U

UCINET

Stephen P. Borgatti[1], Martin G. Everett[2], and Linton C. Freeman[3]
[1]LINKS Center for Social Network Analysis, Gatton College of Business and Economics, University of Kentucky, Lexington, KY, USA
[2]Mitchell Centre for Social Network Analysis, School of Social Science, University of Manchester, Manchester, UK
[3]Department of Sociology, School of Social Sciences, University of California, Irvine, CA, USA

Synonyms

Graph theory; NETDRAW; Social network analysis

Glossary

Social capital. The potential benefit of having certain ties or occupying a certain position in the network.

Definition

Tool's ID card:

- Title: UCINET for Windows: Software Package for Social Network Analysis
- Creation date: 1992 (current incarnation, first appeared in 1983)
- Authors: Borgatti SP, Everett MG, Freeman LC
- Specialization: General social network analysis
- Copyright: Commercial
- Type: Program
- Size limits: approximately 10,000 nodes
- Platforms: Windows
- Programming language: Delphi
- Orientation: social science data analysis

Introduction

UCINET is a general package for social network analysis. It is mostly used in the social sciences to analyze sociometric survey data. The program features a large number of metrics that can be used to characterize whole networks and positions of nodes within networks. It also features a number of analytical techniques such as finding cohesive subgroups (clustering), blockmodeling, and multivariate statistical analysis.

Key Points

UCINET is strong in providing metrics that can be used to characterize nodes and whole networks. Although it contains a number of advanced features, it is designed to be usable by researchers who are not technically oriented.

R. Alhajj, J. Rokne (eds.), *Encyclopedia of Social Network Analysis and Mining*,
DOI 10.1007/978-1-4614-6170-8,

	HOLLY	BRAZEY	CAROL	PAM	PAT	JENNIE	PAULINE
HOLLY	0	2	15	8	4	12	10
BRAZEY	1	0	12	2	10	11	5
CAROL	17	15	0	1	2	4	6
PAM	9	5	6	0	3	4	1
PAT	4	10	8	3	0	1	2
JENNIE	11	9	4	2	1	0	7
PAULINE	14	10	5	1	3	4	0

UCINET, Fig. 1 Full matrix data format

Historical Background

UCINET was initially created by Linton C. Freeman in 1983; he gathered together social network analysis programs written by different people for the DOS operating system and put them together into a single distribution library (at that time, a set of 5.25″ floppy disks). This first version was written in a variety of different languages including Fortran, Basic, and Algol, but by using compiled versions, Freeman provided a package which made a variety of network analysis methods available to a larger community. For version 2 of the program, Freeman designed a common front end for all the programs so that all would accept data in a single format and so extended the reach of the program. For version 3, Bruce MacEvoy rewrote the separate programs in the Basic language. At the same time, Stephen P. Borgatti wrote an independent suite of programs in Pascal called NetPac. For version 4 of UCINET, Freeman decided it did not make sense to have competing programs, and so in 1992 the next incarnation of NetPac was renamed UCINET IV and the Basic version of UCINET was discarded. Together with Martin G. Everett, Borgatti took responsibility for the software. Some years later, with the help of students from the University of Greenwich (UK), UCINET was converted to Windows, creating UCINET version 5, which also incorporated a context-sensitive help system. In 2002, version 6 was released. It is continually being updated. At this writing, the current version is 6.459.

Features and Capabilities

Input and Output

UCINET accepts a large number of data and file formats. The usual method of data entry is to cut-and-paste the contents of an Excel file into one of UCINET's data editors, in particular the one known as the DL Editor. The DL Editor accepts data in a variety of formats and can also represent data in a variety of formats. The latter capability makes it easy to export data from UCINET by cutting and pasting from the DL Editor into programs like Excel. We give four examples of data formats here.

Full matrix. In this format, the data are arranged as a node-by-node adjacency matrix in which xij indicates the presence or strength of a tie from node i to node j. This format is mainly used for small, dense datasets, as it is inefficient for data containing a large number of non-ties (Fig. 1).

Nodelist. In this format, each row of data consists of a list of nodes that a given node is connected to. This is a highly compact format suitable for large datasets in which the presence of ties but not the strength of ties is recorded (Fig. 2).

Edgelist. In this format, each row of a data is a triple consisting of the sending node, the receiving node, and the strength of the tie between them (representing such things as the subjective liking of one node for the other, or the frequency of contact, or the duration of the relationship) (Fig. 3).

bill	tom	steve	dan
tom	steve		
dan	steve	tom	

UCINET, Fig. 2 Nodelist format. The first column gives the "sending" node. The remaining columns indicate who the focal node has ties to

ROMUALD	PETER	3
ROMUALD	VICTOR	1
ROMUALD	HUGH	2
BONAVENTURE	PETER	3
BONAVENTURE	LOUIS	2
BONAVENTURE	AMAND	1
AMBROSE	BONAVENTURE	2
AMBROSE	VICTOR	3
AMBROSE	ALBERT	1
BERTHOLD	AMBROSE	2
BERTHOLD	PETER	3
BERTHOLD	JOHN-BOSCO	1

UCINET, Fig. 3 Edgelist format. The first column gives the sending node, the second column gives the receiving node, and the third column gives the weight of the tie

Edgelist23. Whereas in the edgelist format it is assumed that all of the ties are all of the same type, in the edgelist23 format different kinds of ties may be specified. Hence each row of data consists of a quadruple giving the sending node, the receiving node, the type of tie, and the strength of tie (which is 1 for binary data) (Fig. 4).

Other Formats and Data Editing

UCINET is able to import a variety of other data formats which are used in other network programs. These include the formats used in Pajek, Negopy, and Krackplot as well as the VNA format used by the drawing package NETDRAW that is associated with UCINET. In addition it can read a raw data matrix that is in plain text using a variety of field separators. Once imported a UCINET data file can be simply edited using the UCINET spreadsheet editor. In addition the software provides a host of data management routines that allow for data to be manipulated and restructured once it has been imported.

Techniques

UCINET incorporates a large number of different techniques for analyzing network data.

Multivariate statistics. UCINET provides factor analysis, correspondence analysis, cluster analysis, and multidimensional scaling. In addition, it offers lower-level procedures such as singular value decomposition and extraction of eigenvectors of symmetric and nonsymmetric matrices.

Cohesive subgroups. An important element of network analysis is the detection of cohesive subgroups. UCINET offers a wide variety of techniques for detecting groups, including cliques, n-cliques, n-clans, k-plexes, lambda sets, factions, and many more.

Equivalence. Another important element of network analysis is the detection of structurally similar classes of nodes. UCINET offers the ability to detect structurally equivalent nodes, automorphically equivalent nodes, and regularly equivalent nodes.

Hypothesis testing. UCINET provides a number of routines that use permutation tests to test hypotheses about networks. These include node level or monadic hypotheses, such as the hypothesis that more central people tend to be happier, and dyadic hypotheses, such as the hypothesis that coworkers who socialize together collaborate more effectively. We also can have mixed monadic and dyadic hypotheses and test if people who are of a similar age are more likely to be friends.

Metrics

UCINET is capable of computing a wide variety of metrics for dyads, nodes, and groups or whole networks. It is not possible to give a comprehensive list so we describe general areas and give examples.

Cohesion. The extent to which actors in the network are intertwined with each other.

UCINET, Fig. 4 Edgelist23 format. The first two columns give the sending and receiving nodes, the third column gives the type of tie, and the fourth column gives the strength of tie

Node1	Node2	Relation	Weight
ACCIAIUOL	MEDICI	Marriage	1
ALBIZZI	GINORI	Marriage	1
ALBIZZI	GUADAGNI	Marriage	1
ALBIZZI	MEDICI	Marriage	1
BARBADORI	CASTELLAN	Marriage	1
BARBADORI	CASTELLAN	Business	1
BARBADORI	GINORI	Business	1
BARBADORI	MEDICI	Marriage	1
BARBADORI	MEDICI	Business	1
BARBADORI	PERUZZI	Business	1
BISCHERI	GUADAGNI	Marriage	1

Measures include simple density and average degree through to more sophisticated ideas such as fragmentation, transitivity, and homophily. Fragmentation measures the proportion of pairs of actors that are not mutually reachable, whereas homophily takes into account actors' attributes and examines the extent to which actors connect to actors with the same or different attributes.

Centrality. The most widely used metric in social network analysis and perhaps as a consequence the largest array of techniques. UCINET has all the major centrality measures often with additional options. For example, the closeness routine implements the standard sum of geodesic distances but also allows the user to choose sums of reciprocal distances, sum of lengths of all paths, or the sum of the length of all the trails. It also allows the user to select how to deal with unreachable actors.

Core/periphery. Core-periphery methods sit between equivalence and centrality. The discrete core-periphery method assigns actors to a position based upon an overall metric, whereas the continuous method gives each actor core-periphery values with the interpretation that actors with a high score are in the core and those with a low score are in the periphery.

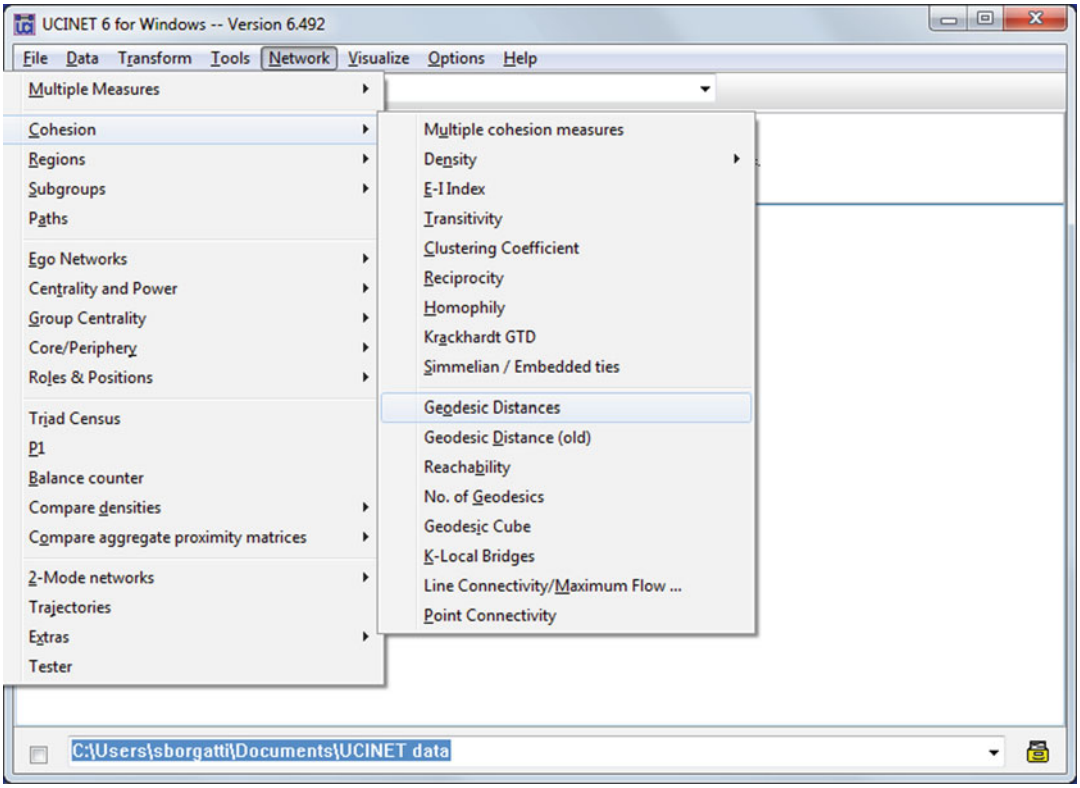
Ego-metrics. UCINET can take whole networks and then analyze them as if they were a collection of ego networks. The ego network for each actor in the data is extracted and then analyzed separately. Typically we look at both the composition of the ego network (e.g., percentage of women and political allegiance) and the structure (e.g., density and number of components) including Burt's measures of structural holes. A collection of ego networks can be submitted as a whole network, but in this case it is only the egos who will have interpretable metrics.

Visualization. UCINET is installed with a companion program called NETDRAW which is used to visualize networks. By default, the program uses a spring-embedding type of graph layout algorithm to locate points in space. The program is also capable of using principal components and metric multidimensional scaling as alternative layout algorithms. In addition, the user may lay out nodes based on node-level variables, such as centrality or status.

NETDRAW allows the user to map node and tie characteristics to drawing features such as color, size, and shape of symbols used to represent nodes and lines. For example, nodes can be colored to represent genders, and lines can have thickness proportional to strength of tie.

The program also makes it easy for users to see subsets of the network, defined both in terms of nodes (e.g., just women) and ties (e.g., only friendship ties).

NETDRAW also includes a basic set of analysis routines, such as clustering and measuring centrality.



UCINET, Fig. 5 UCINET menu

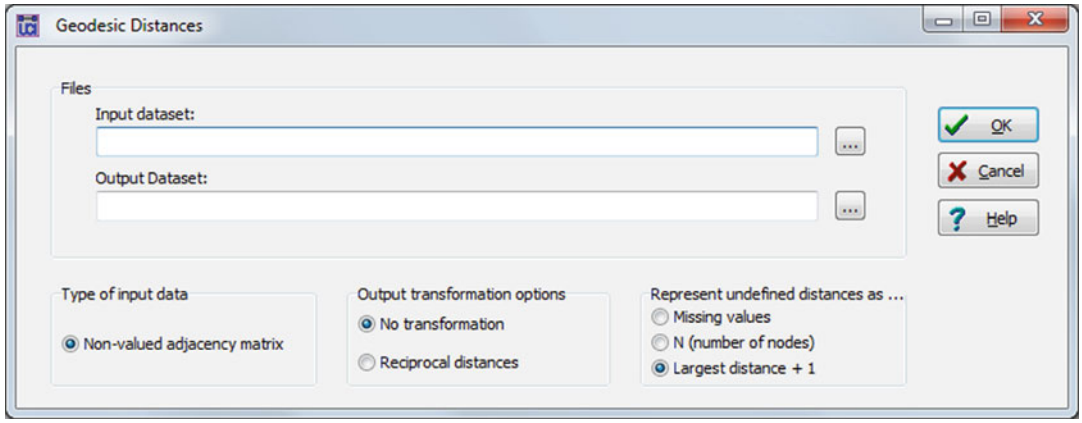
Network diagrams produced by NETDRAW can be saved in a variety of formats, including Windows metafile and jpg. In addition, the program can print directly to a printer, using the full resolution of the printer. In conjunction with programs like Adobe Acrobat, it is also possible to print the diagrams to high resolution pdf files.

Usage

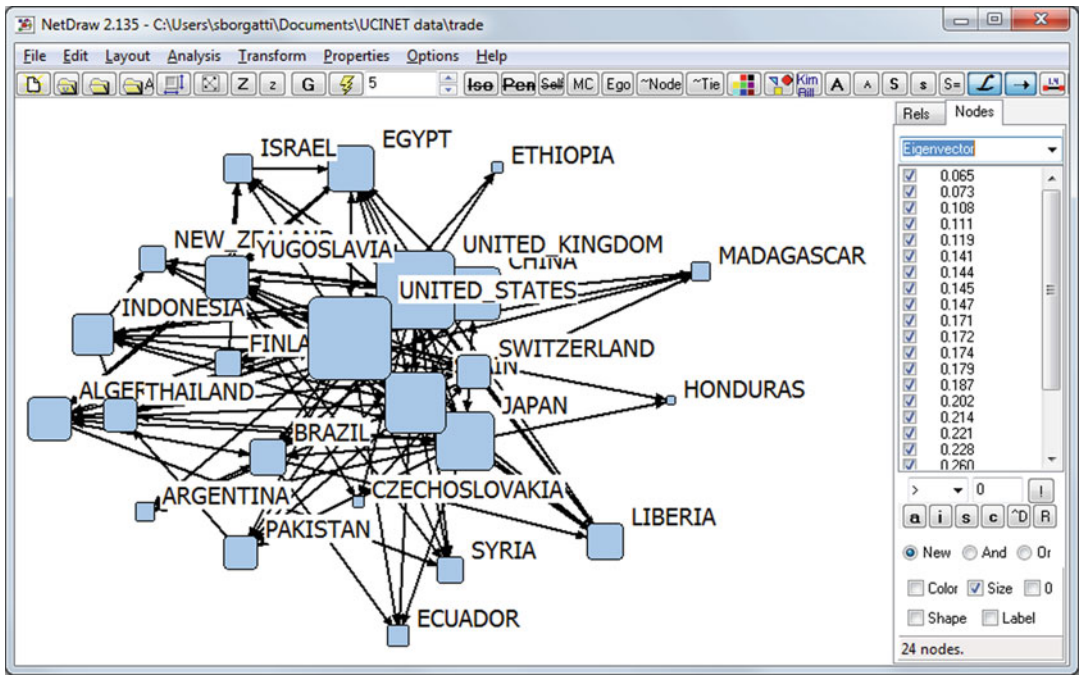
When a UCINET routine is run, there are usually two types of output. Firstly there is a text file that gives the results of any analysis. This is known as the output log and is, by default, displayed using Windows Notepad. In any given session, all the output files are kept and it is possible to look back and review any analysis in the same session. Once UCINET is closed, these files are deleted and so the user must actively save any text output required. In addition UCINET usually creates a UCINET data file that also contains the results. These files are not deleted at the end of the session (but they may be overwritten if a routine is run more than once) and are available to the user. These can be reviewed at a later date, exported by viewing them in a data editor and then say pasting the results into excel, or used as an input into a further analysis. Some of the routines also produce some simple graphical outputs such as a scatterplot from multidimensional scaling or a dendrogram from hierarchical clustering, and these will also be deleted at the end of a session and should be saved if they are required for later use.

Examples

Figure 5 shows the UCINET interface and the selection process needed to find the geodesic distances between every pair of nodes.



UCINET, Fig. 6 UCINET dialogue box



UCINET, Fig. 7 NETDRAW screenshot

Methods are arranged under general headings intended to aid finding a particular routine.

Running the above causes a dialogue box to open as shown below in which the user provides information such as the source of the data and any parameters required (Fig. 6). The related software package NETDRAW can be launched from UCINET and is interactive; the screenshot in Fig. 7 gives a flavor of the software.

Documentation

The program comes with a help system, which is also provided as a pdf file. In addition, there is a "quick start" guide that is available online at http://www.analytictech.com/ucinet/documentation/quickstart.pdf

Users of UCINET can obtain and offer help via an online forum. The address is: http://tech.groups.yahoo.com/group/ucinet/

There is a third-party tutorial offered free online by Robert Hanneman and Mark Riddle that can be found here: http://faculty.ucr.edu/~hanneman/

Finally, a book entitled *Analyzing Social Networks* by Borgatti et al. (2013) is also available, published by Sage Publications. This book discusses how to design and execute social network research, drawing extensively from UCINET. In addition the book has a website (https://sites.google.com/site/analyzingsocialnetworks) that gives detailed descriptions of how the results described in the book are produced using the software.

