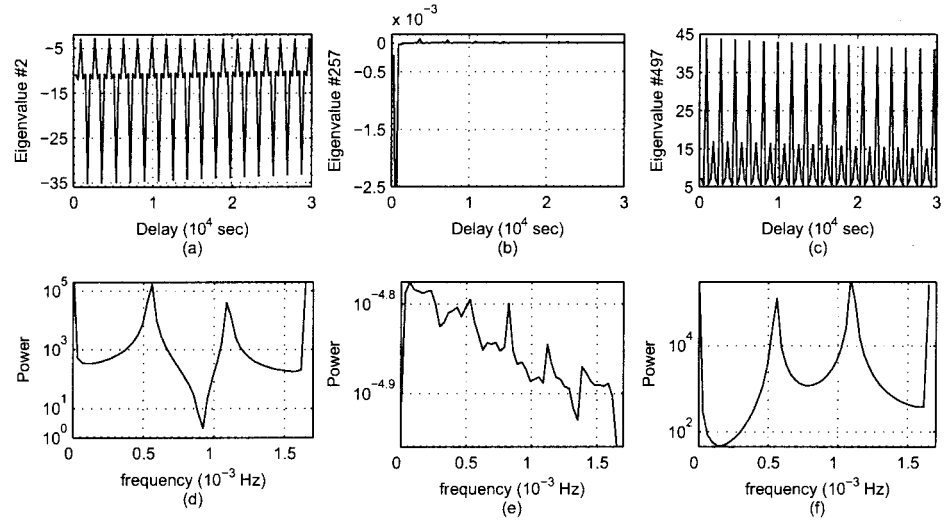

Figure 34. Eigenvalues number 2, 257, and 497: The results of the injections with $30\,min$ period.

correlation matrices provide essential dimensional reduction in the investigation of long-ranged dependence (LRD) of the network traffic.

We also demonstrated that the time delay invariant behavior of non-edge eigenvalues of $D(\tau)$ reflects the self-similar nature of delayed correlations. Meanwhile, the time-scaling of edge eigenvalues or their lagged-time dependence is an expression of self-similarity in delay correlations. In addition, we established that the IPRs of eigenvectors computed from $D(\tau)$ can be used in feature extraction and building a realistic model of network congestion.

It is noteworthy, that IPRs for $D(\tau)$, where $\tau > 0$, reveals the new localization trend, which has different origin from those of τ=0. The significantly increased and time delay invariant IPRs around the median eigenvalue indicate presence of lead-lag relationship between time series.

The experiments altering the original traffic time series led to several important effects. First of all, we demonstrated that tempering with time series has no effect on self-similar transient behavior of eigenvalues and IPRs, located in the middle segment of the spectrum. Yet, both stochastic and periodic injections affected non-random segments yielding dramatic changes in their temporal behavior. In particular, we recorded the destructive effect of random noise on otherwise simplistic double-peaked power spectra.

The above described time-lagged correlations analysis has a broad area of applications, where delayed correlations between system substructures are essential. For instance, it can be applied to electro-physiological time series of brain response [129], earthquake relocations [130], financial portfolios [108, 98], and atmospheric data [98]. To support this assertion we point out that edge eigenvalues of $D(\tau)$ behave almost identically to these of atmospheric data, while the delay eigenvalues of the stock market data act just like the eigenvalues, we termed random [98].

CHAPTER VIII

CONCLUSION

A RMT based algorithms and methodology behind them

Our focus in this work was unsupervised learning, that is, approach requiring no further information aside from the one already present in data. We were particularly interested in spectral methods of feature selection and extraction. After reviewing main classical approaches we turned out attention to the least successful of all, linear methods of data mining, involving classification and clustering.

As a rule, such methods start with singular value decomposition, rather than with cost function. The PCA, which is the most common and trivial representative method of this family, then attempts to find linear manifolds, which would most accurately represent original correlations in data set. Despite its popularity in engineering and life sciences, PCA's successes are just about as often as failures.

As we discussed earlier, such failures sometimes are preventable, via data preprocessing. The latter, however, requires a lot of the *a priori* knowledge, and essentially non-linear polishing. This may violate unsupervised nature of the method on the one hand, and introduce spurious information into the data, on the other.

Meantime, the majority of non-linear methods we reviewed and studied in application to the data sets of this work, are computationally complex. Sometimes - too complex to be considered feasible. These methods, for example KPCA do require considerable polishing and 'inside information'.

Hence, after several fruitless attempts to apply conventional methods to the network time-series data, as well renown bioinformatics micro array data, we turned to the methodology of the RMT. The theory of large random matrices is phenomenological tool, which is currently being developed into analytical apparatus with applications far beyond its origin in nuclear and atomic physics. Such

applications include among many others financial engineering, theory of optimal portfolios, computer and mobile networks, and bioinformatics [71].

The starting point for the methods developed in this work is, unsurprisingly, correlation matrix build through time averaging for time series, or feature averaging for micro array experiments data. Subsequent eigen-decomposition fits into a usual spectral methods template. However, our next step was a novel feature selection procedure based on the RMT.

The key principle behind it, can be expressed as follows. The noise and system specific information are tightly intertwined in the original data set, and is even more so, in the correlation matrix. The latter, being frequently used to decide on connections between pair of financial assets or similarity of samples of genes expressions, has to be meticulously scrutinized. The main issue is of course to find an effective way to unwind noise and useful information.