Key Applications

UCINET is used primarily in the social sciences, notably the sociology and management fields, although usage is growing in the life sciences and especially health sciences.

Most of this usage is in support of fundamental research. Two basic types of network research can be identified as prototypical. The first type is the study of social capital. Social capital studies investigate how an entity's social connections affect the opportunities and constraints they face. For example, an entrepreneur's connections can be used to access resources such as expertise, sources of funding, and material aid. The second type of research is the study of diffusion and influence. For example, consumers adopt new products in part because their friends have adopted them.

Future Directions

UCINET is constantly evolving. It is like a journal in that it provides a way for authors to have their newest metrics and methods published and made useful to others. With respect to the architecture of the system, it is extremely likely that alternative methods of accessing its capabilities will be developed, such as command line interfaces and automated processing.

Acknowledgments

The UCINET team acknowledges the help of hundreds of individuals who have reported bugs, suggested improvements, submitted algorithms, and so on.

Cross-References

- ▸ Centrality Measures
- ▸ Clustering Algorithms
- ▸ Correspondence Analysis
- ▸ Distance and Similarity Measures
- ▸ Eigenvalues, Singular Value Decomposition
- ▸ Matrix Algebra, Basics of
- ▸ Network Data File Formats
- ▸ Pajek
- ▸ Principal Component Analysis
- ▸ Regression Analysis
- ▸ Similarity Metrics on Social Networks
- ▸ Social Capital
- ▸ Social Influence Analysis
- ▸ Social Network Datasets
- ▸ Spectral Analysis
- ▸ Structural Holes
- ▸ Univariate Descriptive Statistics

References

Borgatti SP, Everett MG, Johnson JC (2013) Analyzing social networks. Sage, London

http://faculty.ucr.edu/~hanneman/

http://tech.groups.yahoo.com/group/ucinet/

http://www.analytictech.com/ucinet/documentation/quickstart.pdf

UI Mash-Ups

- ▸ Web Mash-ups

Uncertain Data

- ▸ Outlier Detection with Uncertain Data Using Graphics Processors

Uncertainty

▸ Probabilistic Analysis

Unicode

Ethan V. Munson
Department of EECS, University of Wisconsin-Milwaukee, Milwaukee, WI, USA

Synonyms

ISO/IEC 10646: the universal character set

Glossary

Character An abstraction that represents a symbol used to construct words in a written language

Code Point A number used to uniquely identify a character

Encoding A mapping from characters to code points

Glyph An abstract symbol that can be rendered at multiple sizes and resolutions in order to represent a character visually to people

Definition

Unicode (The Unicode Consortium 2013) is an industry standard for representing text characters that is designed to facilitate internationalization and localization of software systems. The Unicode effort was initiated because previous attempts at defining a universal text encoding had important limitations that limited their scope or made their encoding complicated. Unicode supports the scripts of almost all languages in active use plus some important historical languages and even imaginary scripts invented for scholarly and artistic reasons. Unicode has a flexible design that is capable of supporting all known human languages and all of their variant scripts. The standard is a critical element in the internationalization of software systems. The development of the Unicode standard is coordinated by the Unicode Consortium.

Historical Background

Prior to the creation of Unicode, a variety of text encoding standards had been defined. ASCII and EBCDIC defined standards that were sufficient to cover English text and punctuation. The ISO-8859 family of encodings expanded ASCII to support a wide range of phonetic writing systems but principally for European languages. There were also language-specific standards for some Asian languages. At the time, this might have been sufficient, because the computing industry tended to have national champions and many operating systems were focused on national or regional markets.

The rapid adoption of the personal computer, and especially of graphical user interfaces, changed this environment because a small number of operating systems were serving people from many different cultures. This change forced computer and operating system manufacturers to consider how to easily make their systems work well for very different writing systems. Standardization had also become more widely accepted and was obviously a key part of the solution to internationalizing and localizing systems.

The development of Unicode began in the late 1980s and was coincident with a separate ISO/IEC standardization effort that had started in 1984. The ISO/IEC effort produced a formal Draft Proposal 10646 in 1989 while Unicode 1.0 was published in 1990. The two groups saw that cooperation was superior to competition and decided to merge the two standards in 1991. A total of 15 revisions of Unicode have been released to date, with the last being version 6.2, which was released in September 2012 (The Unicode Consortium 2012). Most revisions have been quite substantial and added multiple scripts, though version 6.2 added only the single character for the Turkish lira symbol. Version 6.2 covers 100 languages and contains 110,182 characters.

Design Principles

Graham (2000) describes the main design principles underlying Unicode. These principles are not adhered to in all cases and sometimes conflict, but they still guide the overall structure of the standard. The most important are:

Full encoding: Unicode is intended to be sufficiently large to encode all written scripts with any substantial use. Each Unicode character has a unique code value, in contrast to earlier standards that defined multiple character sets with overlapping coding schemes.

Characters, not glyphs: Unicode encodes abstract characters, rather than the particular glyphs that a computer presents on the screen or page.

Logical order: Characters are to be stored in the order that a literate user would read them, rather than the order they appear on the screen. This is particularly important when a document mixes scripts with different writing directions.

Unification: Similar characters from different languages are often encoded as a single character. A simple example is the Western punctuation characters. More controversial was the reduction of over 100,000 Chinese, Japanese, and Korean characters into about 28,000 Han character codes. This substantially reduced the size of the standard but complicates rendering in some specific cases.

Convertibility: Unicode is designed to support round-trip conversion with preexisting character coding standards.

Code Points, Encodings, and Properties

Each Unicode character has a unique *code point* which is its numeric value and will be in the range 0_{hex} to $10FFFF_{hex}$. The standard originally anticipated a full 31-bit code space, but UTF-16 supported only a smaller space of 1,114,112 code points, and this was adopted for the full standard. Code points are written in a convention with "U+" followed by a hexadecimal number of four to six digits, depending on the value.

An encoding is a mapping of code points into actual sequences of bits that will be stored or transmitted. The Unicode standard and the ISO/IEC 10646 standard have included a number of different encodings over their life spans. The currently supported encodings are UTF-8, UTF-16, and UTF-32, where UTF stands for the *Unicode Transformation Format* mapping method and the number stands for the minimum number of bits that can be used to represent a character.

- UTF-8 is an 8-bit variable-width encoding that uses between 1 and 4 bytes per character. It is directly compatible with ASCII and represents Latin characters efficiently. It was originally designed to support a full 31-bit character space with up to 6 bytes but was limited to 4 bytes for compatibility with UTF-16.
- UTF-16 is a 16-bit variable-width encoding that uses one or two 16-bit words per character. It is more space efficient for characters with high code numbers (such as Han unification characters) but adds some complexity because of the need to deal with byte-order issues within the 16-bit values. The prevalence of spaces, Latin punctuation, and English words in even non-English text can reduce its practical space efficiency. Finally, it is not compatible with ASCII and has many 0 bytes that prevent the use of null-byte termination for UTF-16 strings.
- UTF-32 is a 32-bit fixed-width encoding. Except for byte-order issues, it is logically simpler than UTF-8 and UTF-16 but is always less space efficient.

In practice, UTF-8 and UTF-16 are the most widely used encodings. UTF-8 is extensively used on the World Wide Web, while UTF-16 is widely supported by software development tools such as the Java programming language.

In addition to a code point, characters have other properties that can be useful to software developers. These include the following:

- A category classification such as Letter, Number, or Punctuation
- A descriptive character name that is guaranteed to be unique

- A writing order marker specifying left-to-right or right-to-left
- For characters that can be constructed by combining other characters (as with the accented Latin characters), a decomposition into simpler characters
- For characters that represent numbers, a numeric value

Cross-References

▸ HTML
▸ International Hyperlink Networks
▸ XSLT

References

Graham T (2000) Unicode: a primer. M&T Books, Foster City

The Unicode Consortium (2012) The Unicode standard, Version 6.2.0. Mountain View. ISBN 978-1-936213-07-8

The Unicode Consortium (2013) The Unicode standard (latest version). Mountain View. http://www.unicode.org/versions/latest. Accessed 3 May 2013

Univariate Descriptive Statistics

Ayona Chatterjee[1] and Bruce E. Trumbo[2]
[1]Department of Statistics and Biostatistics, California State University, Hayward, CA, USA
[2]Department of Statistics and Biostatistics, California State University, East Bay, Hayward, CA, USA

Glossary

Here we give brief definitions of several commonly used sample statistics and graphical displays. Later we introduce additional sample statistics, some of which are related to those we show here.

Bar Chart For categorical data, an array of vertical bars, of equal width and slightly separated, one for each class. Extending upward from a common baseline, the height of each bar is proportional to the number of observations in the class it represents. Typically, class names or designations are shown beneath the baseline

Boxplot A graphical display of the distribution of a numerical sample based on its ***five-number summary.*** A *box* represents the middle half of the data, and lines called *whiskers* extend on each side of the box towards the maximum and minimum, respectively. Often the whiskers of a boxplot are embellished to show individual values of observations considered to be **outliers**

Categorical Data A number k of *classes* (also called *categories*, *groups,* or *levels*) is chosen to partition the envisioned possibilities. Each observation is associated with exactly one class. Examples of categorical variables are religious preference, gender, and agreement with a statement (strongly agree, agree, neutral, disagree, strongly disagree). *Ordinal* categories have a natural order (as for agreement with a statement), *nominal* categories do not (as for religious preference). *Binary* nominal data have two levels (such as male/female, yes/no)

Coefficient of Variation The sample ***standard deviation*** divided by the sample ***mean***, usually multiplied by 100 and expressed as a percent. The coefficient of variation (CV) is used only for nonnegative numerical data. It is a pure number, having no units

Continuous Numerical Data In theory, observations can take any value on the real line or in a particular subinterval thereof. Examples are lengths of objects, times to run a race, and temperatures. In practice, values must be rounded to some number of decimal places. Optimally, there are very many possible rounded values, and ties are nonexistent or relatively few

Discrete Numerical Data Possible values of observations have gaps between them. The most usual case is integer counts of events: mouse clicks during a visit to a website, number of withdrawals posted to a checking account in a given month, and so on

Dotplot A graphical display of the distribution of a numerical sample, in which each individual

observation is represented by a dot placed above a horizontal number line (somewhat similar to a *stripchart*)

Five-Number Summary From smallest to largest: the sample minimum, lower quartile Q_1, median, upper quartile Q_3, and maximum

Histogram A graphical display of the distribution of a numerical sample. An interval including the minimum and maximum observations is partitioned into *bins* (also called *classes* or *subintervals*), and the number of observations in each bin is determined. Bars above each bin have areas proportional to bin frequencies

Interquartile Range The interval (Q_1, Q_3) between the lower and upper quartiles of a sample contains (about) half of its observations. The interquartile range (often abbreviated *IQR*) is the length $Q_3 - Q_1$ of this interval

Mean Let $x_1, x_2, \ldots, x_n$ be values in a sample of size n. Then the sample mean (sometimes called the *average* or *arithmetic mean*) is

$$\bar{x} = \frac{x_1 + x_2 + \cdots + x_n}{n} = \frac{\sum_{i=1}^{n} x_i}{n}.$$

In some applications with positive data, the *harmonic mean* $\left(\frac{1}{n}\sum_{i=1}^{n} x_i^{-1}\right)^{-1}$ and *geometric mean* $\left(\prod_{i=1}^{n} x_i\right)^{1/n} = \exp\left(\frac{1}{n}\sum_{i=1}^{n} \ln x_i\right)$ are also used. For example, the mean of the $n = 5$ observations $2, 5, 8, 7$, and 4 is $\bar{x} = 26/5 = 5.2$, the harmonic mean is $5/1.21786 = 4.1056$, and the geometric mean is $2{,}240^{0.2} = e^{1.54285} = 4.6779$

Median Suppose that the observations $x_1, x_2, \ldots, x_n$ in a sample are arranged from smallest to largest: $x_{(1)}, x_{(2)}, \ldots, x_{(n)}$, where the parentheses in the subscripts indicate the sorting. These $x_{(i)}$ are called the *order statistics* of the sample. The smallest order statistic $x_{(1)}$ is the *minimum* value of the sample, and the largest $x_{(n)}$ is its *maximum* value. If there are tied values, then some adjacent sorted values will be equal. If the sample size n is odd, then the sample median, which we denote $\tilde{x}$, is the middle value $x_{(\frac{n+1}{2})}$ in the sorted list. If n is even, then the sample mean is usually taken to be the average of the two middle observations: $(x_{(\frac{n}{2})} + x_{(\frac{n}{2}+1)})/2$. Thus, the median of the five sorted observations $2, 4, 5, 6$, and 8 is $x_{(3)} = 5$, and the median of the six sorted observations $1, 2, 4, 5, 7, 8$ is 4.5. The median can also be called the middle ***quartile*** or the 50th ***percentile***

Mode If a numerical sample has a most frequently occurring value, then that value is called the sample mode. Similarly, if data are categorical, we say the class with the most observations is the *modal class*. For example, in a random sample of residents of Italy, the modal religious preference would likely be Roman Catholic

Outlier A numerical observation that is judged, according to some criterion, to be markedly "separated" from the "main body" of values in a sample. ***Boxplots*** are often used to identify outliers

Percentile If n observations in a sample are sorted from smallest to largest, the kth percentile ($0 \le k \le 100$) is a number below which not more than $nk/100$ of the data lie and above which not more than $n(1-k)/100$ of the data lie. The kth percentile need not be equal to any sample value, and various textbooks and software have different ways of resolving ambiguities. Fortunately, percentiles are most profitably used with large datasets, for which slight differences among rules for resolving ambiguities are inconsequential. The kth percentile is also is called *quantile* $k/100$. For example, the median is the 50th percentile and 0.5 quantile

Pie Chart A circular chart with one sector representing each class of a ***categorical variable***. The central angle of the sector is proportional to the frequency of the class it represents. The angles add to 360° so that the entire circle is used. Each sector is labeled with a class name or designation

Quartiles The lower quartile Q_1 is the 25th percentile, and the upper quartile Q_3 is the 75th percentile. Although the ***median*** is the middle quartile (50th percentile), the notation

Q_2 is not often used. Roughly speaking, the lower quartile, median, and upper quartile are three values that divide the sorted data into four groups of approximately equal size. For example, Q_1 is approximately the median of the values smaller than the median of the sample. Less commonly, the terms *decile* (for tenths) and *quintile* (for fifths) are also in use, so one could refer to the median as the fifth decile

Range The range R of a sample is defined as the sample maximum minus the sample minimum. In the notation of the definition of the sample ***median,*** this is $x_{(n)} - x_{(1)}$. Thus, the noun *range* is defined as a number, not an interval

Standard Deviation The standard deviation of a sample is the positive square root s of its ***variance*** s^2. If several samples, perhaps labeled with xs and ys, respectively, are used in the same discussion, then subscripts are used to avoid confusion: s_x and s_y

Skewness Lack of symmetry about the mean. Various sample statistics are in use to quantify skewness. The US National Institute for Standards and Technology (NIST) defines sample skewness as

$$\frac{\sum_{i=1}^{n}(x_i - \bar{x})^3}{(n-1)s^3},$$

where s is the sample ***standard deviation*** (NIST/SEMATECH 2012). Several statistical software packages use the Fisher-Pearson skewness statistic, which multiplies the NIST statistic by $\frac{n}{n-2}$. For large samples, the Fisher-Pearson adjustment is negligible. By either measure, a perfectly symmetrical sample would have 0 skewness

Stripchart A graphical display of the distribution of a numerical sample, in which each individual observation is indicated by symbol placed along a number line. Tied observations may be overprinted to appear as one. In R, various options are available to prevent overprinting (somewhat similar to a ***dotplot***)

Variance If a sample of size n has mean $\bar{x}$, then the deviations from the mean are the n quantities $(x_i - \bar{x})$, which sum to 0. Roughly speaking, the variance of this sample is an average of its squared deviations $(x_i - \bar{x})^2$. More precisely, the sample ordinarily variance is defined as

$$s^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}.$$

As the notation indicates, this is the square of the sample ***standard deviation.*** There are several technical and historical reasons why $n - 1$ is ordinarily used in the denominator instead of n. For example, in a random sample from a population with mean μ and variance σ^2, consider the sample variance S^2 as a random variable. Then, $\mathsf{E}(S^2) = \sigma^2$, so that S^2 is an unbiased estimator of σ^2

Introduction

Numerical and graphical descriptive statistical methods are used to summarize a collection of data so that its essential features can be more easily understood and visualized than is possible by looking at a listing of the data. Although descriptive methods do not include formal inference such as testing hypotheses or making confidence intervals, this does not detract from the importance of descriptive methods. Here are some situations in which descriptive statistical methods may be of practical use.

- The data are a complete census of the population with respect to the particular variable of interest. In this case inference is not relevant because the population distribution is fully described by the data. Nevertheless, descriptive methods may assist in understanding that population distribution.
- The data consist of a very small sample from a population. Although there may not be enough data for meaningful inference, it seems worthwhile to get a clear view of the limited information at hand.
- A very large amount of data results from mining or some other algorithmic process. Even if no formal probability model is envisioned,

descriptive methods can help determine whether the data are of practical use.

- A sample of moderate size is chosen at random from a population with an assumed shape (e.g., normal or exponential). As a prelude to formal inference involving population parameters, we seek a preliminary view of the data – perhaps to assess its quality, to help decide what kind of inference is appropriate, or to anticipate the inferential results and how to communicate them to a nontechnical audience. In extreme instances, one may even conclude the data are not of sufficient quality to warrant inferential analysis.

In each case, it is important to realize that the effective use of descriptive methods requires an understanding of the reasons for collecting the data and the meaning of the measurements used. For example, if data are annual incomes of employees of a corporation, is the purpose to get an idea whether a "typical" employee earns enough to be above an officially determined poverty level or is the purpose to decide whether anticipated profits from the next year are sufficient to cover total salary costs? For the first purpose we might want to look at the median or modal salary; for the second, we might want to look at the mean salary, which is based on the total.

As another example, if data measure airflow of asthma patients breathing air slightly polluted with SO_2, it would be important to know whether the lowest airflow values reflect minor discomfort or life-threatening distress and whether the highest ones are consistent with those of people without asthma or still indicate serious disease. Are outliers noise to be disregarded or the main signal?

Before the computer age much of the advice about descriptive methods dealt with which few numerical measures or graphical displays were most worthwhile in view of the time and effort required for computing and graphing. Nowadays, fast hardware and convenient software make it possible, tempting, or even perhaps obligatory to look at a very wide variety of numerical and graphical summaries. However, while descriptive methods may help understand the data, it is important to realize that they do not create new information beyond what is already in the data. In the end, one must decide which few of the many available procedures are the most useful for understanding, accurately reporting, and clearly communicating the essence of the data at hand.

In the section "Key Points" we discuss important properties of some commonly used descriptive statistics and graphical displays. Then in the section "Illustrative Examples" we give examples of specific uses of these (and a few other) descriptive methods.

Key Points

In this section we discuss properties and uses of a few of the most important descriptive statistics for numerical and categorical data, followed by brief comments on graphical methods: stripcharts, histograms, bar charts, and pie charts.

Measures of centrality and location for numerical data The mean, median, and mode are measures of *centrality* for numerical data. Of the three, the mean is most sensitive to outliers. For the 11 numbers from 0 through 10, the mean and median are both 5. By contrast, the 11 numbers 0 through 9 and 1,000 still have median 5, but their mean is 95. Depending on the circumstances, the sensitivity of the mean to outliers may be an advantage or a disadvantage.

Even for discrete data, the mode may not exist: 2 is the mode of the sample $1, 2, 4, 2, 5$, but the sample $1, 2, 4, 3, 5$ has no mode. In theory, continuous data have no ties and hence no mode; thus, "modes" in continuous data are often artifacts of rounding.

In addition to the measures of centrality, the quantiles are also called measures of *location*. Of these, the maximum, minimum, and upper and lower quartiles are perhaps the most often used.

All of the measures of location discussed here behave predictably when data are transformed values by a constant shift a. Let $x_1, \ldots, x_n$ be a sample, and define $y_i = a + x_i$, for $i = 1, \ldots, n$.

Then the relationship between the sample means and medians is $\bar{y} = a + \bar{x}$ and $\tilde{y} = a + \tilde{x}$, respectively. Also, for a general linear transformation, if $y_i = a + bx_i$, then $\bar{y} = a + b\bar{x}$. Similar relationships hold for the median and mode. If $b > 0$, similar relationships hold for the quantiles. However, if $b < 0$ the sequence of the order statistics is reversed; for example, the transform of the upper quartile of the x_i becomes the lower quartile of the y_i.

Measures of dispersion and shape for numerical data The most common measures of dispersion or variation are the standard deviation s, variance s^2, coefficient of variation (CV), range, and interquartile range (IQR). Formulas and definitions are given in section "Glossary." All but two of these measures of dispersion have the same units as the data. Exceptions are the variance, which has squared units, and the coefficient of variation, which has no units.

The only measure of shape we consider here is skewness, defined in section "Glossary." It has no units and is sensitive to outliers. A nearly symmetrical sample has approximately 0 skewness. A positively skewed (also called right-skewed) sample tends to have a tail (of trailing values) extending to the right of the median and a negatively skewed sample a tail to the left.

All of these measures of dispersion and shape, except the coefficient of variation, are invariant under a shift transformation, because of the subtractions involved in their definitions. If the standard deviation of the ages of five friends is $s = 6.2$ years today, and they have a reunion 5 years from today, their average age will increase by 5 years, but the standard deviation of their ages will still be 6.2 years at the reunion.

Under the linear transformation $y_i = a + bx_i$, we have $s_y = |b|s_x$, and similarly for the range and IQR. For the variance, $s_y^2 = b^2 s_x$. The coefficient of variation is invariant under a change of scale $y_i = bx_i$, for $b > 0$. The CV of the weights of the five friends is the same whether measured in pounds, kilograms, or stones. Also, the CV for weights of a sample of ants might be greater than for a sample of elephants.

Example 1 (Nonlinear transformations) Monotone nonlinear transformations can have unexpected effects on the sample mean, variance, and skewness. In the USA fuel efficiency of vehicles using gasoline is measured in miles per gallon (MPG); in many other countries fuel efficiency is measured in liters per 100 km. The conversion from observations x_i in MPG to the metric observations y_i is $y_i = 235.22/x_i$, which is a reciprocal transformation, not a linear one.

Suppose a fleet of 235 light trucks owned by a company has MPG values with descriptive statistics shown in the first column of Table 1 below and metric equivalents in the second column. For the means, we have $235.22/\bar{x} = 15.35 \neq \bar{y} = 16.78$. Also, for standard deviations we have $235.22/s_x = 57.09 \neq s_y = 5.18$. Because the mean and the median of the MPG measurements are approximately the same and skewness is near 0, we conclude that these data are approximately symmetric. However, the metric version of the data has a distinctly positive skewness, and a mean pulled noticeably to the right of the median by a long right-hand tail. See panels (a) and (b) of Fig. 1.

It seems reasonable that the metric measurements might be right-skewed. Stopped, waiting for a red light to change, a truck is momentarily getting 0 MPG, while the metric measure is infinite. If all of the trucks are driven about the same distance and we are interested mainly in the total amount of fuel used, then it would be best to use the harmonic mean of the MPG measurements and the arithmetic mean of the metric measurements.

By contrast with the mean, the median of the transformed values is the transformed value of the median. Even though the reciprocal transformation reverses the direction of the order statistics, the middle value on the MPG scale corresponds directly to the middle value on the metric scale: $235.22/\tilde{y} = 15.27 = \tilde{x}$.

The empirical rule In a normal population with mean μ and standard deviation σ, the probability that lies within $\mu \pm \sigma$ is 0.6827, the

Univariate Descriptive Statistics, Table 1 Fuel efficiencies of trucks

Statistic	MPG	Metric
Mean	15.32	16.78
Median	15.40	15.27
SD	4.12	5.18
Skewness	−0.05	1.92

probability within $\mu \pm 2\sigma$ is 0.9545, and the probability within $\mu \pm 3\sigma$ is 0.9774. Experience has shown that symmetrical samples with moderate tails on both sides (sometimes described *mound-shaped*) tend to have roughly these properties in common with a normal population. Specifically, the empirical rule for the data states that

(i) About 68 % of the n observations in a sample lie in the interval $\bar{x} \pm s$.
(ii) About 95 % of the observation lie in the interval $\bar{x} \pm 2s$.
(iii) All or almost all of the observations lie in the interval $\bar{x} \pm 3s$.

Parts (ii) and (iii) are often the most reliable, sometimes giving remarkably close results even for some asymmetrical samples.

One use of the empirical rule is to build intuition about sample standard deviations. For a typical auditorium full of people, it would clearly be a bad "guess" to say that the average height of the adults in the crowd is 220 cm (7′ 3″). By contrast, a guess that the standard deviation of their heights is 30 cm (11.8″) might not be immediately recognized as foolish. However, from the empirical rule one can judge that a span as large as ±60 cm would hardly be required in order to include 95 % of the adults and that a span as large as ±90 cm seems truly excessive to include "almost all." (Depending on the country and the nature of the crowd, reasonable guesses might be $\bar{x} \approx 175$ cm and $s \approx 7$ cm.)

Example 2 (Illustrating the empirical rule for two standard deviations) To explore how part (ii) of the empirical rule works in practice, we simulated 10,000 samples of size 50 from a normal population. Across these 10,000 samples, the average number out of $n = 50$ observations falling within $\bar{x} \pm 2s$ was 47.9 (95.8 %). For $n = 50$, the attainable percentages nearest 95 % are 92, 94, 96, and 98 %, and together these outcomes accounted for 9,534 of the 10,000 cases (95.34 %) (See Fig. 1c).

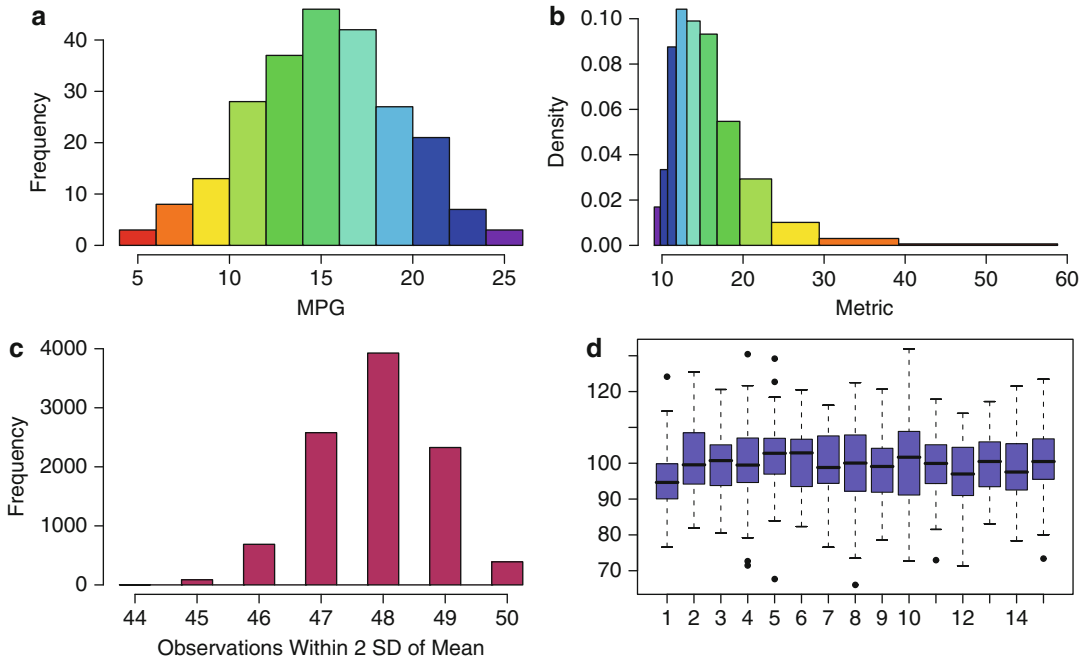
Numerical descriptive measures for categorical data Suppose a categorical variable has n observations tallied into k classes. Then the fundamental summary is a list of the *frequencies* f_j, for $j = 1, \ldots, k$. The k *relative frequencies* are defined as $r_j = f_j / \sum_{j=1}^{k} f_j = f_j / n$.

The modal category d, if it exists, has the largest frequency: $f_d > f_j$, for $j \neq d$. The classes of an ordinal categorical variable have a natural order from $j = 1$ to k. The median class, if it exists, is such that the total frequency of lower classes and the total frequency of upper classes are both less than $n/2$. Here is an example of a categorical variable with no median class: $n = 12$, $k = 5$, with $f_1 + f_2 = 6$ and $f_3 + f_4 + f_5 = 6$.

If a binary categorical variable has classes 0 = no and 1 = yes then the mean of the 0s and 1s is the fraction of yes answers in the sample. Otherwise, it does not automatically make sense to speak of the mean for categorical data because the class labels cannot be added. For example, if the classes are A = strongly opposed, B = opposed, C = neutral, D = favor, and E = strongly favor, then it is not necessarily clear what is meant by A + C.

Nevertheless, classes are sometimes given digital labels (such as $1, 2, \ldots, 5$), and then these labels are treated as numbers. For example, grade point averages at most US universities are computed by assigning artificial numerical values to essentially ordinal categories A, B, and so on. (Before the computer age, the sorting required to find a median grade would have been prohibitive.) Similarly, Likert scales, widely used in psychometrics, seek to treat such digital class names of an ordinal variable as a meaningful numerical scale.

In summary, here is a diagram of data types, in which the measures of centrality that naturally apply to each are marked with ⋆. (The symbol + indicates that additional arguments are required to justify the use of the mean.)



Univariate Descriptive Statistics, Fig. 1 Graphs for section "Key Points." Example 1: Histograms (**a**) and (**b**) show MPG data for 235 trucks and the positively skewed metric equivalents; bars of the same color represent the same trucks. Example 2: Histogram (**c**) illustrates the empirical rule. A large majority of 10,000 simulated normal samples of size 50 have nearly 95 % of their observations within $\bar{x} \pm 2s$. Example 3: Each of the boxplots in panel (**d**) summarizes a sample of size 50 from a normal population; 6 of the 15 show outliers

	Mode	Median	Mean
Nominal	*		
Ordinal	*	*	+
Numerical	*	*	*

Graphical methods for numerical data We discuss stripcharts, dotplots, boxplots, and histograms here. A few other graphical representations of data are shown in the examples of section "Illustrative Examples."

Stripcharts and dotplots In the simplest cases, stripcharts and dotplots put one symbol for each observation along a number line (almost always horizontal for dotplots). Treatments are slightly different from one statistical software system to another.

The main distinctions between stripcharts and boxplots lie in the treatment of values that are tied or lie very close together. For example, Minitab dotplots stack dots for tied values vertically and nearly equal values are treated as equal to avoid overlapping and undue congestion. In R, stripcharts can use a variety of symbols, and narrower symbols (such as |) are less likely to overlap than wider ones. The default is for ties to appear as a single symbol, but several options are available to separate coincident symbols.

At their best, both kinds of graphs show the number of observations and (very nearly) the position of each. Neither is entirely satisfactory for large datasets.

Boxplots Boxplots may be plotted on either a vertical or horizontal scale. A box spans the interval between the lower quartile Q_1 and upper quartile Q_3 (with a mark indicating the location of the median), and two whiskers extend from the minimum to Q_1 and from Q_3 to the maximum (whiskers can be missing in case quartiles happen to match the maximum or minimum values) (Frigge et al. 1989).

In most software packages the default style of boxplot embellishes the whiskers to show symbols at the exact positions of outliers. Two *fences* are located at $Q_1 - 1.5\,\text{IQR}$ and $Q_3 + 1.5\,\text{IQR}$. Points between the lower fence and the minimum and between the upper fence and the maximum are declared as outliers and plotted individually.

Boxplots should be avoided for very small samples (certainly those of size below a dozen) because the quartiles carry little useful information about such samples. Ordinarily, boxplots give no clue as to the sample size, so it is necessary to know the sample size from other sources in order to assess the amount of real information in a boxplot. One remedy is to plot a stripchart parallel to the boxplot.

Example 3 (Boxplot outliers in normal data) Numerous or extreme outliers in a boxplot sometimes arouse suspicion that the sample plotted did not come from a normal distribution. However, boxplots of normal data often show outliers – increasingly often with larger samples (Hoaglin et al. 1986). Boxplots were developed for use in exploratory analysis and can be very effective when used as intended, but there are more reliable ways to test a sample for normality.

As an example, we counted outliers in 10,000 boxplots of simulated samples of size $n = 50$ from a normal population. Among the 10,000 samples, about 36 % showed at least one outlier by the 1.5 IQR criterion discussed above. Among the samples with boxplot outliers, the average number of outliers was about 1.6. Fifteen of the 10,000 boxplots are shown in Fig. 1d.

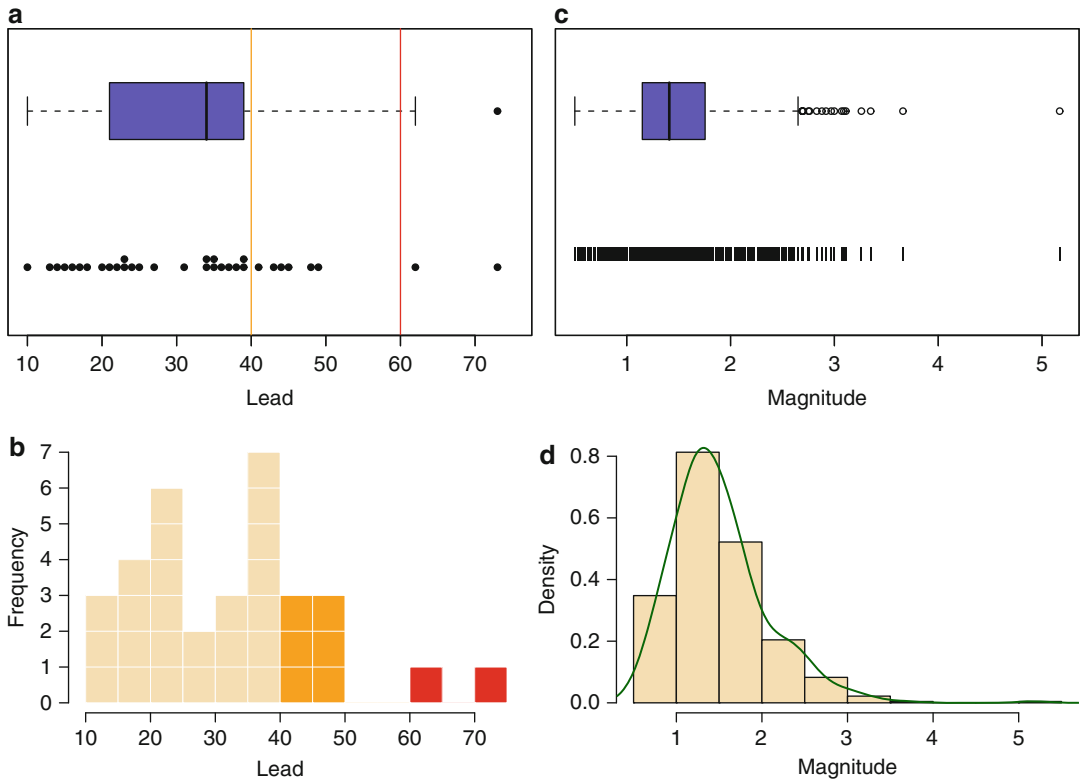
Histograms Histograms are based on putting numerical data into k adjacent bins (or subintervals) that cover the interval from minimum to maximum. Typically, bins are chosen so that their endpoints or midpoints are convenient "round" numbers. There are various principles and rules for determining the approximate number of bins in a histogram, which we discuss briefly in Example 3. In practice, it may be best to try several values of k to see which gives the most informative (or most pleasing) result.

In the simplest case, the bins are of equal length. Then to make a *frequency histogram*, a rectangular bar is drawn for each bin j ($j = 1, \ldots, k$). The base along the horizontal scale matches the jth bin, and the height is the frequency f_j of observations in that bin. The effect is that each of the n observations is represented by the same area under the histogram (see Fig. 2b). In principle, one can determine the sample size from a frequency histogram by adding the heights of its k bars.

Moreover, if you regard the bars as having weight, the balance point of a histogram along the horizontal axis is (approximately) the sample mean. Some information is lost when data are grouped into classes, but the approximate value of the mean is $\bar{x} \approx \frac{1}{n}\sum_{j=1}^{k} m_j f_j$, where m_j are the bin midpoints. This approximation would be exact if every data point fell precisely at a bin midpoint.

Alternatively, the vertical scale of a histogram can be made so that the total area of all histogram bars is unity and each of the n observation is represented by area $1/n$. This is called a *density histogram*, which can be taken as a rough approximation to the density function of the population distribution. If bin lengths are unequal, the density scale must be used, with bar heights adjusted so that each observation is represented by $1/n$ of the total area. There is no way to discern the sample size from a density histogram. Examples of density histograms are shown in Figs. 2d and 3.

Graphical methods for categorical data A bar chart for categorical data has k bars of heights f_j (or r_j) for $j = 1, \ldots, k$, extending vertically from a horizontal base line. Typically, there is some space between the bars, and class labels are put below the base line beneath each bar. (Sometimes a bar chart is rotated 90° clockwise so that the bars extend horizontally.) In many cases, it is best to use bar charts for ordinal categorical data, with the bars in order from left to right, thus emphasizing the ordinal character of the classes. If used for nominal data, it may be useful to arrange bars left to right in decreasing order of frequency.