The RMT is a theory, which predicts spectral behavior of completely random matrices with pre-determined symmetries. Thus, we came up with an idea of using it in role, which can be roughly described, as a spectral filter. Knowing spectral statistics of completely random matrix we attempted to superimposed its eigenstatistics with ours, and come back to the feature space with significantly reduced number of relevant features.

The random Wishart matrix, i.e. matrix build out of two equal rectangular matrices through matrix multiplication, was well studied in the sixties by nuclear physicists [142]. Our correlation matrix, being build from non-random rectangular data arrays, did showed much of the features pertinent to random Wishart matrix. It also showed plenty of deviations.

It is impossible to define boundaries of randomness on the original correlation matrix. Indeed, any kind of the cut-off value, used by the direct correlation filters and applied to correlation coefficients, has to be justified. Randomness boundaries in spectrum are, on the other hand, known in most cases. By that we simply mean RMT boundaries of Wishart matrix. All the eigenvalues, and corresponding eigenvectors, which fall outside those boundaries are expected to bear meaningful information. And, of course, such eigenvectors are primary candidates for selection into reduced set of

features we look for. They can be later used in classification and clustering.

The other idea, we implemented, is to use spectral decomposition, in which correlation matrix is represented through the sum over dyadic products of its eigenvectors, weighted by eigenvalues. This decomposition allows to break original correlation matrix into three parts: bulk, based on the RMT-like, deviating one and the one based on the largest eigenvalue. Our hypothesis, proven to be true, was founded on the fact that among those three, only correlation matrix built upon deviating eigenvalues, matters in searching for a pattern. Even though such matrix loses the meaning of correlators, the matrix we term 'group' [58]in the text explores the subspace of eigen space that is rid of the overall system behavior as well a s of the genuine noise. W use this denoising procedure in data clustering.

All in all, the philosophy of noise removal through spectral statistics, is highly workable and universal. Because of the uniform approach to correlations in finance, network and biomedical sciences, the techniques we presented are suitable over a much wider range of data mining applications.

B Summary of the results

The data sets we used in this work are network traffic series and gene expressions micro arrays. In both cases we rejected the possibilities of non-linear data polishing as well as non-linear data-processing. Instead, we split the spectrum into the RMT and deviating parts, and constructed the basis for denoising.

We took on the network time series first, and discovered, that traffic alterations can be successfully diagnosed with our projection algorithm. We run statistical tests on the correlation matrix, and proved that RMT behavior is statistically present in eigenstatistics. We made sure that denoising procedure is meaningful. The use of mid-spectrum, bulk eigenvalues had indeed proven useful, because randomness was unaffected by regular temporal dependencies of alterations.

Selecting RMT-like features and building a projection kernel out of them was a key ingredient in our procedure. The resulting reconstruction error algorithm brought out consistently accurate results. Its ROC curves demonstrated, that our algorithm was superior compared to standard the PCA. The algorithm is also fast and robust. It

certainly requires no pre-processing or non-linear polishing of the original data. Most importantly, it is easy to implement in real time, on real traffic time series, with real DDOS taking place.

Upon using bulk of eigenvalues in our algorithm construction, we turned to regular part of the spectrum. Our target was benchmark bioinformatics data set of [94]. We used spectral decomposition to denoise the correlations and to build the distance matrix [105] for clustering. Without RMT based algorithm we employed, the hierarchical clustering algorithm fails. But, after we go through the denoising steps, described in Chapter V, the resulting accuracy becomes phenomenal. Furthermore, in our studies ALL and AML cancer types, we were able to discover sub-classification, which is normally notoriously difficult to unmask.

To complete our study we reconsidered network data from a different point of view. We attempted to find lead-lag relationships concealed by delay correlations matrix. Just as in our earlier experiments with traffic alterations, we injected random and non-random segments into the time series. We then compared temporal dependencies for eigenvalues and IPRs taken from different parts of the spectrum.

Our studies revealed that time series alteration has no effect on time behavior of eigenvalues and corresponding IPRs, taken from the bulk of the spectrum. In the meantime, artificial DDOS had strong effect on eigenvalues and corresponding IPRs from the left and right spectrum subparts. Specifically, large and small eigenvalues of the delay matrix, have their double periodicity destroyed by the injections. We observed suppression peaks in Fourier spectra of edge eigenvalues by noise-like insertions into time series. Mid-eigenvalues, however, remained unaffected, proving the point, that RMT part of the spectrum is largely responsible for self-similarity found in traffic.

With IPRs, the result are even more diverse and profound. The IPRs were found to change their support in the spectrum. For our specific data we were also able to determine, that about four time series drive traffic's correlation pattern. The peaks in IPR react at the traffic alterations via changes its position height. The observed effects prove IPRs to be good candidates as indicators of network's congestion state.

The most important outcome of our experiments with delay correlation

eigensystem, was resonance effect. Whenever the period of injection coincided with one of the characteristic time scales of the network - oscillation periods of edge eigenvalues - the corresponding spectral peak was increased. The Fourier transform peak, corresponding to the other scale (typically there were only two) was suppressed and sometimes annihilated. And of course, any injection with the period non-commensurate with either of network's characteristic time scales produced no effect on Fourier spectra. Yet, an injection having a period matching the above mentioned time scales, strongly influenced the shape of power spectra of eigenvalues and IPRs.