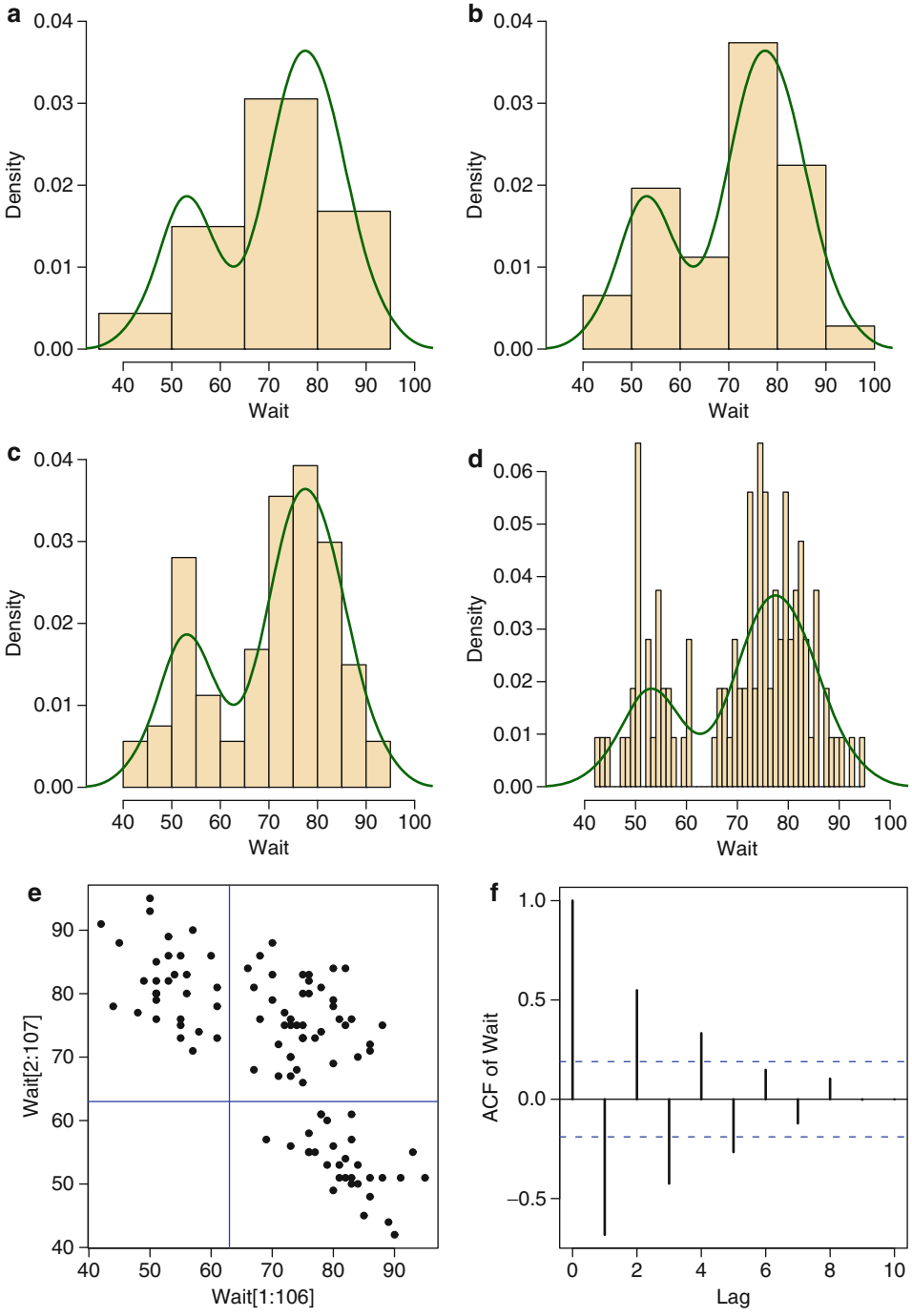
Univariate Descriptive Statistics, Fig. 2 Example 1: Panels (**a**) and (**b**) show blood lead levels of 33 children. The important information is that about a quarter of the children have level above 40 μ g/dl and so have serious lead poisoning. Example 2: Panels (**c**) and (**d**) show magnitudes of 260 California earthquakes in the period from August 29 through September 9, 2000. The only one of these quakes that did damage was the most extreme outlier at magnitude 5.17

A pie chart is a circle with k sectors, each having a respective central angle of $360r_i$ degrees. Pie charts are often used for nominal categorical data because there is no particular order in which the eye tends to focus on the sectors. Classes are often arranged by decreasing order of frequency, either clockwise from the top (12 o'clock) or counterclockwise from the right (3 o'clock), depending on software. However, if one of the categories is a miscellaneous category ("other"), then it should ordinarily be put last in order. Moreover, if a pie chart is used for ordinal data, then the order is preserved; perhaps colors or density of shading or hatching can further assist the viewer to maintain the order. Finally, if several pie charts using the same classes are shown in sequence (as for data from different populations or times), then the same order of assigning classes to sectors is used for all of the pie charts. If there is room, class labels may be put within the sectors, otherwise just outside each sector or in a legend.

Illustrative Examples

In this section we illustrate descriptive statistics and graphics with real datasets. Many of the methods described earlier are illustrated in applied settings, and a few additional methods are introduced.

Example 1 (Lead absorption in children) Blood lead levels (in μg/dl) were measured for all 33 children whose fathers work at a small factory where lead is used to make batteries (Morton et al. 1982).



Univariate Descriptive Statistics, Fig. 3 Example 3: Histogram (**a**) has too few bars to show bimodality of the data. Histograms with Freedman-Diaconis (**b**) and Sturgis (**c**) binning give satisfactory displays of the data. Histogram (**d**) has too many bins and hence a ragged appearance. The same density estimator is overlaid on all four histograms. In (**e**), each of the waiting times 2 through 107 (*vertical axis*) is plotted against the previous one, revealing no instances of two consecutive short waits. The ACF plot (**f**) shows substantial autocorrelation through lag 5 and thus that the data are not an independent sample

One question of interest is whether children absorb lead inadvertently brought home on the bodies or clothing of the workers.

Descriptive statistics are shown in Table 3. Of urgent concern are $Q_3 = 40$ and the maximum value 73. Toxicologists seem in agreement that children with levels above 40 need medical treatment and those with levels above 60 μ g/dl should be hospitalized for immediate treatment (Miller and Kean 1987). By these criteria, about a quarter of the children in this population have serious cases of lead poisoning. The other descriptive statistics may be of some technical interest, but the upper quartile and the maximum tell the important part of story of these data.

Panels (a) and (b) of Fig. 2 show a boxplot, a stripchart, a histogram of these data. The boxplot outlier 73 lies above the upper fence at $Q_3 + 1.5(IQR) = 39 + 1.5(18) = 66$. The upper whisker ends at the value 62, which corresponds to serious lead poisoning, but this observation happens to lie just below the fence and so is not declared an outlier by the boxplot. The particular style we chose for the stripchart in panel (a) makes it hardly distinguishable from a dotplot. A comparison of the stripchart and the histogram shows how individual data are put into histogram bins.

Bars of the histogram are shown as stacks of "bricks," each unit area representing one of the $n = 33$ subjects. Also, we have colored the bars for the bins towards the right to emphasize the degrees of the lead poisoning.

Any one of these three graphs would have been sufficient to show that there are some cases above 40 μ g/dl. Here, the small sample size helps show how the histogram is made up of equal units of area for each observation, but for most purposes, samples of this size are better illustrated by a boxplot or a stripchart.

To finish this story, we mention that the study included a control group of 33 children paired by age and neighborhood of residence with each "exposed" child of a factory worker. Lead values for the control group are nearly symmetrical, with mean and maximum levels about 15 and 25 μ g/dl, respectively (thus, roughly consistent with the cluster of dots for exposed children at the left end of the stripchart and with the lower hump of their histogram). Also, results of a questionnaire revealed that the most serious cases of lead poisoning in the exposed group are associated with fathers who had the most frequent contact with lead on the job and were least attentive to requirements to shower and change clothes before leaving work. This example of transport of lead from factory to home to children of workers provides a cautionary tale for establishing standards of exposure and sanitation in factories where hazardous substances are used. Nevertheless, formal statistical inference from the data on lead levels of the 33 exposed children is not relevant, because these data are complete for the factory in the study.

Example 2 (California earthquakes) Early in the morning of September 3, 2000, an earthquake of magnitude 5.17 occurred in the Napa Valley wine-growing region of California. The quake injured about two dozen people, several of them seriously; did about $50 million in property damage; and spilled many gallons of expensive wine. The data for this example are the magnitudes of the 260 earthquakes recorded in California and parts of northern Nevada during the 2-week period centered on the time of the Napa Valley quake (U.S. Geological Survey Menlo Park CA Berkeley Seismological Laboratory University of California Berkeley Berkeley CA).

Earthquake magnitudes are measured on the Richter scale, a logarithmic scale of earthquake energy that keeps the numbers for the largest quakes within reasonable bounds. Even so, the distribution of magnitudes of quakes in this 2-week period are distinctly skewed to the right. Some numerical indications of skewness are shown in the first column of Table 2. The Fisher-Pearson skewness statistic is 1.35, distinctly larger than 0, and the sample mean is larger than the sample median.

The boxplot in Fig. 2c shows 17 outliers, the smallest of which, at magnitude 2.69, is barely above the upper fence at 2.65. With as many as 260 observations, it is difficult to make a

Univariate Descriptive Statistics, Table 2 Blood lead levels

n	33	Min	10
Mean	31.85	Q_1	21
SD	14.41	Median	34
Skewness	0.78	Q_3	**39**
		Max	**73**

Univariate Descriptive Statistics, Table 3 California earthquakes

	All 260 earthquakes	Omitting Napa Valley
n	260	259
Min	0.50	0.50
Q_1	1.15	1.15
Median	1.41	1.41
Mean	1.50	1.50
Q_3	1.75	1.75
Max	**5.17**	**3.66**
SD	**0.56**	**0.54**
Skewness	**1.35**	**0.91**

stripchart in which all the observations can be identified individually, but the extreme values are clearly identifiable.

At the other end of the distribution, the smallest recorded magnitude is 0.50. Usually, 0.50 is the smallest value recorded for earthquake data – not because there are no smaller seismic events but because it is difficult to detect them at a distance or to distinguish them from mining explosions, construction noise, and so on. Looking at the numerical descriptive statistics, we have to remember we are looking only at seismic events that have been *detected and recorded.* With this somewhat arbitrary minimum cutoff point, it is unclear what inferential value the descriptive statistics may have for a theoretical population of seismic events.

If the big quake is deleted, then the largest remaining one has magnitude 3.66, hardly big enough to be noticed by people who are not very near the epicenter and unlikely to do damage even locally. The distribution of the magnitudes of the remaining quakes is still markedly skewed. Other than the new maximum, the only changes in the descriptive statistics (expressed to two places) are a slight decrease in the standard deviation and a marked decrease in the skewness statistic, which remains well above 0.

In practical terms, the difference between data summaries for the 260 and the 259 observations is that the latter is missing the only earthquake of interest to the general public. For measurements in many emerging fields, pushing the limits of technology, *the signal may be mainly in the outliers,* while the more common values may be mostly or entirely noise.

The histogram in Fig. 2d is plotted on the density scale, so that the sum of the areas of all of its bars is unity. The rightmost bar, including only the Napa Valley quake, is barely visible. The modal bin is $(1, 1.5]$. We have superimposed a *density estimator* on the histogram, which takes its maximum at a *smoothed mode* of 1.31. For right-skewed data, the mode, median, and mean tend to be in increasing order: here, $1.31 < 1.41 < 1.50$.

> There are many kinds of density estimators. Here we used the default in R, but with a bandwidth 1.5 times the default. See R documentation of the procedure `density` and its references, especially (Silverman 1986). Another way to smooth this particular histogram is to fit a lognormal density function (Purcaur and Zorilescu 1971); with or without the Napa Valley quake, the log-magnitudes (a log-log-energy scale) are consistent with a random normal sample.

Example 3 (Geyser eruptions) During the period August 1–8, 1978, a ranger at Yellowstone National Park recorded 107 waiting times between eruptions of Old Faithful geyser, along with the durations of each eruption (both variables in minutes), hoping to learn how to predict times until the next eruption for the benefit of tourists (Weisberg 1985).

> It turned out that the minutes to wait for the next eruption could be predicted, within about plus or minus 15, as 35.8 plus 10.74 times the duration of the previous one. Since 1978 the channels and volume of underground water flow have changed many times, and several websites provide current information.

Univariate Descriptive Statistics, Table 4 Waits between eruptions

n	107	Min	42
Mean	71.00	Q_1	58
SD	12.97	Median	75
Skewness	−0.50	Q_3	81
		Max	95

Here we look only at the 107 waiting times between eruptions. This has become a famous dataset in statistics, partly because the data are *bimodal*. That is, a *properly drawn* histogram has two distinct humps. Numerical descriptive statistics provided in Table 4 do not show bimodality – and neither does a boxplot (omitted here) – because it is based on the five-number summary in the right-hand column of the table.

One purpose in this example is to illustrate the effect of various choices of binning in histograms. For convenience, histogram bins are often chosen to have endpoints or midpoints at "round" numbers. Most statistical analysis programs use a default rule for the approximate number of bins, compromise somewhat in favor of round endpoints, and allow the user to vary the binning if the default result is unsatisfactory. Figure 3 shows four histograms of the waits for eruptions with numbers k of bins ranging from so few that the histogram obscures the bimodal nature of the data to so many that the uneven heights of its bars is distracting.

We consider two popular rules for the number k of bins to use for n observations. The *Sturgis rule* suggests $k \approx 1 + 3.32 \log n = 1 + \log_2 n$ (Sturges 1926). The *Freedman-Diaconis rule* suggests bin length $2n^{-1/3}$ IQR and so $k \approx n^{1/3} R/2$ IQR, where R is the sample range (Freedman and Diaconis 1981). For $n = 107$ the Sturgis rule calls for $k = 16$ or 17, but R reduces this to 11 to give bin endpoints that are multiples of 5. Alternatively, in R one can specify the rule of Freedman and Diaconis, which suggests a bin width of 9 or 10, and then R uses 10 to give $k = 6$. For very large n the Freedman-Diaconis rule, developed with optimal density estimation in mind, tends to give the larger number of bins as $n^{1/3}$ overwhelms $\log n$. For comparison of the four histograms in Fig. 3, the default density estimator of R is overlaid on each of them.

Another purpose of this example is to introduce three additional exploratory methods that may suggest whether data are random. They all require that data are available in the time order of collection. A frequent departure from randomness is that neighboring observations are associated, rather than independent. The Old Faithful waiting times are clearly not independent, as seen in Fig. 3e, where observation x_{i+1} is plotted against x_i. In particular, arbitrarily using 63 min to separate short waits from long ones, we never see two short waits in succession.

The *correlation coefficient* r of samples $x_1, \ldots, x_n$ and $y_i, \ldots, y_n$ is defined as

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{(n-1)s_x s_y}.$$

For a single sequence, the *autocorrelation of lag* ℓ is the correlation of the subsequences $x_1, \ldots, x_{n-\ell}$ and $x_{\ell+1}, \ldots x_n$. The *autocorrelation function* (ACF) of a sample is a list of autocorrelations of various lags. If data are independent, then autocorrelations of all lags should be negligible. Instead, the ACF plot in Fig. 3f shows that the successive waits for eruptions have substantial autocorrelations for lags up through five. The autocorrelation for lag 1 quantifies the linear component of the association shown in Fig. 3e.

ACF plots are widely used in physical, social, and administrative sciences as a descriptive tool for judging whether data are independent. If the goal is to do certain kinds of statistical inference, independence is a requirement. But for other purposes, it is often possible to exploit association to discover patterns and make predictions.

To end this example, we show how the degree of association can be described in terms of the number of runs. A *run* is a sequence of (one or more) identical values. If we denote short waits by **S** and long ones by **L**, then there are $a = 31$ **S**s and $b = 76$ **L**s. Among the **S**s and **L**s,

there are 63 runs. However, if observations were independent, one would expect only $\mu \approx 45$ runs with a standard deviation $\sigma \approx 4.2$.

In the spirit of the Empirical Rule, notice that the discrepancy 18 between the actual number 63 and the expected number 45 of runs is much greater than $3\sigma = 12.6$. Roughly speaking, a discrepancy exceeding 2σ can be taken as a contradiction of randomness. For independent data, the theoretical number of runs has $\mu = 2ab/n + 1$ and $\sigma = [(\mu - 1)(\mu - 2)/(n - 1)]^{1/2}$.

In the geyser data, runs of class **S** eruptions are all of unit length, and this substantially increases the number of runs. The information in runs has been widely used in many fields, including linguistics, data compression, quality management, finance, and athletics.

Example 4 (Heights of students) In the 1940s, anthropologists made careful measurements of the heights of 41 young men attending a boarding school in Bengal, India (Majundar and Rao 1956). Suppose our first attempt summarizing these data is the boxplot on the left in Fig. 4a. From what we know about the heights of young men, we recognize the outlier at 254 cm (about $8'\,4''$) as a very unlikely value, discover that it is a data entry error, determine the correct value to be 154 cm (almost $5'\,1''$), and make the second boxplot in that panel. Now there are no outliers: the correction lowers the lower fence just enough that the minimum is no longer designated an outlier.

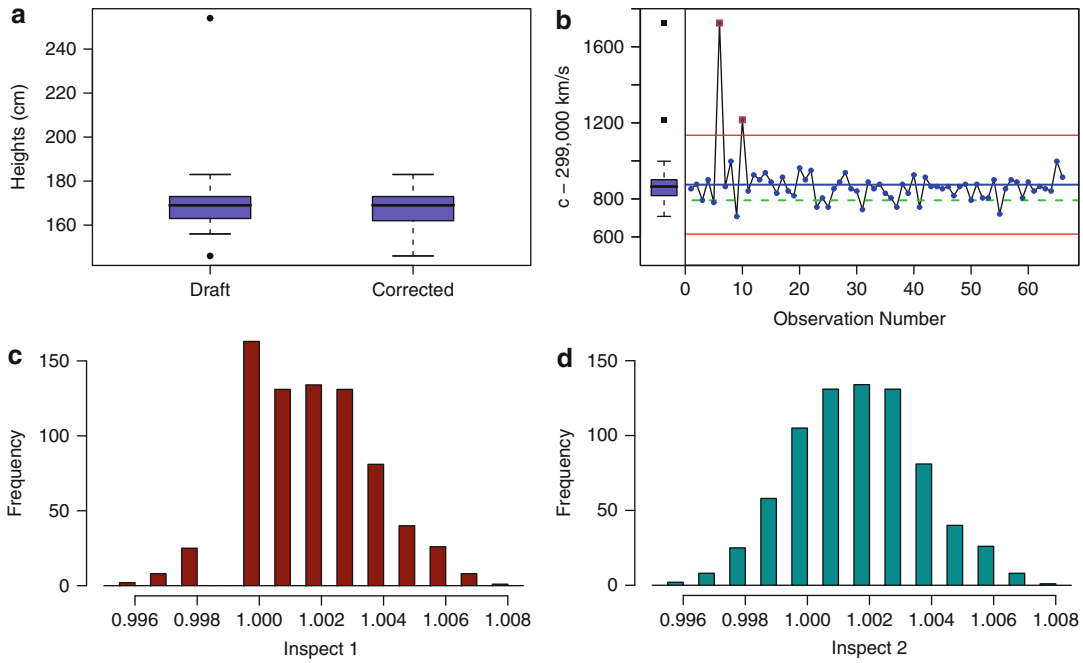
Among numerical descriptive statistics, the maximum also shows a problem with the draft data, but not that there is only one error. In Table 5 we compare descriptive statistics of two versions of the data to see the effect the one extreme outlier at 254 cm has on the mean, standard deviation, and skewness (in addition, of course, to the maximum) of our relatively small sample.