Our findings have potentially broad area of applications. That includes time series data sets, describing, for example, electro-physiological brain response, seismic activity, financial portfolio, gene expression, and climate change. Indeed, reconstruction error scheme, hierarchical clustering algorithm and delayed eigenstatistics monitoring, are entirely independent from nature of the data, as long as it is packed into arrays with number of features exceeding number of samples, and with both numbers being sufficiently large.

C Future work

It is clear that restrictions on data dimensions we mentioned above, cannot possibly have any physical grounds. In the end, array elements are unchanged under the transposition. Furthermore, construction of either zero delay or delay correlation matrix, can be done by averaging in either of two directions. In fact the majority of current works on clustering and classification is currently dealing with two-way approaches. Our future work is thus oriented towards removal of the above restrictions.

According to [56] the RMT spectral boundaries can be determined at least numerically. Once this is done, and we would want revisit the experiments we have run in this work. We would like to explore either the possibility of the other form of averaging, or investigate two-way correlators. And we would want to do it for both types of data sets: time series and micro arrays.

It is unnecessary to believe that reconstruction error procedure we developed is unique. Potential study we have n mind is concerned with finding possible alternatives and improvements. Our focus should also be on diagnostic abilities of this family of

methods. We plan on bringing in more diverse data, including sets with missing values. The goal is to test the limits and enhance power of developed methodology.

The robustness of our clustering algorithm relies on distance metric introduced by Ref. [105] into financial applications. Despite it is numerous successes including the one in present work, the metric is somewhat arbitrary, and appears to be non-unique. One of our future goals includes search for a better distance metric among functions of Pearson's coefficients and other correlation measures.

Even more challenging problem than missing values or outliers in data sets is a problem of data for which metric cannot be defined, or for which uniform shifts create more damage than remedy. Joining RMT methodology with non-metric approaches, such as, for example, those of [3] could offer a tremendous opportunities in pattern recognition.

In present work we only investigated feature selection for dynamic data set with the help of symmetrized delay matrix. An entirely different route is a study of pure delay correlations. Such an approach requires study in a larger feature space - the eigenvalues and eigenvectors are complex in this case. Similar study was done for financial data [108]. Yet, none of the algorithms we developed here were ever used in this context. We plan on using our procedures together with some of the approaches of [108], such as, for example, building a correlation matrices based on specific eigenvalues, to determine individual impact of a given time series or a gene.

Due to a success of our group based denoised correlation matrix, we would also want to try to entertain the idea of correlation matrices based on individual eigenvalues. It would be interesting to see individual impact of features on hierarchical clustering results. And last but not least is a possibility of having theoretical predictions for eigenstatistics of delayed correlation matrix. Study of [56] is the first step in this direction.

REFERENCES

[1] Nello Cristianini and John Shawe-Taylor. An Introduction to Support Vector Machines and other kernel-based learning methods. Cambridge University Press, 2000

[2] Introduction to Feature Extraction, Isabelle Guyon and Andre Eliseef, Studies in Fuzziness and Soft Computing, 1-25, vol 207, Springer Verlag, 2006.

[3] Y.-H. Taguchi and Y. Oono, Relational patterns of gene expression via nonmetric multidimensional scaling analysis, Bioinformatics, 21(6):730-740, 2005;

[4] N.S. Bolter et al., Fundamental patterns underlying gene expression profiles: simplicity from complexity. Proc. Natl Acad. Sci., 8409-8414, 2000.

[5] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein (1998). Cluster Analysis and Display of Genome-Wide Expression Patterns. Proc. Natl. Acad. Sci. USA 95, 14863-8.

[6] J. Ye, Least squares linear discriminant analysis ICML; Vol. 227 archive Proceedings of the 24th international conference on Machine learning, 1087 - 1093 (2007).

[7] Isabelle Guyon, Steve Gunn, Massoud Nikravesh, Lotfi A. Zadeh, Feature Extraction Foundations and Applications, Eds., Studies in Fuzziness and Soft Computing, vol 207, Springer Verlag, 2006.

[8] Mohamed Chaouch and Anne Verroust-Blondet, Enhanced 2D/3D Approaches Based on Relevance Index for 3D-Shape Retrieval Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006 (SMI'06).

[9] H. Stoppiglia, Méthodes statistiques de sélection de modèles neuronaux ; applications financières et bancaires, Ph.D. thesis l'Université Pierre et Marie Curie - Paris VI (1997);

[10] L. Oukellou, P. Aknin, H. Stoppiglia, and G. Dreyfus. A new decision criterion for feature selection: Application to the classification of non destructive testing signatures, European SIgnal Processing Conference (EUSIPCO, 1998), Rhodes, 1998.

[11] T. N. Lal *et al*, Embedded Methods, Studies in Fuzziness and Soft Computing, 137-165, vol 207, Springer Verlag, 2006.

[12] G. Dreyfus and I. Guyon, Assessment Methods, Studies in Fuzziness and Soft Computing, 65-88, vol 207, Springer Verlag, 2006.

[13] Jolliffe I.T., Principal Component Analysis, Springer Series in Statistics, 2nd ed., Springer, NY, 2002.

[14] Cox, T.F., Cox, M.A.A., Multidimensional Scaling, Chapman and Hall, 2001.

[15] W. Duch, Filter Methods, Studies in Fuzziness and Soft Computing, 89-117, vol 207, Springer Verlag, 2006.

[16] D.A. Bell and H.Wang, A formalism for relevance and its application in feature subset selection, Machine Learning, 41, 175-195, 2000.

[17] W.H. Press et al, Numerical recipes in C. The art of scientific computing. Cambridge university Press, Cambridge UK, 1988.

[18] D. Ergodmus and J.C. Principe, Lower and upper bounds for misclassification probability based in renyis information. Journal of VLSI Signal Processing Systems, 37 (2-3): 305-317, 2004.

[19] Y. Liu, An Information Content Measure Using Multiple-point Statistics, Geostatics Banff 2004, p.1947, Springer, 2005 (O. Leuangthong and C. V. Deutsch, eds.).

[20] Y. Saeys, I. Inza, and P. Larrañaga, A review of feature selection techniques in bioinformatics, Bioinformatics 2007 23(19): 2507-2517.

[21] A. Dasgupta, P. Drineas, B. Harb, V. Josifovski, and M. W. Mahoney, Feature selection methods for text classification, Proceedings of the 13th ACM SIGKDD

international conference on Knowledge discovery and data mining, ACM, New York, 2007.

[22] M. Pechenizkiy, S. Puuronen, and A. Tsymbal, Feature extraction for classification in the data mining. International Journal "Information Theories & Applications" Vol.10, 2003, p. 271.

[23] C. Shannon, A mathematical theory of communication, The Bell System Technical Journal, 27, 379-423, 623-656, July, October 1948.

[24] R. Tibshirani, Regression shrinkage and selection via Lasso. Journal of the Statistical Society. Series B (Methodological), 58 (1), 267-288, 1996.

[25] P. M. Narendra, K. Fukunaga, A Branch and Bound Algorithm for Feature Subset Selection, IEEE Transactions on Computers, vol 26, issue 9, 1977.

[26] W. Siedlecki and J. Sklansky, On automatic feature selection. International Journal of Pattern Recognition and Artificial Intelligence, 2 (2), 197-220, 1988.

[27] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, Gene selection for cancer classification using support vector machines. Machine Learning, 46, 389-422, 2002.

[28] M. Mejía-Lavalle, E. F. Morales, and G. Arroyo, Two Simple and Effective Feature Selection Methods for Continuous Attributes with Discrete Multi-class, Proceedings of 6th Mexican International Conference on Artificial Intelligence, Vol. 4827, p. 452, Springer, 2007.

[29] J.-B. Jeon, J.-H. Kim, J.-H. Yoon, and K.-S. Hong, Performance Evaluation of Teeth Image Recognition System Based on Difference Image Entropy, vol. 2, pp. 967-972, Proceedings of Third International Conference on Convergence and Hybrid Information Technology, 2008.

[30] Jain, A.K., Duin, R.P.W., Mao, J.: Statistical pattern recognition: A review. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 4–37 (2000).

[31] M. Cebe and C. Gunduz-Demir, Test-Cost Sensitive Classification Based on Conditioned Loss Functions, Machine Learning: ECML 2007 Proceedings of 18th European Conference on Machine Learning, Springer 2007.

[32] Kohavi, R., John, G. H. (1997), Wrappers for Feature Subset Selection, Artificial Intelligence, Volume 97, Issue 1-2, Special issue on relevance, p273 – 324.

[33] A. Bjorck, Solving linear least squares problems by Gram-Schmidt orthogonalization, BIT 7:121 (1967).

[34] E. Tuv, A. Borisov, and K. Torkkola. Feature selection using ensemble based ranking against artificial contrasts. In Proceedings of the International Joint Conference on Neural Networks (IJCNN), 2006.

[35] T. G. Dietterich. Ensemble methods in machine learning. In First International Workshop on Multiple Classifier Systems 2000, Cagliari, Italy, volume 1857 of Lecture Notes in Computer Science, pages 1–15. Springer, 2000b.

[36] Y. Freund, R.E. Schapiro, Experiments with a new boosting algorithms, In Machine Learning: Proceedings of Thirteenth International Conference, 148-156, 1996.

[37] V. Svetnik, T. Wang, C. Tong, A. Liaw, R. P. Sheridan, Q. Song, Boosting: an ensemble learning tool for compound classification and QSAR modeling, Journal of Chemical Information and Modeling, Volume 45, 3, 786-799, 2005.

[38] X. Guorong, C. Peiqi, and W. Minhui, Bhattacharyya distance feature selection. In Proceedings of the Thirtheenth International Conference on Pattern Recognition, vol 2, 195-199, 2002.

[39] S. Raychaudhuri, J. Stuart, R. Altman, Principal components analysis to summarize microarray experiments: application to sporulation time series. Pacific Symposium on Biocomputing 2000, 455-466; O. Alter, P. Brown, D. Botstein, Singular value decomposition for genome- wide expression data processing and modeling. Proceedings of the National Academy of Sciences 97, 10101-10106, 2000.

[40] J. Goeman J, S. van de Geer, de Kort F, H. van Houwelingen, A global test for groups of genes: testing association with a clinical outcome. Bioinformatics, 20, 93-99, 2004; X. Chen, L. Wang, Integrating biological knowledge with gene expression profiles for survival prediction of cancer, Journal of Computational Biology, 16, 265-278, 2009.