Such *data cleaning* is an important part of real-world data analysis. Datasets in textbooks are usually vetted before publication. However, large datasets from designed studies or data mining almost always contain errors: a male hysterectomy, a 185-year-old nursing home resident, over 5,000 withdrawals in a month from a modest savings account, and so on. Less absurd errors are more difficult to find, and one must be cautious using algorithmic screening that is too aggressive, for fear of missing an important changing trend.

Example 5 (Velocity of light) The currently accepted value of the velocity of light is $c = 299{,}792.5$ km/s. Around 1880, Simeon Newcomb made 66 determinations of the velocity of light as shown in the *control chart* in Fig. 4b (Stigler 1993). For convenience, we have subtracted 299,000 km/s from all observations in making the chart. Typically, a control chart has a horizontal line at the mean $\bar{x}$ of the data and lower and upper *control limits* at $\bar{x} \pm 2s$. As an alternative to the boxplot criterion for outliers, values outside the control limits warn of a process "out of control." The boxplot and control chart criteria for designating outliers happen to agree that there are two of questionable observations in Newcomb's data, but these methods do not always give the same results.

It is difficult to know whether the extreme values should be retained because they may be characteristic of Newcomb's apparatus or discarded simply as mistakes at the early stage of his experiment. One response to such a situation is to use a *trimmed mean*, obtained by routinely disregarding some percentage (most often 5 %) of the data at the extremes of each tail. The hope is that most erroneous observations will be eliminated without too much loss of information from throwing away good ones. For Newcomb's data, the mean of all 66 observations is 874.7, the mean without the 2 outliers is 856.1, and the 5 % trimmed mean (based on 60 observations) is 860.4. Upon adding 299,000 km/s to each of these 3 versions of the mean, we see that all of them are well above the currently accepted value of c.

Example 6 (Diameters of steel shafts) One component of an electromechanical device is a steel shaft about 1 cm in diameter. There is no harm if a shaft is a little too thick. However, if a shaft is even a little below 1.000 cm in diameter, the device fails. Before assembly, all shafts are inspected in order to discard the ones that are too thin.



Univariate Descriptive Statistics, Fig. 4 Example 4: (**a**) Boxplots of student heights before and after correction of a data entry error. Example 5: (**b**) Control chart of measurements of the velocity of light (minus 299 km/s), highlighting two extreme values, with a corresponding boxplot at *left*. Example 6: Histograms of dishonest (**c**) and honest (**d**) measurements of diameters of steel shafts

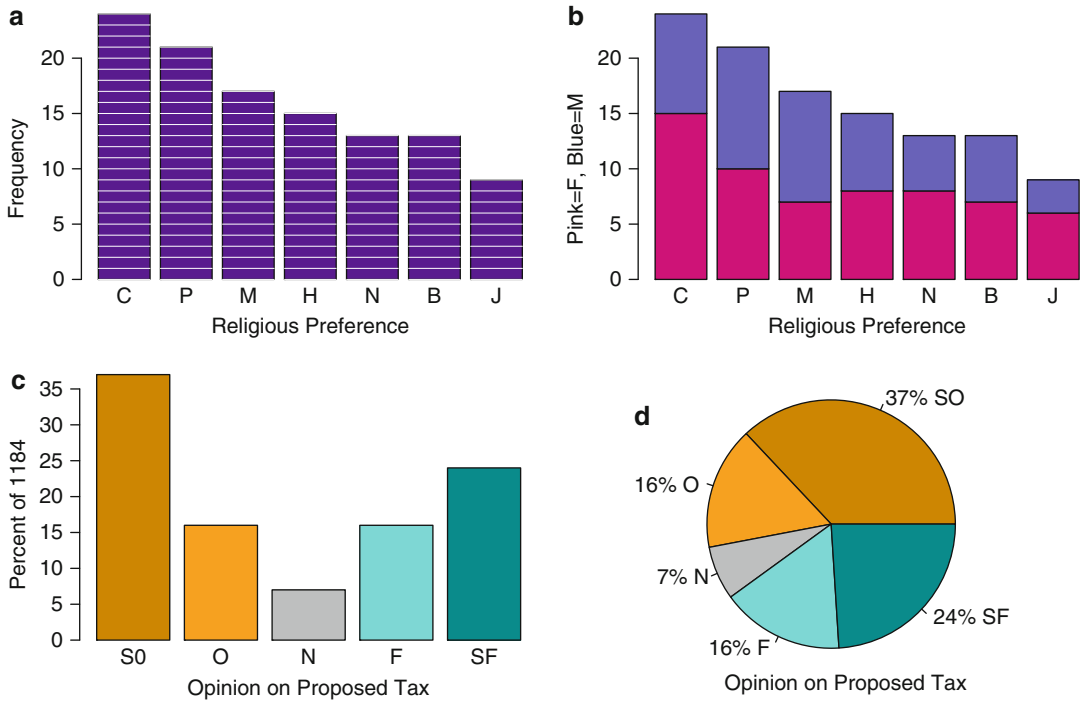
Univariate Descriptive Statistics, Table 5 Heights of students

	Draft data	Corrected
Min	146	146
Q_1	163	162
Median	169	169
Mean	**170.6**	**168.1**
Q_3	173	173
Max	**254**	**183**
SD	**15.43**	**8.06**
Skewness	**3.99**	**0.38**

After several bad days of device failures, a quality engineer made a histogram of measurements of a batch of 750 shafts that had just gone through inspection and noticed the gap at 0.999 cm adjacent to a too-tall bar at 1.000 cm; see Fig. 4c. Interviews revealed that newly hired inspectors were "helping" to reduce the number of junked shafts by fudging data to pass through shafts they considered trivially smaller than 1.000 cm. A reinspection of all 750 shafts yielded the measurements in panel (d).

Neither numerical descriptive statistics nor a boxplot would have revealed this difficulty, which does not involve extreme values. Also, a carelessly binned histogram might not show anything wrong. In data exploration, it is a good idea to look at many possibly relevant numerical and graphical descriptions and to focus on the ones that are most useful. (These are simulated data very closely modeled after unavailable proprietary data, but with a slight change in scale.)

Example 7 (Student religious preferences) A survey of the 112 residents of an international student dormitory at a US university asked each participant to record religious preference and gender. The results are shown in Fig. 5a, b. Each of the seven religious preferences chosen by participants is shown as a bar, in decreasing order of frequency: Catholic, Protestant, Muslim, Hindu, None, Buddhist, Jewish. Only the first letter of each nominal class is used to label bars in



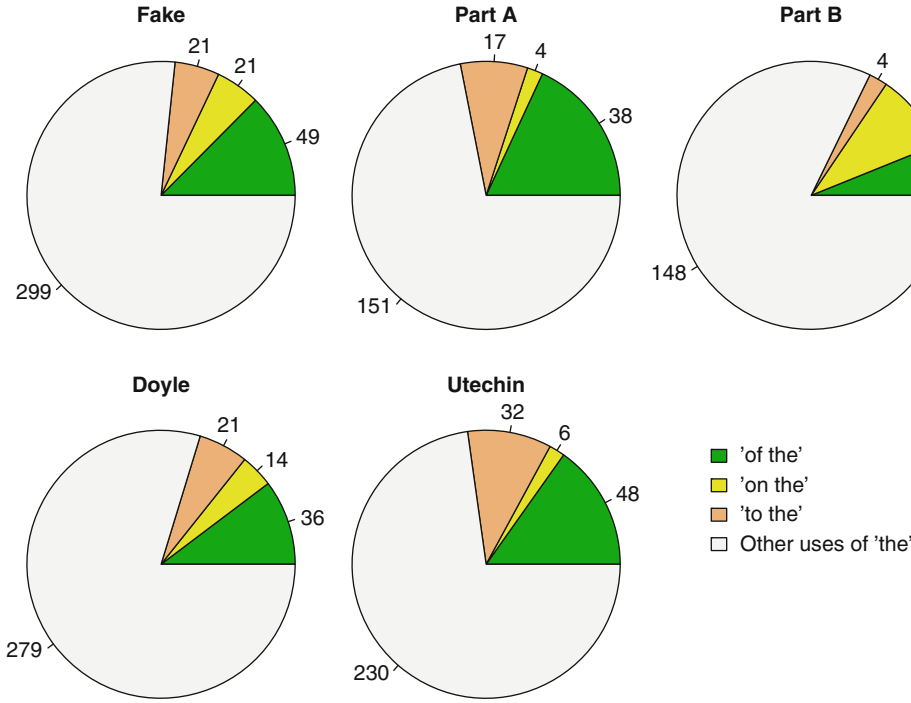
Univariate Descriptive Statistics, Fig. 5 Example 7: (**a**) The bar chart shows 112 observations of a nominal variable on religious preference with seven classes arranged in decreasing order of frequency. Each segment of a bar represents one of the respondents. (**b**) The stacked bar chart shows the same data, but with colored segments indicating gender. Example 8: (**c**) The bar chart shows percentages of 1,184 observations in each of 5 classes of an ordinal variable on opinions about a proposed tax on sugary drinks, arranged in order of decreasing opposition to the tax. (**d**) The pie chart of the same data uses the same colors as the bar chart to indicate the natural order of the classes

the graphs. Gender is not shown in panel (a), and each segment of a bar represents one respondent. This graphical device is intended to emphasize how little data we have here. In panel (b), gender counts are stacked within each bar, with the first segment (up from the baseline) representing women.

> Class frequencies run from 24 to 9. Dormitory residents may find the differences in frequencies large and interesting. However, with only 112 subjects, a formal test cannot distinguish the sample distribution from equally likely random assignments of respondents to classes.

One measure of the diversity of the relative frequencies r_i of nominal data is $\sum_{j=1}^{k} r_i^2 = 0.155$. In biology and ecology this is often called the *Simpson index* (Simpson 1949) and in business and economics the *Herfindahl-Hirschman index* (HHI). Roughly, it is the probability that two randomly chosen participants are in the same class. When there are k classes, the *minimum* possible value, representing the *greatest* diversity, is $1/k$. In our example, the minimum value 0.143 would occur if each class contained exactly 16 of the 112 respondents. There are many other measures of diversity and entropy, which we do not discuss here.

Example 8 (Opinions on a proposed tax) In October 2012, a Field Poll asked 1,184 registered voters in California for their opinions about a proposed tax (1 cent per ounce, roughly 1 euro cent per 40 ml) on sugar-sweetened beverages (Allday 2013). Responses were recorded on an ordinal scale: strongly oppose, oppose somewhat, no opinion expressed, favor somewhat, or strongly favor. Results, claimed accurate within ±3 %, are shown in Fig. 5c, d. The bar chart in



Univariate Descriptive Statistics, Fig. 6 Example 9: The five pie charts represent different samples of writing. In each of them, counts of various uses of the word *the* are represented by sectors as indicated in the legend. Because the pie chart for Part A of the fake Sherlock Holmes story has nearly the same structure as the pie chart for a story by Nicholas Utechin just below it, we conclude that Utechin wrote that part of the fake story

panel (c) shows percentages, and the bars are arranged in the natural order of the scale.

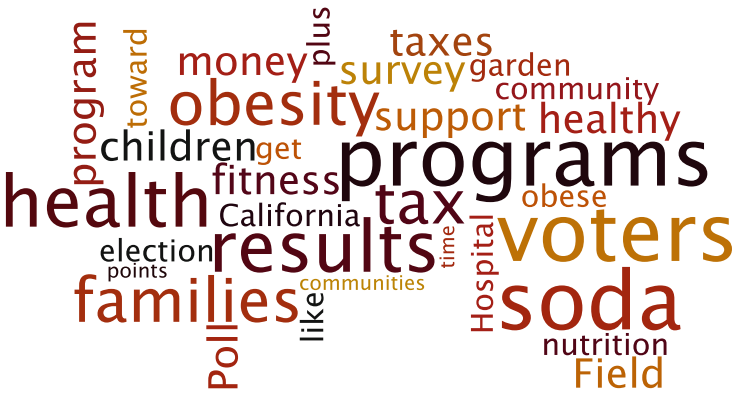
The pie chart in panel (d) shows the same information, but here the colors play a key role. To emphasize the ordered nature of the data, we used two tones of one hue for the two negative opinions and a different hue for positive opinions. It is clear from this chart that, even if those who expressed no opinion came to have a positive view, the majority would still be in opposition. Also, the darker tones, which represent stronger views, occupy more than half of the pie, possibly indicating strong polarization on this issue.

> However, modest changes in the wording of questions of a poll can have large effects on the results. For example, when it was suggested that proceeds of such a tax might be devoted to improving school nutrition and supporting physical activity programs, almost half of the respondents in the Field Poll were strongly in favor of this arrangement, and 68 % were at least somewhat in favor.

Example 9 (Author identification) The science of stylometry has been used to establish authorship of manuscripts. One technique is to compare patterns of usages of words that involve the basic structure of language rather than specific subject matter. In English, these are often "little words," such as articles, prepositions, and various forms of the verbs *to be* and *to have* (Morton 1987).

In the mid-1970s two admirers of A. Conan Doyle's stories about Sherlock Holmes collaborated to write some additional Holmes adventures, attempting to imitate Doyle's style, wit, and suspense. One of their fake stories "The Earthquake Machine" can be divided into Parts A and B (Mitchelson and Utechin 1976). We know that Austin Mitchelson wrote one part and Nicholas Utechin the other, but not who wrote which part.

As a possible means of deciding authorship, counts of occurrences of the phrases *of the*,



Univariate Descriptive Statistics, Fig. 7 A word cloud based on a newspaper article (Allday 2013) describing the poll of Example 8. Larger fonts represent more frequently used subject-matter words. Colors are random and decorative (a few frequently occurring local California place and institution names have been suppressed)

on the, *to the*, and all other uses of *the* were made: altogether, there were 210 uses of *the* Part A and 180 Part B, for a total of 390 in the entire fake story. For comparison, counts of these same uses of *the* were made for two additional stories: one non-Holmes story written later by Utechin (1977) (316 uses) and one of Doyle's authentic Holmes stories (Doyle 1891) (350 uses). No comparable writing sample for Mitchelson was available (Phrase counts are from Morton 1987).

The resulting phrase counts are shown in the pie charts of Fig. 6. It seems clear that Utechin's pie chart is similar in structure to the pie chart of Part A – and not to the chart of Part B. So we guess Utechin must have written Part A. Also, when Parts A and B are merged, it is difficult to distinguish between the pie chart for the whole of the fake Holmes adventure and the chart for Doyle's writing sample. Perhaps different kinds of word counts could be used to make that distinction.

> It seems we have enough data to warrant the speculation that Utechin wrote Part A. Formal statistical tests reveal no difference in the distribution of uses of *the* between Utechin and Part A. However, Utechin and Part B differ significantly. Similarly, we found significant differences comparing Part A vs. Part B and Utechin vs. Doyle. So, among our pie charts, the ones that appear to be substantially different really are different. By contrast, suppose each of the writing samples had been only a quarter as large and counts in each *the*-class happened to be reduced to about a quarter. Any change in the appearance of the pie charts would be barely noticeable. However, with so little data, the difference between any two pie charts could be ascribed to chance.

A considerable difficulty with some stylometric studies has been the lack of standards: for selecting test phrases, for deciding what constitutes "proof" pro or con, and for methods of combining information from multiple sources. Such issues have been discussed by statisticians in an investigation of the disputed authorship of certain Federalist Papers (essays of the late 1780s that advocated the adoption of the US Constitution) (Mosteller and Wallace 1964).

By contrast with the use of small structural words in author identification, the word cloud in Fig. 7 shows subject-matter words used in a newspaper article (Allday 2013) explaining the background of the proposed soda tax of Example 8. Words repeated more often appear in larger type. Word clouds of text mined from recent news articles, blogs, tweets, and so on have been used to display topics and issues that represent recent trends of social interest, without directly seeking to quantify opinion or to distinguish among sources.

Acknowledgments

For uniformity and to facilitate production, all of the graphs in this essay were made using core R for Windows (R Developmnt Core Team 2013). Summary statistics were computed in R, except that skewness computations were done in Minitab 16. Graphs of comparable or better quality can be made with other statistical and graphics software, including special graphics

libraries in R. We thank Bryson Hagerman for helpful discussions about making pie and bar charts in R. Data in our Examples 1–5 and 9 are also used in Trumbo (2002), where complete datasets are shown. Some of these data have been used and quoted in various other statistics textbooks (Moore 2013; Rice 2006; Weisberg 1985; Suess and Trumbo 2010).

Cross-References

- ▸ Centrality Measures
- ▸ Distance and Similarity Measures
- ▸ Regression Analysis
- ▸ Social Network Datasets
- ▸ Spatial Statistics
- ▸ Theory of Statistics, Basics, and Fundamentals

References

Allday E (2013) Support builds for soda tax on one condition. San Francisco Chronicle. Available: http://www.sfgate.com/health/article/More-support-soda-tax-if-for-kids-health-4277195.php. Accessed 8 July 2013 (Online)

Doyle AC (1891) A case of identity, the adventures of Sherlock Holmes. Available: http://www.eastofthe-web.com/short-stories/UBooks/CaseIden.shtml. Accessed 8 July 2013 (Online)

Freedman D, Diaconis P (1981) On the histogram as a density estimator: l_2 theory. Probab Theory Relat Fields 57(4):453–476

Frigge M, Hoaglin DC, Iglewicz B (1989) Some implementations of the boxplot. Am Stat 43(1):50–54

Hoaglin DC, Iglewicz B, Tukey JW (1986) Performance of some resistant rules for outlier labeling. J Am Stat Assoc 396:991–999

Ott RL, Longnecker M (2006) An introduction to statistical methods and data analysis 3e. Duxbury, Boston. ISBN: 978-0-495-01758-5

Majundar DN, Rao C (1956) Bengal anthropometric survey, 1945: a statistical study. Difference between evening and morning measurements of stature. Sankhya 19:296–298. These data also appear in Rao CR (1989) Statistics and truth: putting chance to work. International Cooperative Publishing House

Miller BF, Kean CB (1987) Encyclopedia of medicine, nursing and applied sciences, pp 697–698

Mitchelson A, Utechin N (1976) The earthquake machine. Belmont Tower Books, New York

Moore DS (2013) The basic practice of statistics 6e. Freeman, New York. ISBN: 978-1-4641-0254-7

Morton AQ (1987) Literary detection – how to prove authorship and fraud in literature and documents, Chapter 14. Charles Scribner's Sons, New York

Morton DE, Saah AJ, Silbery SL, Owens WL, Roberts MA, Saah ME (1982) Lead absorption in children of employees in a lead-related industry. Am J Epidemiol 115(4):549–555

Mosteller FW, Wallace D (1964) Inference and disputed authorship. The Federalist papers. Springer, New York

NIST/SEMATECH (2012) Engineering statistics handbook, section 1.3.5.11: measures of skewness and kurtosis. Available: http://www.itl.nist.gov/div898/handbook/eda/section3/eda35b.htm. Accessed 8 July 2013 (Online)

Purcaur G, Zorilescu D (1971) A magnitude-frequency relation for the lognormal distribution of earthquake magnitude. Pure Appl Geophys 87(1):43–53

R Development Core Team (2013) R: a language and environment for statistical computing, R foundation for statistical computing. Available: http://www.r-project.org. Accessed 8 July 2013 (Online)

Rice JA (2006) Mathematical statistics and data analysis 3e. Duxbury, Boston. ISBN: 978-0534399429

Silverman B (1986) Density estimation. Chapman and Hall, London

Simpson EH (1949) Measurement of diversity. Nature 163:688

Stigler SM (1977) Do robust estimators work for real data? Ann Stat 5(6):1055–1098. These data also appear in Rice (2006) and in Moore DS, McCabe GP (1993) Introduction to the practice of statistics, 2e. W.H. Freeman

Sturges HA (1926) The choice of a class interval. J Am Stat Assoc 21(153):65–66

Suess EA, Trumbo BE (2010) Introduction to probability simulation and Gibbs sampling with R. Springer, New York. ISBN: 978-0-387-40273-4

Trumbo BE (2002) Learning statistics with real data. Duxbury, Boston. ISBN:0-534-36213-3

Tukey JW (1977) Exploratory data analysis. Addison-Wesley, Reading. ISBN:0-201-07616-0

U.S. Geological Survey Menlo Park CA Berkeley Seismological Laboratory University of California Berkeley Berkeley CA, Northern California Seismic Network Data Center (NCEDC) Northern California Earthquake Catalog Search. Search: 8/28/2000–9/9/2000, Latitude 32–42N, Longitude 114–125W. Available: http://quake.geo.berkeley.edu/ncedc/catalog-search.html. Accessed 7 July 2013 (Online)

Utechin N (1977) The villages Perche. Blackwood's Magazine, Edinburgh

Utts J (2013) Seeing through statistics 3e. Wadsworth, Belmont. ISBN: 978-0534394028

Velleman P, Hoaglin DC (1981) Applications, basics and Computing of exploratory data analysis. Duxbury, Boston. ISBN: 0878722734
Weisberg S (1985) Applied linear regression. Wiley. Data provided by Yellowstone National Park geologist Roderick A. Hutchinson reposted on pages 231–234. The Geyser Observation and Study Association. Available: http://www.geyserstudy.org/geyser.aspx?pGeyserNo=OLDFAITHFUL. Accessed 8 July 2013 (Online)

Universal Scaling

▸ Scale-Free Nature of Social Networks

Unsolicited Bulk e-Mail

▸ Spam Detection, E-mail/Social Network

Update

▸ Topic Modeling in Online Social Media, User Features, and Social Networks for

Urban Form

▸ Analysis and Planning of Urban Networks

Urban Networks

▸ Spatial Networks

User Action

▸ User Behavior in Online Social Networks, Influencing Factors

User Behavior

▸ User Behavior in Online Social Networks, Influencing Factors

User Behavior in Online Social Networks, Influencing Factors

Bo Hu, Mohsen Jamali, and Martin Ester
School of Computing Science, Simon Fraser University, Burnaby, BC, Canada

Synonyms

Online social media; Social factor; Social network; Social weight; User action; User behavior

Glossary

Social Network Snapshot A snapshot of social network
Item Adoption Adopting an item to another user
FW Factor weight
IC Independent cascade
Action Prediction Predicting user actions

Introduction

Millions of people now use social networking websites to enjoy online interaction with friends and meeting new people. Social network sites, such as Flixster, Flickr, and Digg, are attracting an increasing number of users, many of whom have integrated these sites into their daily practices. Meanwhile, the rapid development of online social networking is increasingly attracting the attention of academic and industry researchers and motivating the study of prominent phenomenons, such as homophily (Miller et al. 2009), i.e., similar users are more likely to connect to each other, and social

influence (Friedkin 1998), i.e., friends tend to influence each others preferences and behavior. Thus, a lot of research has investigated patterns of user interests and user actions in social networks for viral marketing (Domingos and Richardson 2001; Richardson and Domingos 2002), recommender systems (Bell et al. 2007; Koren 2008), and trust propagation (Guha et al. 2004; Jamali and Ester 2009).

Most of the current work considers only binary social relations (friends or not). However, the real cases of online social networks are much more complicated. Due to the swift and low cost of creating social relations online, people tend to establish a large number of online friendships with different social groups, such as friends, relatives, co-workers, or even strangers, with different degrees of friendships strength (Granovetter 1973; Gilbert and Karahalios 2009). This means that valuable information may be lost if only one strength of friendship is considered.

Several works (Goyal et al. 2010; Liu et al. 2010; Tang et al. 2009; Xiang et al. 2010; Gilbert and Karahalios 2009) have modeled the tie strength between pairs of users as hidden parameters and developed methods to learn them from observed user behavior. However, none of them considers all factors affecting the dynamics of online social networks. As discussed in the literature (Anagnostopoulos et al. 2008; Koren 2009), the factors driving user behavior include (1) homophily and social influence, (2) user, (3) item, and (4) sparsity factors. Since in this paper we do not focus on distinguishing homophily from social influence effects (La Fond and Neville 2010; Aral et al. 2009), we use the term "social correlation" to represent the combined effect of homophily and social influence. For example, if a friend of the user has gone to the "The Social Network" movie, she may watch it as well (social correlation), and the user could watch the movie because of her own preference for drama movies (user); the movie may be so popular that she cannot resist it (item), or users do not have enough time to check on every movie (sparsity). The sparsity factor significantly reduces the amount of available data and should not be neglected. When modeling user behavior in online social media, all above factors should be considered, and some parameters representing these factors will be introduced in our proposed model.

Key Points

In this paper, our goal is to learn a probabilistic model of modeling user behavior in online social media, where user, item, social correlation, and sparsity factors are all included. In the proposed model, we address the problem of learning the strengths of social correlation, user, item, and sparsity factors. We propose a probabilistic model for learning these strengths by maximizing the joint probability of the observed user behavior.

The major contributions of this paper are as follows:

1. We introduce a more comprehensive user behavior model, which considers all major factors impacting user behavior: social correlation, user, item, and sparsity.
2. The proposed model also considers the strengths of different factors, which is crucial in online social networks.
3. We present an extensive experimental evaluation on four real data sets (Epinions, Flixster, Flickr, and Digg, which are crawled from http://www.epinions.com, http://www.flixster.com, http://www.flickr.com, and http://www.digg.com, respectively), demonstrating the accuracy of our proposed model and providing interesting insights into the most important effects in real-life social networks.

Historical Background

In this section, we review related work from the following three areas: (1) maximization of social influence, (2) testing of existence for homophily and social influence, and (3) modeling of social correlation and the strength of social correlation.

Many works have been widely investigated to maximize social influence in social networks. Domingos and Richardson (2001) and Richardson and Domingos (2002) are the first to describe

the problem of choosing influential sets of users given a social network and considering propagation of the social influence from a data mining perspective. Kempe et al. (2003) prove that the problem formulated as discrete optimization is NP-hard and present a greedy approximation algorithm. Leskovec et al. (2007) present an optimized greedy algorithm. Chen et al. (2009, 2010) also propose two faster greedy algorithms for influence maximization. All the above methods only find the top-k influential users and do not model the social influence at the interaction level. Besides these models find the top-k influential users to predict the future actions, while we exploit the actions in the past to learn the social influence weight among pairs of users.

Some work tests for the existence of homophily and social influence. While we determine the local strength of correlation between two users, these methods test whether globally in the entire data set there is significant homophily or social influence. Anagnostopoulos et al. (2008) propose a randomization method to show that there is significant social correlation in tagging behavior data set, which cannot be attributed to influence. La Fond and Neville (2010) also present a randomization procedure for assessing whether a data set exhibits social influence and homophily effects. Aral et al. (2009) distinguish social influence from homophily for item adoption using match sampling.

A growing work of research (Cosley et al. 2010; Crandall et al. 2008) has focused on modeling social networks and user behavior. Crandall et al. (2008) studies the temporal evolution of the interplay between attribute similarity and social ties. They identify and model the interaction between social influence and selection effects in Wikipedia data set and find feedback effects between these two factors. From a different perspective, Cosley et al. (2010) considers snapshot observations of a social network and the detailed temporal dynamics of a social network. A method which infers the detailed dynamics from observed snapshots is proposed. The main focus of these works is modeling user behavior by taking social influence into account, and they do not model the strength of all effects. Furthermore, none of them considers all effects influencing the user actions.

The works most relevant to our proposed method are those of Goyal et al. (2010), Liu et al. (2010), Tan et al. (2010), Tang et al. (2009), Xiang et al. (2010), and Gilbert and Karahalios (2009). Tang et al. (2009) introduce the problem of topic-based social influence analysis and propose a topical affinity propagation approach. Their recent work (Liu et al. 2010) further investigates the topic-based social influence and propagation on heterogeneous networks and proposes a generative graphical model. Their work is based on heterogeneous networks with user nodes and content nodes. They assume that the user is going to perform an action, and their method predicts on which item the user is going to perform that action. This problem definition is different from ours: given an item, we predict whether a user is going to perform an action or not. Another work (Tan et al. 2010) from the same authors also proposes a graphical model to model the actions and predict user's future actions; however, they do not consider items in their problem definition. Hence, they ignore the item and sparsity effects.

Xiang et al. (2010) propose a latent variable model which estimates relationship strength between users in online social networks from user behavior. Their model mainly considers the effect of homophily. Gilbert and Karahalios (2009) present a predictive model that maps social media data to tie strength. The model builds on a real data set of over 2,000 social ties, and it distinguishes between strong and weak ties. While in this paper, we focus on all effects.

Gomez Rodriguez et al. (2010) maximizes the social influence weight based on user behavior. However, the social network is the output of their method, while it is the input in our model. Moreover, in their experiments, the model only considers one item, while our model involves multiple items.

Goyal et al. (2010) learn the influence probabilities using the independent cascade model (IC). At a given timestamp, each user is either active or inactive, and each individual's tendency to become active increases as more of its friends

become active. Since there is an assumption that the probability of different friends influencing the user are independent, each friend has a probability to trigger the user to perform the action. The influence probability $p_{v,u}$ from user v to user u is computed by Eq. 1, where $|I_{v,u}|$ is the number of items adopted from v by u, and $|I_v|$ is the number of actions performed by v.

$$p_{v,u} = \frac{|I_{v,u}|}{|I_v|} \tag{1}$$

Therefore, the joint probability $p_u(N_u)$ can be defined as follows.

$$p_u(N_u) = 1 - \prod_{v \in N_u} (1 - p_{v,u}) \tag{2}$$

where N_u is the set of friends of user u.

The authors of Goyal et al. (2010) claim that the influence of a particular action of a user on her friends is decreasing over time. Accordingly, a temporal model (IC_{temp}) is introduced, and the temporal influence probability $p^t_{v,u}$ is computed in Eq. 3, where $p_{v,u}$ is the probability in Eq. 1 and $\tau_{v,u}$ is the average time difference between $t_{u,i}$ and $t_{v,i}$ over all items i that u adopted from v, where $t_{u,i}$ is the time of user u perform an action on item i.

$$p^t_{v,u} = p_{v,u} \times e^{-\frac{t - t_{v,i}}{\tau_{v,u}}} \tag{3}$$

The Proposed Factor Weight Model

In this section, we first explain some definitions and intuitions behind our model, and then we introduce our research problem and describe our probabilistic model.

Preliminaries

In this subsection, we introduce the basic definitions and our research problem. We are given users (henceforth, interchangeable with "individuals") and items (henceforth, interchangeable with "products," "movies," "photos" etc.), and friend relationships (henceforth, interchangeable with "neighbors"). For the sake of clarity, we reserve special indexing letters for distinguishing users from items: u and v for users and i for items. We use t for time or date. In addition, we reserve superscripts for time and subscripts for users and items.

In this work, a social network is represented as a directed graph $G = (U, E)$, where U denotes the set of users, and a directed edge $(u, v) \in E, u \in U, v \in U$ from user u to user v represents the fact that u has added v as her "friend." We assume an ordered set of discrete timestamps $T = \{1, 2, \ldots t_{\text{end}}\}$, where t_{end} is the last timestamp. We use $t_{u,v} \in T$ to denote the time of creating the edge from u to v. We define a snapshot of a social network as containing all the nodes, and edges that have been created up to time t.

Definition 1 (Social network snapshot) A social network snapshot is defined as $G^t = (U^t, E^t)$, where U^t is the set of users at time $t \in T$ and E^t is the set of links between users that have been created at or before time t.

Note that most available data sets, such as Epinions, Flixster, Flickr, and Digg in our experiments, contain only creations of nodes and edges and no deletions, but our model can easily be applied to both cases.

In this paper, we consider a sequence of social network snapshots $[G^1, G^2, \ldots, G^{t_{\text{end}}}]$ together with user actions. In our context, user actions are performed on items, and we assume that only one type of actions on one item for only once. We associate a user with a set of actions, i.e., the set of items on which she has performed an action at some point of time. We define the action more formally as follows:

Definition 2 (Action) Given a set of items I, each user $u \in U$ is associated with a set of actions, denoted as

$$I_u = \{i_1, i_2, \ldots, i_M\}$$

with $I_u \subseteq I$. We use $t_{u,i} \in T$ to denote the time of the action of user u on item i.

Interaction is a common way to maintain friend relationships. Typically, users are willing to spend some time to communicate with other users, especially friends. In the Flixter data set, for example, users' interaction is based on actions on the same movies. We define this interaction formally as item adoption similar to the notation of Crandall et al. (2008).

Definition 3 (Item adoption) For a pair of friends u and v, $(v, u) \in E$, and an item i, there is an item adoption $i_{u,v}$ from user u to user v, if the following two conditions hold: (1) $t_{u,i} < t_{v,i}$, i.e., user u performs the action on item i ahead of user v (2) $t_{v,u} < t_{u,i}$ and $t_{v,u} < t_{v,i}$, i.e., the time $t_{v,u}$ of establishing the friend relationship is ahead of both actions by user u and user v. Furthermore, the set of all items adopted by v from u is defined as follows:

$$I_{u,v} = \{i | i \in I_u \wedge i \in I_v \wedge t_{u,i} < t_{v,i} \wedge t_{v,u} < t_{u,i}\}$$

Notice that based on the definition of item adoption, there are two types of actions. Actions that are adopted from friends are called "social actions," i.e., users follow their friends, while other actions are not adopted from friends, which we call "individual actions." Social actions could be attributed to social correlation. An individual action is the result of the user and item effects, which is well known in the recommender systems literature, where latent factor models (Jamali and Ester 2010; Koren 2008, 2009), especially matrix factorization techniques, have been successful in explaining the user actions by characterizing both the user and item factors. For movies, for example, each user factor measures how much the user likes movies based on user characteristics, such as demographic attributes and preferences for movie genres, while the item factors may measure how much the movie attracts users on different dimensions, such as movie genre or the depth of character development.

Based on the above definitions, we introduce the factor strength prediction problem in online social networks as follows:

Problem 1 (Factor strength prediction) Given a sequence of social network snapshots $[G^1, G^2, \ldots, G^{t_{\text{end}}}]$, sets of actions I_u for all users $u \in U$ and a series of timestamps T, the goal of the factor strength prediction problem is to output sets of strength of the social correlation, users, items, and sparsity factors.

The Probabilistic Model

In order to infer the strength of social correlation, user, item, and sparsity factors, we plan to model the user states.

For every timestamp $t \in T$ and every item $i \in I$, there are two states for a user u: active, i.e., the user has already performed the action on the item, and inactive, i.e., the user has not performed the action. Therefore, we use a binary random variable $x^t_{u,i} \in \{0, 1\}$ to represent a state of the relationship between the user u and the item i, where $x^t_{u,i} = 1$ indicates that user u has performed an action on item i by time t, and $x^t_{u,i} = 0$ means that user u has not performed an action on item i by time t. We also assume that once a user u performed an action on a particular item i at time t, she stays active on this item up to the end, so that all random variables in the set $\{x^{t+1}_{u,i}, \ldots, x^{t_{\text{end}}}_{u,i}\}$ are equal to one. In our data sets, all user states are observed. Our goal is to compute the probability of these observed states and to obtain a joint probability of distribution of the $x^t_{u,i}$ for all $u \in U$, $i \in I$, and $t \in T$. To that purpose, we introduce the factor weight (FW) model.

Now we define the notations needed for our FW model, which are listed in Table 1. N^t_u is the set of friends of user u by time t. We assume that user actions are driven by social correlation, user, item, and sparsity factors. In the following, we use term "weight" instead of the term "strength," and we use different parameters represent different factors. Let $w_{u,v}$ denote the social correlation weight from user u to user v, θ_u denote the user weight of user u, θ_i denote the item weight of item i, and ρ denote the sparsity weight. In addition, we use $X_{N^t_u,i}$ to represent the set of all $x^t_{v,i}$ for $v \in N^t_u$.

User Behavior in Online Social Networks, Influencing Factors, Table 1 Notations in the FW model

$x_{u,i}^t$	Binary random variable: whether user u has performed an action on item i by time t
N_u^t	Set of friends of user u by time t
$X_{N_u^t,i}$	Set of random variables $x_{v,i}^t$ for all friends $v \in N_u^t$ of user u
$w_{u,v}$	Social correlation weight from user u to user v
$W_{N_u^t}$	Set of $w_{u,v}$ for all $v \in N_u^t$
θ_u	User weight of user u
θ_i	Item weight of item i
ρ	Sparsity weight of online social network

To model the probability of the states of each user at each timestamp, we assume the following generative process. For every user u, every item i, at every time t, the following two steps are performed: (1) with probability $1-\rho$, user u is exposed to item i; (2) if user u is exposed to the item i, user u performs the action on item i with a probability depending on the combination of the weights of user, item, and social correlation factors. The intuition behind the model is that the online users cannot perform actions on all the items, because there are far too many items. Therefore, we attribute most of the absent actions to the sparsity weight ρ.