[41] B. Schölkopf, C. J. C. Burges, A. J. Smola. Advances in Kernel Methods - Support Vector Learning. MIT Press, Cambridge, MA, 1999

[42] B. Schölkopf, A. J. Smola, K.-R. Müller, Nonlinear components analysis as a kernel eigenvalue problem. Technical report 44, Max Planck Institute for Biological Cybernetics, Tubingen, Germany, 1996

[43] L. Journaux, X. Tizon, I. Foucherot, and P. Gouton, "Dimensionality reduction techniques: an operational comparison on multispectral satellite images using unsupervised clustering," in Proceedings of the 7th Nordic Signal Processing Symposium (NORSIG '06), pp. 242–245, Reykjavik, Iceland, June 2006; M. Lennon, G. Mercier, M. C. Mouchot, and L. Hubert-Moy, "Curvilinear component analysis for nonlinear dimensionality reduction of hyperspectral images," in Image and Signal Processing for Remote Sensing VII, vol. 4541 of Proceedings of SPIE, pp. 157–168, Toulouse, France, September 2002.

[44] D. A. Landgrebe, Signal Theory Methods in Multispectral Remote Sensing, John Wiley & Sons, 2003.

[45] A. Hyvärinen, J. Karhunen, and E. Oja, Independent Component Analysis, John Wiley & Sons, New York, NY, USA, 2001. B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a Kernel eigenvalue problem," Neural Computation, vol. 10, 5, 1299–1319, 1998.

[46] M. Fauvel, J. Chanussot, and J. A. Benediktsson, Kernel principal component analysis for the classification of hyperspectral remote sensing data over urban areas, EURASIP Journal on Advances in Signal Processing, 783194 (January 2009).

[47] I. Borg and P. Groenen, "Modern Multidimensional Scaling: theory and applications" (2nd ed.), Springer-Verlag, New York, 2005.

[48] L. Dyrskjot, T. Thykjaer, M. Kruhhofer, J. L. Jensen, N. Marcussen, S. Hamilton-Dutoit, H. Wolf, and T. F. Orntoft, Identifying distinct classes of bladder carcinoma using microarrays, Nature genetics, 33, 90, 2002.

[49] J. B. Kruskal, and M. Wish, Multidimensional Scaling, Sage University Paper series on Quantitative Application in the Social Sciences, Sage Publications (1978).

[50] J. A. Snyman, Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms. Springer (2005).

[51] M.L Mehta, Random matrices, Academic Press, Boston, 1991.

[52] E.P. Wigner, On a class of analytic functions from the quantum theory of collisions, Ann. Math. **53**, 36, 1951, Proc. Cambridge Philos. Soc. **47**, 790, 1951.

[53] P. Šeba, Parking and the visual perception of space, e-print: http://arxiv.org/pdf/0907.1914v1

[54] M. Krbalek and P. Seba, Headway statistics of public transport in Mexican cities, J. Phys. A: Math. Gen. 36 L7, 2003; M. Krbalek and P.Seba, Statistical properties of the city transport in Cuernavaca (Mexico) and random matrix theory. J. Phys. **214** (2000), 1, 91-100.

[55] P. Seba, Random Matrix Analysis of Human EEG Data, Physical Review Letters, Vol. 91, No. 19. (2003).

[56] Z. Burda, A. Jarosz, J. Jurkiewicz, M. A. Nowak, G. Papp, I. Zahed, Applying free random variables to random matrix analysis of financial data, cond-mat/0603024 (2006).

[57] A. Utsugi, K. Ino, M. Oshikawa, Random Matrix Theory analysis of cross correlations in financial markets, Phys. Rev. E, 70, 026110, 2004.

[58] J. D. Noh, Model for correlations in stock markets, Phys. Rev E, vol61, 5, 2000.

[59] D. Helbing, Traffic and related self-driven many-particle systems, Rev. Mod. Phys. 73, 1067, 2001.

[60] T. Ohira and R. Sawatari, Phase transition in a computer network traffic model, Phys. Rev. E 58, 193, 1998 ; R. V. Sole, S. Valverde, Information transfer and phase transitions in a model of Internet traffic, Physica A, 289, 595, 2001 .

[61] D. Helbing, M. Schreckenberg, Cellular automata simulating experimental properties of traffic flow, Phys. Rev. E 59, R2505, 1999.

[62] Science Vol. 284. No. 5411, 1999.

[63] Hans-Jürgen Stockman, Quantum Chaos: An Introduction, 1999.

[64] G. Frahm, U. Jaekel, Random matrix theory and robust covariance matrix estimation for financial data, arXiv:physics/0503007; L. Laloux, P. Cizeau, J.-P. Bouchaud, M. Potters, Noise Dressing of Financial Correlation Matrices. Physical Review Letters 83, 1467, 1999.

[65] V. Plerou, P. Gopikrishnan, B. Rosenow, L. A. Nunes Amaral, T. Guhr, and H.E. Stanley, Random matrix theory approach to cross correlations in financial data, Phys. Rev. E, vol 65, 066126, 27 June, 2002.

[66] P. J. Forrester, N. C. Snaith and J. J. M. Verbaarschot, J. Phys. A: Math. Gen. 36, 1–10, 2003.