Now we introduce our FW model for user, item, social correlation, and sparsity weight prediction. Given a social network snapshot G^t and a set of actions up to time $t-1$, we model the conditional probability of the user u state on an item i by time t in Eqs. 4 and 5. We make a Markov assumption that the state of user u by time t is conditionally independent of all the previous states given the state of all users at time $t-1$.

$$p(x_{u,i}^t = 1 | x_{u,i}^{t-1}, W_{N_u^{t-1}}, X_{N_u^{t-1},i}, \theta_u, \theta_i, \rho) = \begin{cases} (1-\rho) \times p(x_{u,i}^t = 1 | W_{N_u^{t-1}}, X_{N_u^{t-1},i}, \theta_u, \theta_i) & \text{if } x_{u,i}^{t-1} = 0 \\ 1 & \text{if } x_{u,i}^{t-1} = 1 \end{cases} \tag{4}$$

$$p(x_{u,i}^t = 1 | x_{u,i}^{t-1}, W_{N_u^{t-1}}, X_{N_u^{t-1},i}, \theta_u, \theta_i, \rho) = 1 - p(x_{u,i}^t = 0 | x_{u,i}^{t-1}, W_{N_u^{t-1}}, X_{N_u^{t-1},i}, \theta_u, \theta_i, \rho) \tag{5}$$

Note that after a user performs an action on a given item, the probability of performing the same action afterwards is always one, and in order to model user states at timestamp $1 \in T$, we assume a special timestamp 0, which is ahead of timestamp 1, for which we set all user state random variables x to 0.

We assume that a user is influenced by each friend according to the social correlation weight similar to the linear threshold model (Kempe et al. 2003). At each timestamp, the neighbor has a chance to trigger the user to perform the action, and for each user, the social correlation weight of active neighbors is summed up. In addition to take into account the user and item factors, we add the user and item weights. In conclusion, we define $p(x_{u,i}^t = 1 | W_{N_u^{t-1}}, X_{N_u^{t-1},i}, \theta_u, \theta_i)$ as follows:

$$p(x_{u,i}^t = 1 | W_{N_u^{t-1}}, X_{N_u^{t-1},i}, \theta_u, \theta_i) = \frac{1}{1 + e^{-\left(\sum_{v \in N_u^{t-1}} (w_{v,u} x_{v,i}^{t-1}) + \theta_u + \theta_i\right)}} \tag{6}$$

where we use a sigmoid function (nonlinear) for two reasons. The first one is to simplify the parameter learning. The second one is that we observe the social correlation weight grows up to a number of friends but then it stabilizes, which

means when the number of active friends meets a threshold, adding another active friend cannot make contribution anymore. The sigmoid function fits this observation very well based on our experiment results. To simplify the presentation, we replace $\sum_{v \in N_u^{t-1}}(w_{v,u} x_{v,i}^{t-1})$ with $\sum_W$.

We assume that user u can influence user v if and only if v has added u as her friend and define the social correlation weights only on pairs of users that are "friends" at the end of the time window t_{end}. Note that social correlation is not symmetric, and the social correlation weight $w_{u,v}$ has in general a different value than $w_{v,u}$.

We plug Eq. 6 into Eq. 4, and we obtain the following Eq. 7 for the case $x_{u,i}^{t-1} = 0$.

$$\begin{aligned} p(x_{u,i}^t = 1 | W_{N_u^{t-1}}, X_{N_u^{t-1},i}, \theta_u, \theta_i, \rho) \\ = (1-\rho) \times \left(g\left(\sum\nolimits_W + \theta_u + \theta_i \right) \right) \end{aligned} \tag{7}$$

where $g(x) = \frac{1}{1+e^{-x}}$ is the sigmoid function. Furthermore, we obtain $p(x_{u,i}^t = 0 | W_{N_u^{t-1}}, X_{N_u^{t-1},i}, \theta_u, \theta_i, \rho)$ for the case $x_{u,i}^{t-1} = 0$ as follows:

$$\begin{aligned} p(x_{u,i}^t = 0 | W_{N_u^{t-1},u}, X_{N_u^{t-1},i}, \theta_u, \theta_i, \rho) \\ = \rho + (1-\rho) \\ \times \left(1 - g\left(\sum\nolimits_W + \theta_u + \theta_i \right) \right) \end{aligned} \tag{8}$$

In order to make the learning problem tractable, we assume that all the weights are static, i.e., do not change in the course of the time as also implemented in Goyal et al. (2010) and Xiang et al. (2010). However, one might argue that the weight of correlation of an item adoption decreases over time. We also consider such a temporal version of our model. Intuitively, the social correlation from user u to user v is at its peak at the time of action by user u. After that time, the correlation is decreasing and finally disappears after a certain amount of time. We assume that the peak time coincides with the time $t_{u,i}$ of the friends' action. Hence, we replace the term $w_{v,u}$ in Eq. 7 by $w_{v,u} \times e^{-\frac{t-t_{u,i}}{\tau}}$, where τ is the decaying parameter. In our experiments, we tune τ in order to obtain the best results. Based on the above equations, we obtain the following likelihood function L as a joint probability of every observed user states on every item on every timestamps in Eq. 9.

$$\begin{aligned} L = \prod_{u \in U} \prod_{i \in I} \prod_{t \in T} p(x_{u,i}^t = 1 | x_{u,i}^{t-1}, W_{N_u^{t-1}}, X_{N_u^{t-1},i}, \theta_u, \theta_i, \rho)^{x_{u,i}^t} \\ \times p(x_{u,i}^t = 0 | x_{u,i}^{t-1}, W_{N_u^{t-1}}, X_{N_u^{t-1},i}, \theta_u, \theta_i, \rho)^{1-x_{u,i}^t} \end{aligned} \tag{9}$$

Parameter Learning

In this section, we employ a maximum likelihood estimation method to learn the latent parameters $\{w_{u,v}, \theta_u, \theta_i, \rho\}$ from the observed user states $\{x_{u,i}^t\}$ of our probabilistic model. The likelihood function is given in Eq. 9. To simplify the learning process, we replace the parameter ρ with values in the range between $[0, 1]$ with a sigmoid function $g(\rho)$ in order to make the domain of ρ from $-\infty$ to $+\infty$.

Instead of simply maximizing the log-likelihood function, we minimize the following error function in Eq. 10, which is the sum of the negative log-likelihood and quadratic regularization terms (Bishop 2006) on $w_{v,u}$, θ_u, θ_i, and ρ.

$$\Phi = \sum_{u\in U}\sum_{i\in I}\sum_{t\in T}\left(\left(-x_{u,i}^{t}\times\ln\left(1-g(\rho)\right)-x_{u,i}^{t}\times\ln\left(g\left(\sum_{W}+\theta_u+\theta_i\right)\right)\right)\right.$$
$$\left.+\left(-(1-x_{u,i}^{t})\times\ln\left(g(\rho)+(1-g(\rho))\times\left(1-g\left(\sum_{W}+\theta_u+\theta_i\right)\right)\right)\right)\right)$$
$$+\frac{\lambda_w}{2}\sum_{u\in U}\sum_{v\in N_u^{t-1},t\in T}{w_{v,u}}^2+\frac{\lambda_{\theta_u}}{2}\sum_{u\in U}{\theta_u}^2+\frac{\lambda_{\theta_i}}{2}\sum_{i\in I}{\theta_i}^2+\frac{\lambda_\rho}{2}\rho^2 \tag{10}$$

The initial values of all parameters are sampled from normal distributions with zero mean. We define the update method of the parameters $\mu \in (\{w_{u,v}\}\cup\{\theta_u\}\cup\{\theta_i\}\cup\{\rho\})$ in Eq. 11.

$$\mu_{\text{new}} = \mu_{\text{old}} - step\times\frac{\partial\Phi}{\partial\mu} \tag{11}$$

where *step* is the learning rate which varies from different random variables and different data sets. We omit the derivative equations to save space.

Key Applications

In this section, we report our experimental results on four real-life data sets comparing various versions of our FW model and one of the latest state-of-the-art methods (Goyal et al. 2010). Closely related works are Xiang et al. (2010), Liu et al. (2010), Tan et al. (2010), Tang et al. (2009), and Goyal et al. (2010). Xiang et al. (2010) use user features to compute the strength of homophily on pairs of friends. If we represent the action as user features, then we could make this work very similar to Goyal et al. (2010). In Liu et al. (2010), Tan et al. (2010), and Tang et al. (2009), their models are for the heterogeneous networks, while our model is for the homogeneous networks. In conclusion, we choose the most related work (Goyal et al. 2010) as our comparison partner.

Data Sets

The data sets in our experiments are from Epinions, Flixster, Flickr, and Digg. We remove those users who have no friends and the users who have not performed any actions. For the sake of efficiency, we used samples of the data sets. Our samples preserve the distributions of the number of actions for users as well as for items.

We present relevant statistics for four data sets in Table 2 and describe the data sets in the following:

- In the Epinions data set, which is available in http://alchemy.cs.washington.edu/data/epinions/, an action is users' ratings on products, and item adoption is that friends have rated the same item. Notice that the items in Epinions are from different categories including digital cameras, music, and books.
- The Flixster data set was crawled by us from Flixster.com, which is publicly available from http://www.cs.sfu.ca/~sja25/personal/datasets/. Actions are defined as ratings on movies by users, and item adoption is defined as ratings on the same movies by a user and a friend. Unlike Epinions, all the items in the Flixster data set are movies, but there are different genres of movies: action, romance, fantasy, and so on.
- The Flickr data set was collected by Cha et al. (2009) from Flickr.com. The action is defined as a user tagging photos as favorite. We regard two friends tagging the same photo as item adoption.
- The Digg data set was collected by Lin et al. (2009) from Digg.com. The action is defined as digging stories, i.e., users state that they like a story. We regard two friends digging the same story as item adoption.

Notice that the statistics of social and individual actions show that Flixster and Digg have much higher proportions of social actions than Flickr and Epinions.

User Behavior in Online Social Networks, Influencing Factors, Table 2 Statistics of the Epinions, Flixster, Flickr, and Digg data sets

Statistics	Epinions	Flixster	Flickr	Digg
Users	11 K	20 K	13 K	1.2 K
Items	11 K	13 K	14 K	32 K
Social relations	280 K	130 K	1.3 M	28 K
Friends per user	25	7	101	23
Actions per item	25	62	61	35
Actions per user	25	39	61	959
Actions	277 K	770 K	729 K	1.1 M
Social actions	27 K(10 %)	208 K(27 %)	140 K(19 %)	297 K(27 %)
Individual actions	250 K(90 %)	562 K(73 %)	589 K(81 %)	803 K(73 %)
Beginning time	November 2000	November 2005	November 2006	August 2 2008
Ending time	February 2002	November 2009	March 2007	August 27 2008
Time interval	60 days	90 days	15 days	1 day
Number of timestamps	14	10	10	25

Experimental Setup

In our experiments, we split the whole data set into a training data set and a test data set. For each user, we pick a timestamp $t \in T$ so that the training data set contains all the user states from the beginning time to time t, which contains 90 % of all user actions, while the test data set contains the remaining user states from time t to the end t_{end}, which contains the 10 % most recent user actions.

Our goal is to learn the factor weights based on likelihood function on the training data set, and then we use the learned factor weights to predict the probability of user states in the test data set. Given the learned parameters, we calculate the prediction probability using Eq. 4. We label the user active if the predicted probability of an action is greater than the activation threshold and inactive otherwise. In our experiments, we used 1,000 different threshold values from 0 to 1 with step 0.001.

As in Goyal et al. (2010), the evaluation metric used in our experiments is the ROC curve, which plots the true positive rate (TPR = TP/(TP + FN)) versus the false-positive rate (FPR = FP/(FP + TN)), where TP, FP, FN, and TN represent true positives, false-positives, false negatives, and true negatives respectively. The closer the curve is to the point (0,1), the better the performance. Each point on the ROC curve corresponds to one possible value of the activation threshold, which is the same for all users. To measure the contribution of a factor, we compute the AUC (area under ROC curve) of the corresponding partial model and compare it to the AUC of FW model.

We compare the following methods for factor weight learning:

- IC. This is the discrete Bernoulli model proposed in Goyal et al. (2010), which achieved the best performance in their experiments.
- FW_w. This method is a version of our model which uses only the social correlation effect.
- $FW_{w+\text{temp}}$. This is a version of FW_w, which considers temporal social correlation weights.
- $FW - w$. This is a partial FW model, where the social correlation effect is removed.
- $FW - \theta_u$. This is the model that takes all factors into account except the user effect.
- $FW - \theta_i$. This is an FW partial model, where we remove the item effect from our full model.
- FW. This is our full FW model with all effects.

Note that all the above different versions of the FW model include the sparsity factor. All the methods were implemented in C++, and all experiments were performed on a server running Windows 7 with an Intel Xeon E5630 2.53 GHz CPU and 8 GB RAM. Based on our preliminary experiments, we set the stop criterion parameter

Tol to 1,000, the maximum iterations to 300, the $Step$ to $1e-07$ for ρ and to $1e-03$ for the remaining variables.

Experimental Results

In this subsection, we present the results of our experiments, the prediction performance comparison, the factor contribution analysis, and the temporal social correlation weight analysis. Since the IC model is only applicable to predict social actions, in our experiments, we ignore the individual actions in the test data set.

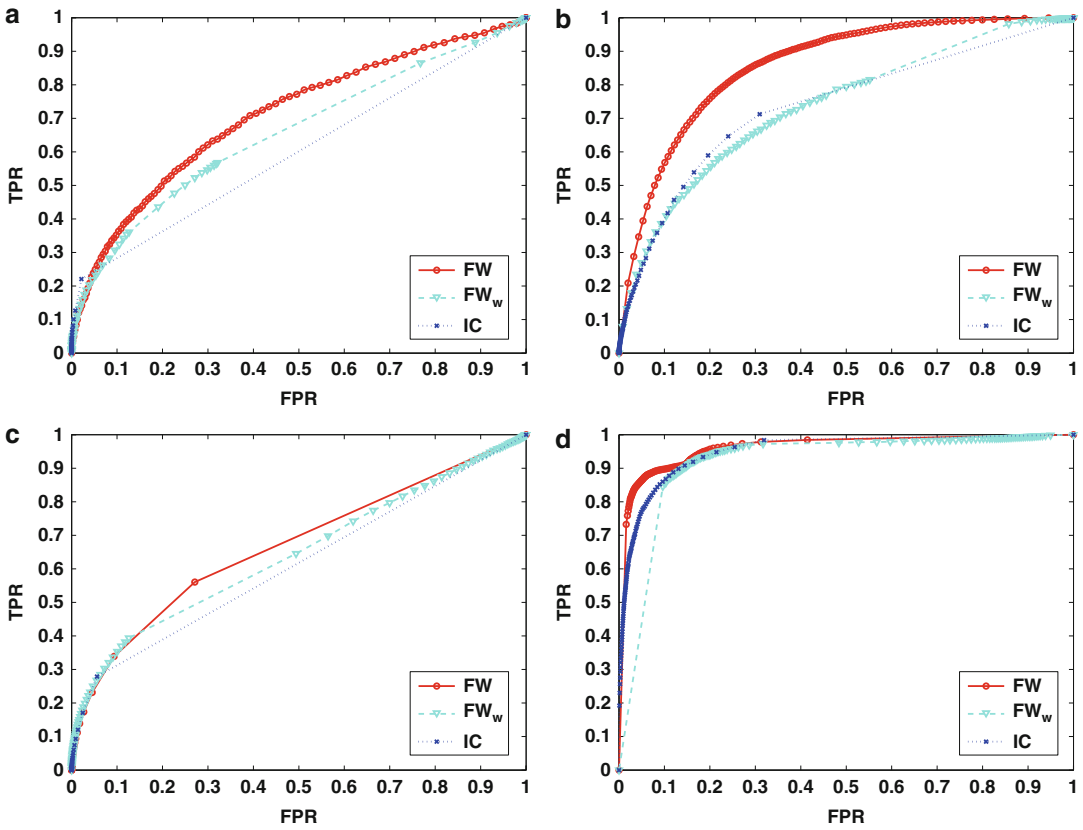
Prediction Performance Comparison

Figure 1 reports the ROC curves of our FW model and the state-of-the-art IC model in the Epinions, Flixster, Flickr, and Digg data sets. The FW model consistently outperforms the IC model in Epinions, Flixster, and Flickr data sets by a substantial margin, while it performs similar in the Digg data set. Our explanation is that in Digg the user action of digging a story is usually influenced by her friends' actions (see also Table 3). Considering only the social correlation factor, we observe similar results of the FW_w model and the IC model in the Flixster and Digg data sets, while there is a slight gain in Epinions and Flickr. This indicates that the sparsity factor can help improve the "social correlation only" model FW_w, because some actions that fail to give rise to item adoption can be attributed to the social correlation, while others may be caused by the sparsity factor (i.e., users missed recommendations from friends). Besides, the gain increases after we add either user or item effects into the FW_w model. These results prove our hypothesis that social actions can be attributed to the user, item, and social correlation factors, not only to the social correlation factor as done in Goyal et al. (2010), Tang et al. (2009), and Xiang et al. (2010).

Factor Contribution Analysis

To analyze the contribution of individual factors in our FW model, we compare the FW model with the $FW - w$, $FW - \theta_u$, $FW - \theta_i$, and FW_w models. Our FW model contains the social correlation, user, and item effects. Specifically, we remove one of the three factors from our full model, and we train and evaluate the performance of the partial models. Figure 2 presents the resulting AUC and shows that all three factors make a significant difference on all four data sets. We also observe that the item factor is strong in the Epinions and Flixster data sets, but is very weak in the Flickr data set. Surprisingly, the $FW - \theta_i$ model, which has no item factor, slightly outperforms the FW model in Flickr. Why does the item factor contribute in the Epinions and the Flixster data sets, but does not help in the Flickr data set? In the Epinions and Flixster data sets, the items (products in Epinions and movies in Flixster) are public and all users can perform actions on them, while in the Flickr data set the items (photos) are normally private on which only friends check. Hence, Flickr users are not aware of the popularity of items, which likely explains the absence of an item factor. In Digg, the AUC remains the same when we remove either the user or item effect, but it decreases when we remove both of them. The reason may be that user actions of digging a story are based on their friends' behavior and disregard the user or item effects.