[67] generally a "few-body" interactions dominate in most physical systmes no matter how complex they are: two-body interaction betweens gas molecules, two or theree stocks affecting each other trends, small clusters of genes, mutually impacting their members' expressions.

[68] T.A. Brody, J.Flores, J.B. French, P.A. Mello, A. Pandey, and S.S.M. Wong, Random-matrix physics: spectrum and strength fluctuations, Rev. Mod. Phys. 53, 385 - 479, issue 3, July 1981.

[69] T. Guhr, A. Muller-Groeling, and H.A. Weidenmuller, Random matrix theories in quantum physics: common concepts, Phys. Rep. 299, 190, 1998.

[70] O. Bohigas, M. J. Giannoni, and C. Schmit, Phys. Rev. Lett. 52, 1, 1984.

[71] F. Luo, Y. Yang, J. Zhong , H. Gao, L. Khan, D. K. Thompson and J. Zhou, Constructing gene co-expression networks and predicting functions of unknown genes by random matrix theory, BMC Bioinformatics, 8, 299, 2007.

[72] W.-C. Lau, S.-Q. Li, Traffic analysis in large-scale high-speed integrated networks:validation of nodal decomposition approach, INFOCOM, Proceedings of twelfth annual joint conference of the IEEE Computer and Communications Societies, vol **3**, 1320-1329, 1993.

[73] W.H. Allen, G.A. Marin, L.A. Rivera, Automated detection of malicious reconnaissance to enhance network security, Southeast Conference, Proceedings of IEEE, issue 8-10, 450-454, 2005.

[74] K. Fukuda, Dissertation Thesis: A study on phase transition phenomena in internet traffic, Keio University, 1999.

[75] T. Ohira, R. Sawatari, Phase transition in a computer network traffic model, Phys. Rev. E 58, 193–195, 1998.

[76] M. Barthelemy, B. Gondran and E. Guichard, Large scale cross-correlations in internet traffic, arXiv:cond0mat/0206185, vol 2, 2002.

[77] A. Lakhina, M. Crovella, and C. Diot, Detecting distributed attacks using network-wide flow traffic, Proceedings of FloCon 2005 Analysis Workshop, 2005.

[78] A. Lakhina, M. Crovella, and C. Diot. Mining Anomalies Using Traffic Feature Distributions. Technical Report BUCS-TR-2005-002, Boston University, 2005.

[79] Theory of Financial Risk and Derivative Pricing: From Statistical Physics to Risk Management. Jean-Philippe Bouchaud, Marc Potters, Cambridge University Press; 2-nd edition, 2004.

[80] A.M. Sengupta and P.P. Mitra, Distributions of singular values for some random matrices, arXiv:cond-mat/9709283 vol 1, 25, 1997.

[81] S. Sharifi, M. Crane, A. Shamaie and H. Ruskin, Random matrix portfolio optimization: a stability approach, Physica A 335, 629–643, 2004.

[82] H. Bruus and J.-C. Angles d'Auriac, Energy level statistics of two-dimensional Hubbard model at low filling, arXiv:cond-mat/9610142 vol 1, 18, 1996.

[83] H. Bruus and J.-C. Angles d'Auriac, The spectrum of two-dimensional Hubbard model at low filling, Europhysics letters, 35 (5), 321–326, 1999.

[84] F. Dyson and M.L. Mehta, Statistical theory of the energy levels of complex systems, J. Math. Phys. 4, 701, 713, 1963.

[85] K. Srinivasan, Congestion Control in Computer Networks, EECS Department University of California, Berkeley Technical Report No. UCB/CSD-91-649, 1991.

[86] H. J. Kim, Y. Lee, B. Kahng, and I. Kim, Weighted scale-free network in financial correlations, Journal of the Physical Society of Japan 71 (9), 2133–2136, 2002.

[87] A. Feldmann, A. C. Gilbert, W. Willinger and T. G. Kurtz, The changing nature of network traffic: Scaling phenomena, ACM SIGCOMM Computer Communication Review 28 (2), 5–29, 1998.

[88] A. Feldmann, A. C. Gilbert, P. Huang, and W. Willinger, Dynamics of IP traffic: A study of the role of variability and the impact of control, in Proc. ACM SIGCOMM 99, 301–313, 1999.

[89] M. Crovella and E. Kolaczyk, Graph Wavelets for Spatial Traffic Analysis, in Proceedings of IEEE Infocom 2003,San Francisco, CA, USA, 2003.

[90] R. Mahajan, S. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker, Controlling high bandwidth aggregates in the network, Technical Report, AT&T Center for Internet Research at ICSI, 2001.

[91] P. Barford and D. Plonka, Characteristics of network traffic flow anomalies, in Proceedings of ACM SIGCOMM Internet Measurement Workshop, San Francisco, CA, USA, 2001.

[92] H. Hoffmann, Kernel PCA for Novelty Detection, Pattern Recognition, vol. 40, 863–874, 2007

[93] Y. Jian, K. Mills, Monitoring the macroscopic effect of DDoS flooding attacks, IEEE Transactions on Dependable and Secure Computing, vol. 2, 4, 324–335, 2005.

[94] T.R. Golub et.al.,Molecular classification of cancer: class discovery and class prediction by gene expression monitoring, Science, vol 286, 15 Oct 1999.

[95] M. E. Crovella, A. Bestavros, Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes, IEEE/ACM Transactions on Networking, 5, 6, 835–846, 1997.

[96] W.E. Leland et al., On the self-similar nature of Ethernet traffic, IEEE/ACM Trans. Networking, vol **2**, 1, 1994, 1-15.

[97] A. Erramilli, O. Narayan, and W. Willinger, Experimental queuing analysis with long-range dependent packet traffic, IEEE/ACM Trans. Networking, vol **4**, 2, 1996, 209-223.

[98] K.B.K. Mayya and R.E. Amritkar, Analysis of delay correlation matrices, oai:arXiv.org:cond-mat/0601279 (2006-12-20).

[99] H-P. Kriegel, P. Kröger, A. Zimek, "Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering", ACM Transactions on Knowledge Discovery from Data (ACM) 3 (1), 2009, 1-58.

[100] R. Agrawal, J. Gehrke, D. Gunopulos, P. Raghavan, "Automatic Subspace Clustering of High Dimensional Data", Data Mining and Knowledge Discovery (Springer Netherlands) 11 (1), 2005, 5-33.

[101] C. Ding, X. He, H. Zha, H. D. Simon, "Adaptive dimension reduction for clustering high dimensional data" , Second IEEE International Conference on Data Mining, 2002, 147-155.

[102] K. Y. Yeung, W. L. Ruzzo, "Principal component analysis for clustering gene expression data", Bioinformatics, 17(9), 2001, 763-74.

[103] L. Giada and M. Marsili, "Data clustering and noise undressing of correlation matrices", Phys. Rev. E 63, 061101, 2001.

[104] F. Luo et all, "Constructing gene co-expression networks and predicting functions of unknown genes by random matrix theory", Bioinformatics, 8, 2007.

[105] L. Kullmann, J. Kertesz, R.N. Mantegna, "Identification of clusters of companies in stock indices via Potts super-paramagnetic transitions", Physica A, 287, 2000, 412-419.

[106] M. Halkidi, Y. Batistakis, M. Vazzirgiannis, "On clustering vaildation techniques", Journal of Intelligent Information Systems, 17, 2, 3, 2001, 107-145.

[107] J. Handl, J. Knowles, D. B. Kell, "Computational cluster validation in post-genomic data analysis", Bioinformatics, 21, 15, 2005, 3201-3212.

[108] C. Biely and S. Thurner, Random matrix ensemble of time-lagged correlation matrices: derivation of eigenvalue spectra and analysis of financial time-series, arXiv:physics/0609053 vol **1** 7 Sep 2006.

[109] V. Rojkova, M. Kantardzic, Feature extraction using random matrix theory approach, Sixth International Conference on Machine Learning and Applications, 410–416, 2007.

[110] V. Rojkova, M. Kantardzic, A. Elmaghraby, Y.Khalil, Use of Simulation and Random Matrix Theory to Identify the State of Network Traffic, 2007 IEEE International Symposium on Signal Processing and Information Technology, 647–652, 2007.

[111] Predrag Cvitanovic. http://chaosbook.org/

[112] Albert R and Barabasi A L 2002 Rev. Mod. Phys. 74 47; Dorogovtsev S N and Mendes J F F 2002 Evolution of Networks: From Biological Nets to the Internet and WWW (Oxford University Press); Pastor-Satorras R and Vespignani A 2004 Evolution and Structure of the Internet: A Statistical Physics Approach (Cambridge University Press)

[113] Mandelbrot, B.B. (1977). Fractals: Form, Chance, and Dimension. W.H. Freeman, San Francisco. Mandelbrot, B.B. (1983). The Fractal Geometry of Nature. New York, W.H. Freeman. Mandelbrot, B.B., Passoja, D.E. and Paullay, A.J. (1984). Fractal character of fracture surfaces of metals. Nature 308, 721722.

[114] M. Faloutsos, P. Faloutsos, and C. Faloutsos, Comput. Commun. Rev. 29, 251 (1999); A.-L. Baraba´si, R. Albert, and H. Jeong, Physica (Amsterdam) 281A, 2115 (2000).

[115] A. Erramilli, O. Narayan, and W. Willinger, Experimental queuing analysis with long-range dependent packet traffic, IEEE/ACM Trans. Networking, vol 4, 2, 1996, 209-223. W.E. Leland et al., On the self-similar nature of Ethernet traffic, IEEE/ACM Trans. Networking, vol 2, 1, 1994, 1-15.

[116] Karagiannis, M. Molle, and M. Faloutsos, Long-range dependence, ten years of Internet traffic modeling, IEEE Internet Computing, Oct 2004;

[117] B. B. Mandelbrot. How long is the coast of Britain? Statistical self-similarity and fractional dimension. Science: 156, 636-638 (1967).

[118] D.J. Watts and S.H. Strogatz, Nature (London) 393, 440 (1998); A.-L. Barabási and R. Albert, Science 286, 509 (1999); E. Ravasz, A.L. Somera, D.A. Mongru, Z.N. Oltvai, and A.-L. Barabási, Science 297, 1551 (2002); E. Ravasz and A.-L. Barabási, Phys. Rev. E 67, 026112 (2003).

[119] K.-I. Goh, G. Salvi, B. Kahng, D. Kim Phys.Skeleton and fractal scaling in complex networks Rev. Lett. 96, 018701 (2006); S. H. Strogatz, Nature (London), 433, 365 (2005); C. Song, S. Havlin, H. A. Makse, Nature (London) 433, 392 (2005); cond-mat/0507216.

[120] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, E. Teller, J. Chem. Phys. 21, 1087 (1953); J. Liu, E. Luijten, Phys. Rev. Lett. 92, 035504 (2004).

[121] Quantum Monte Carlo simulations of solids Reviews of Modern Physics, Vol. 73, Issue: 1, January 01, 2001. pp. 33-83 Foulkes , W. M. C. ; Mitas , L. ; Needs, R. J. ; Rajagopal , G.

[122] Collective effects in cellular structure formation mediated by compliant environments: a Monte Carlo study Authors: I. B. Bischofs, U. S. Schwarz (Heidelberg University) Comments: 45 pages, 7 postscript figures included, revised version accepted for publication in Acta Biomaterialia Journal-ref: Acta Biomaterialia 2: 253-265 (2006)

[123] D. Sornette, Critical Market Crashes (UCLA and CNRS-Univ. Nice) Comments: Latex 89 pages and 38 figures, in press in Physics Reports Journal-ref: Physics Reports 378 (1), 1-98 (2003)

[124] F. Dyson, Statistical theory of the energy levels of complex systems, J. Math. Phys. **3**, 140 (1962).

[125] A.W. Lo and A.C MacKinlay, When are contrarian profits due to stock market over-reaction?, Review of Financial Studies, **3**, 175-206 (1990).

[126] T. Chordia and B. Swaminathan, Trading volume and cross-autocorrelations in stock returns, Journal of Finance, **55**, 913-936 (2000).

[127] M. Krunz, On the limitations of the variance-time test for inference of long-range dependence, IEEE INFOCOM, 2001, 1254-1260.

[128] S. Molnar and T.D. Dang, Pitfalls in long range dependence testing and estimation. GLOBECOM, 2000.

[129] J. Kwapien, S. Drozdz, and A.A. Ioannides, Temporal correlations versus noise in the correlation matrix formalism: an example of the brain auditory response, arXiv:cond-mat/0002175, vol **1**, 11 Feb 2000.

[130] W.-X. Du, C. H. Thurber, and D. Eberhart-Phillips, Earthquake relocation using cross-correlation time delay estimates verified with the bispectrum method, Bulletin of the Seismological Society of America, June 2004, vol **94**, 3, 856-866.

[131] I. M. Johnstone, High Dimensional Statistical Inference and Random Matrices, arXiv.org:math/0611589, 19 Nov 2006.

[132] A. Tulino and S. Verdu, Random matrix theory and wireless communications, Communications and Information theory, vol **1**, issue **1**, June 2004, 1 - 182.

[133] P. Ormerod snd C. Mounfield, Random matrix theory and the failure of macro-economic forecasts, Physica A: Statistical Mechanics and its Applications vol **280**, Issues 3-4, 1 June 2000, pp 497-504.

[134] J. McNutt and M. De Shon, Correlation between quiescent ports in network flows, CERT network situational awareness group report, Carnegie Mellon University, September 2005.

[135] A. Lakhina, M. Crovella, and C. Diot, Characterization of network-wide anomalies in traffic flows, Proceedings of the ACM/SIGCOMM Internet Measurement conference, 2004, 201-206.

[136] L. Min, Y. Shun-Zheng, A network-wide traffic anomaly detection method based on HSMM, Int. conf. on communications, circuits and system proceedings, vol **6**, June 2006, 1636 - 1640.

[137] M. Roughan, T. Griffin, M. Mao, A. Greenberg, and B. Freeman, Combining routing and traffic data for detection of IP forwarding anomalies, Proceedings of the joint int. conf. on Measurement and modeling of computer systems, 2004, 416 - 417.

[138] L. Huang, X. Nguyen, M. Garofalakis, M. Jordan, A. Joseph and N. Taft, Distributed PCA and network anomaly detection, Technical report No. UCB/EECS-2006-99.

[139] J. Boutros and G. Caire, Iterative multiuser joint decoding: unified framework and asymptotic analysis, IEEE Trans. on Information Theory, **vol** 48, 7, pp. 1772-1793, July 2002.

[140] M. L. Honig and R. Ratasuk, Large-system performance of iterative multiuser decision-feedback detection, IEEE Trans. on Communications, **vol** 51, 8, pp. 1368-1377, Aug. 2003.

[141] G. L. Trigg, Physical Review Letters **42**, 747-748 (1979).

[142] [9] Marchenko, V.A. and Pastur, L.A. (1967) The distribution of eigenvalues in certain sets of random matrices. Mat. Sb., 72, 507-536.

CURRICULUM VITAE

NAME: Viktoria Rojkova

ADDRESS: Department of Computer Science
University of Louisville
Louisville, KY 40292

EDUCATION: M.S. Psychology
University of Illinois at Urbana-Champaign 2004
M.S. Computer Science
University of Louisville 2005

RESEARCH: Electrophysiology
Cognitive Psychology
Formal Languages
Complex Networks

TEACHING: Statistics
Software Engineering and Data Structures

AWARDS: SAS 2007 Student Poster Award
CAINE 2007 Best Paper Award
KDD CUP 2009 Second place