To rank the contribution of each factor, we first compute the AUC difference between the full model FW and the partial models $FW - w$, $FW - \theta_u$, and $FW - \theta_i$ and rank the factors in descending order of their AUC differences, e.g., the social correlation factor is more important than the user factor if the AUC difference between FW and $FW - w$ is larger than the AUC difference between FW and $FW - \theta_u$. Table 3 presents the ranking of the social correlation, user, and item effects on all four data sets. The most important factor is the item, user, user, and correlation factor in Epinions, Flixster, Flickr, and Digg respectively. This shows that all these three factors make significant contributions in some, but different data sets. As also discussed in Anagnostopoulos et al. (2008), in some data sets the social correlation factor is not as important as we might expect, such as Epinions and Flixster.



User Behavior in Online Social Networks, Influencing Factors, Fig. 1 ROC comparison of IC and FW models. (**a**) Epinions data set. (**b**) Flixster data set. (**c**) Flickr data set. (**d**) Digg data set

User Behavior in Online Social Networks, Influencing Factors, Table 3 Ranking of social correlation, user, and item factors

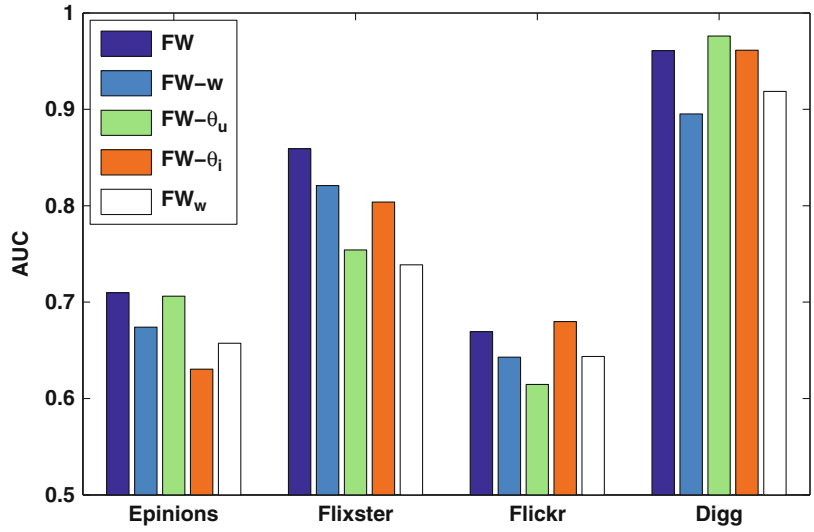
Data set	First	Second	Third
Epinions	Item	Correlation	User
Flixster	User	Item	Correlation
Flickr	User	Correlation	Item
Digg	Correlation	Item	User

Temporal Social Correlation Weight Analysis

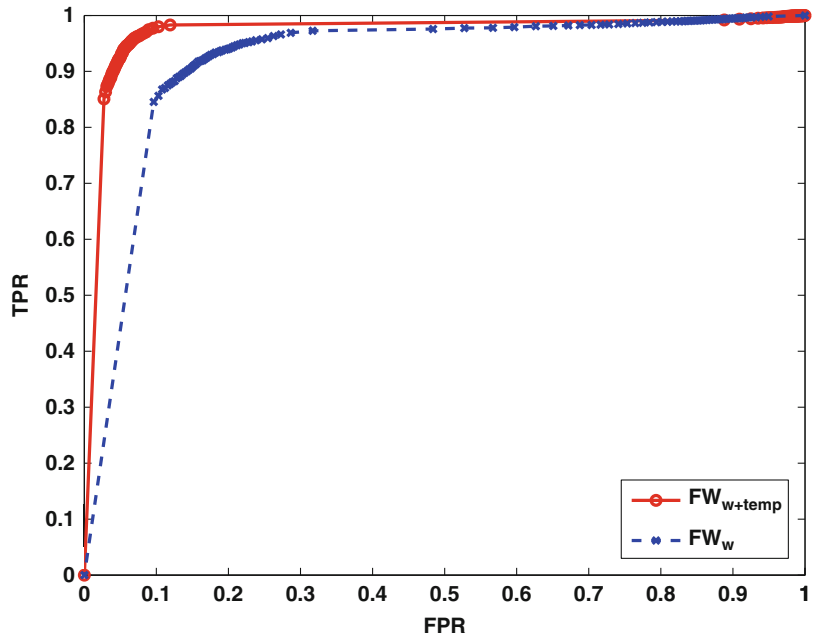
We analyze the models with temporal social correlation weights. On Epinions, Flixster, and Flickr, our experiments show only minor differences between the temporal and static versions for the FW model, likely because the correlation factor is not the major effect impacting the user actions. Figure 3 reports the ROC of the FW_w and $FW_{w+\text{temp}}$ models in the Digg data set. It turns out that the $FW_{w+\text{temp}}$ model has clearly better performance than the FW_w model, which indicates that the social correlation is indeed decreasing in the course of the time. These results confirm similar results observed for the IC model in Goyal et al. (2010).

Future Directions

With the rapid growth of interest in online social networks, such as Flixster and Facebook, social networks are changing the way we live. Social correlation plays an important role in social networks and leads people to adopt the behavior of their friends. However, not all friends have the same degree of social



User Behavior in Online Social Networks, Influencing Factors, Fig. 2 AUC of FW, $FW - w$, $FW - \theta_u$, $FW - \theta_i$, and FW_w models



User Behavior in Online Social Networks, Influencing Factors, Fig. 3 ROC comparison of FW_w and $FW_{w+\text{temp}}$ on the Digg data set

correlation on a user. Users follow the behavior of some of their friends more than other friends. Moreover, the user, item, and sparsity effects impact the user behavior as well. In this paper, we propose a probabilistic model, called factor weight model (FW), to learn the social correlation, user, item, and sparsity weights from user actions. Our model maximizes the

joint probability of the observed user actions. We perform experiments on four real-life data sets (Epinions, Flixster, Flickr, and Digg). We evaluate the quality of action prediction, and experimental results demonstrate that the FW model achieves better performance than a state-of-the-art method.

This paper suggests several interesting directions for future research. First, besides the problem of predicting whether a user will perform an action on specific items, predicting a rating for a given item is also a natural task in recommendation. Although learning and inference will be more complex, considering rating values may improve the accuracy of learning the factor weights. Intuitively, the social correlation weight between two users who have the similar ratings is larger than that one of two users who have a large rating difference. Second, the social correlation we consider in this paper is context independent. However, it would be interesting to investigate the social correlation weight in different contexts. Users may influence friends in some contexts while in other contexts they may not. For example, in Flixster a user may influence friends in action movies if she is an action movie fan, but if she has no interest in other types of movies, she probably has no social influence on other types of movies.

Cross-References

▸ Extracting and Inferring Communities via Link Analysis
▸ Extracting Individual and Group Behavior from Mobility Data
▸ Human Behavior and Social Networks
▸ Inferring Social Ties

References

Anagnostopoulos A, Kumar R, Mahdian M (2008) Influence and correlation in social networks. In: KDD, Las Vegas

Aral S, Muchnik L, Sundararajan A (2009) Distinguishing influence-based contagion from homophily-driven diffusion in dynamic networks. Proc Natl Acad Sci USA 21:21544

Bell R, Koren Y, Volinsky C (2007) Modeling relationships at multiple scales to improve accuracy of large recommender systems. In: KDD, San Jose

Bishop CM (2006) Pattern recognition and machine learning. Information science and statistics. Springer, New York

Cha M, Mislove A, Gummadi KP (2009) A measurement-driven analysis of information propagation in the Flickr social network. In: WWW, Raleigh

Chen W, Wang Y, Yang S (2009) Efficient influence maximization in social networks. In: KDD, Paris

Chen W, Wang C, Wang Y (2010) Scalable influence maximization for prevalent viral marketing in large-scale social networks. In: KDD, Washington, DC

Cosley D, Huttenlocher DP, Kleinberg JM, Lan X, Suri S (2010) Sequential influence models in social networks. In: ICWSM, Washington, DC

Crandall D, Cosley D, Huttenlocher D, Kleinberg J, Suri S (2008) Feedback effects between similarity and social influence in online communities. In: KDD, Las Vegas

Domingos P, Richardson M (2001) Mining the network value of customers. In: KDD, San Francisco

Friedkin NE (1998) A Structural theory of social influence. Cambridge University Press, Cambridge

Gilbert E, Karahalios K (2009) Predicting tie strength with social media. In: CHI, Boston

Gomez Rodriguez M, Leskovec J, Krause A (2010) Inferring networks of diffusion and influence. In: KDD, Washington, DC

Goyal A, Bonchi F, Lakshmanan LV (2010) Learning influence probabilities in social networks. In: WSDM, New York

Granovetter M (1973) The strength of weak ties. Am J Sociol 78:1360–1380

Guha R, Kumar R, Raghavan P, Tomkins A (2004) Propagation of trust and distrust. In: WWW, Manhattan

Jamali M, Ester M (2009) Trustwalker: a random walk model for combining trust-based and item-based recommendation. In: KDD, Paris

Jamali M, Ester M (2010) A matrix factorization technique with trust propagation for recommendation in social networks. In: RECSYS, Barcelona

Kempe D, Kleinberg J, Tardos E (2003) Maximizing the spread of influence through a social network. In: KDD, Washington, DC

Koren Y (2008) Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: KDD, Las Vegas

Koren Y (2009) Collaborative filtering with temporal dynamics. In: KDD, Paris

La Fond T, Neville J (2010) Randomization tests for distinguishing social influence and homophily effects. In: WWW, Raleigh

Leskovec J, Krause A, Guestrin C, Faloutsos C, VanBriesen J, Glance N (2007) Cost-effective outbreak detection in networks. In: KDD, San Jose
Lin Y-R, Sun J, Castro P, Konuru R, Sundaram H, Kelliher A (2009) Metafac: community discovery via relational hypergraph factorization. In: KDD, San Jose
Liu L, Tang J, Han J, Jiang M, Yang S (2010) Mining topic-level influence in Heterogeneous networks. In: CIKM, Toronto
Miller M, Lynn S, James C (2009) Birds of a feather: homophily in social networks. Ann Rev Sociol 27:415–444
Richardson M, Domingos P (2002) Mining knowledge-sharing sites for viral marketing. In: KDD, Edmonton
Tan C, Tang J, Sun J, Lin Q, Wang F (2010) Social action tracking via noise tolerant time-varying factor graphs. In: KDD, Washington, DC
Tang J, Sun J, Wang C, Yang Z (2009) Social influence analysis in large-scale networks. In: KDD, Paris
Xiang R, Neville J, Rogati M (2010) Modeling relationship strength in online social networks. In: WWW, Raleigh

User-Centered Pattern Detection

▸ Fraud Detection Using Social Network Analysis, a Case Study

User Classification

▸ Collective Classification

User-Generated Content

▸ Spatiotemporal Proximity and Social Distance

User Interaction

▸ Weblog Analysis

User Interfaces for Data Disclosure

▸ Graphical User Interfaces for Privacy Settings

User Sentiment and Opinion Analysis

Luana Batista
École de technologie supérieure, Montreál, QC, Canada
Independent scholar, Montréal, QC, Canada

Synonyms

Opinion mining; Sentiment analysis; Text mining

Glossary

ER Error Rate
AER Average Error Rate
Favorability Percentage of *tweets* classified as positive

Definition

Opinion analysis, also known as sentiment analysis, is a subfield of text mining which aims at identifying the user sentiment with respect to a specific subject. This subjective information is extracted from texts by using a combination of machine learning and natural language processing techniques.

Introduction

Due to the large amount of information contained in Twitter messages (*tweets*), often consisting of opinions about different subjects, they quickly became an attractive source of data for opinion analysis (Tumasjan et al. 2010; Pak and Paroubek

2010; Go et al. 2009; Davidov et al. 2010; Bollen et al. 2011; Bifet and Frank 2010; Bermingham and Smeaton 2010; Batista and Ratte 2012; Barbosa and Feng 2010; Agarwal et al. 2011). "How positive (or negative) are people about our products/services?" (Pak and Paroubek 2010), "Which candidate are you most likely to vote for?" (Batista and Ratte 2012; Tumasjan et al. 2010), or even "Is the public mood correlated to the Dow Jones Industrial Average (DJIA)?" (Bollen et al. 2011) are examples of questions that have been automatically answered from Twitter messages using a *sentiment classifier*.

In the field of opinion analysis, unigrams have been successfully employed as binary features. Indeed, in the particular case of microblogging, which are characterized by containing very short texts (e.g., *tweets* cannot exceed 140 characters), the occurrence of certain bigrams or higher-order n-grams may be rare, resulting in sparse feature vectors (Go et al. 2009).

In this work, a Naive Bayes classifier is trained with unigrams in order to classify *tweets* as positive or negative. The Stanford University's Twitter database (Go et al. 2009) is employed to validate the classifier performance. This database contains a subset of hand-annotated *tweets*, which have been employed by some authors to test their systems (Pak and Paroubek 2010; Go et al. 2009; Bifet and Frank 2010).

Moreover, we investigate the hypothesis that opinion analysis extracted from *tweets* may be employed in marketing research (Kolb 2008), one of the most important steps of the development of a new product in a company. We consider the hypothetical scenario of a software company that decides to develop applications for Smartphones, and, before starting, they have to find out what is the most preferred mobile operating system in the market. In other words, by using the proposed system, the objective is to identify the sentiment of people about using Android, Iphone, Windows Phone, and Blackberry. A comparison is performed with statistical results obtained by *comScore Mobilens*® (http://www.comscore.com/Products/Audience_Analytics/MobiLens).

This paper is organized as follows. The next section describes our experimental methodology, that is, the employed datasets, and the adopted strategies for preprocessing, feature extraction, and performance evaluation. Then, the experimental results are presented and discussed.

Experimental Methodology

The proposed system is designed to deal with two classes of sentiments: positive and negative. Such as suggested by Read (2005), Twitter messages containing happy emoticons, e.g., :-), :), ;), =), belong to the positive class, while Twitter messages containing sad emoticons, e.g., :-(, :(, ;(, = (, :/, belong to the negative class.

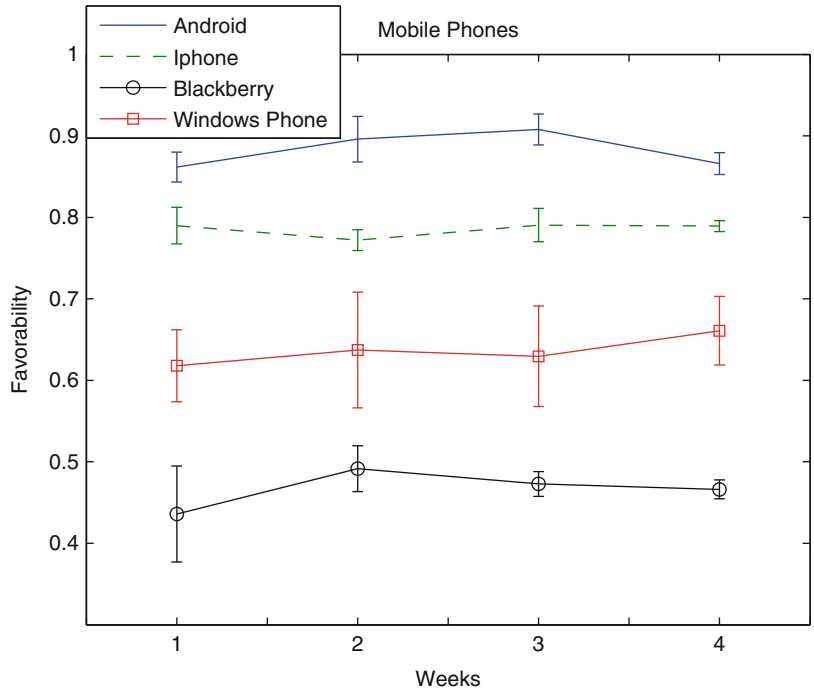
To analyze the performance of the system, a test set composed of hand-annotated *tweets* is used during the experiments (Go et al. 2009). Then, the system is employed as a marketing research tool in order to identify the sentiment of users about the main Smartphone platforms in the market, i.e., Android, Iphone (iOS), Windows Phone, and Blackberry. The results are compared to statistical results obtained by *comScore Mobilens*®.

Datasets

The proposed system is designed using the datasets indicated in Table 1. They come from the Stanford University's Twitter database (Go et al. 2009), where tst is composed of hand-annotated *tweets*. The training set was separated in two subsets: $trn_{(feat)}$, employed in the feature extraction step (i.e., generation of n-grams), and

User Sentiment and Opinion Analysis, Table 1 Data partitioning

Dataset	Positive class	Negative class
$trn_{(feat)}$	200,000	200,000
$trn_{(classif)}$	200,000	200,000
tst	177	182



User Sentiment and Opinion Analysis, Fig. 1 Smartphone platforms favorability, with corresponding standard deviations, obtained by the proposed approach for the period of February 17, 2013–March 17, 2013

Top Smartphone Platforms
3 Month Avg. Ending Jan. 2013 vs. 3 Month Avg. Ending Oct. 2012
Total U.S. Smartphone Subscribers Age 13+
Source: comScore MobiLens

	Share (%) of Smartphone Subscribers		
	Oct-12	**Jan-13**	**Point Change**
Total Smartphone Subscribers	*100.0%*	*100.0%*	*N/A*
Google	53.6%	52.3%	–1.3
Apple	34.3%	37.8%	3.5
BlackBerry	7.8%	5.9%	–1.9
Microsoft	3.2%	3.1%	–0.1
Symbian	0.6%	0.5%	–0.1

User Sentiment and Opinion Analysis, Fig. 2 Statistical results obtained by *comScore Mobilens*® for the "Top Smartphone Platforms" in the United States. Regarding the results presented in Fig. 1, Google corresponds to Android, Apple corresponds to Iphone, and Microsoft corresponds to Windows Phone (Picture extracted from http://www.comscore.com/Insights/Press_Releases/2013/)

$trn_{(classif)}$, employed for training the Naive Bayes classifier.

Finally, a set of 71,342 *tweets* was collected for the marketing research experiment. These data were obtained during a period of a month, by looking for *tweets* containing the following hashtags: #Android, #Iphone, #WindowsPhone, and #Blackberry.

Preprocessing and Feature Extraction

Once the *tweets* are collected, some preprocessing steps are performed (using Natural Language Toolkit library Bird et al. 2009) in order to obtain a proper dataset.

Preprocessing steps consist of removing pontuation, stop words (English language), emoticons, twitter user names (i.e., words starting by "@"), and "http" addresses. Then, the 2,000 most frequent unigrams are extracted from each class, that is, 2,000 from positive *tweets* and 2,000 from negative *tweets*.

Performance Evaluation

The system performance is evaluated by means of an average error rate (AER). The AER indicates the total error of the system, where the error rate (ER) of each class c (i.e., ER_c = number of misclassifications in c/total number of samples in c) is averaged as follows:

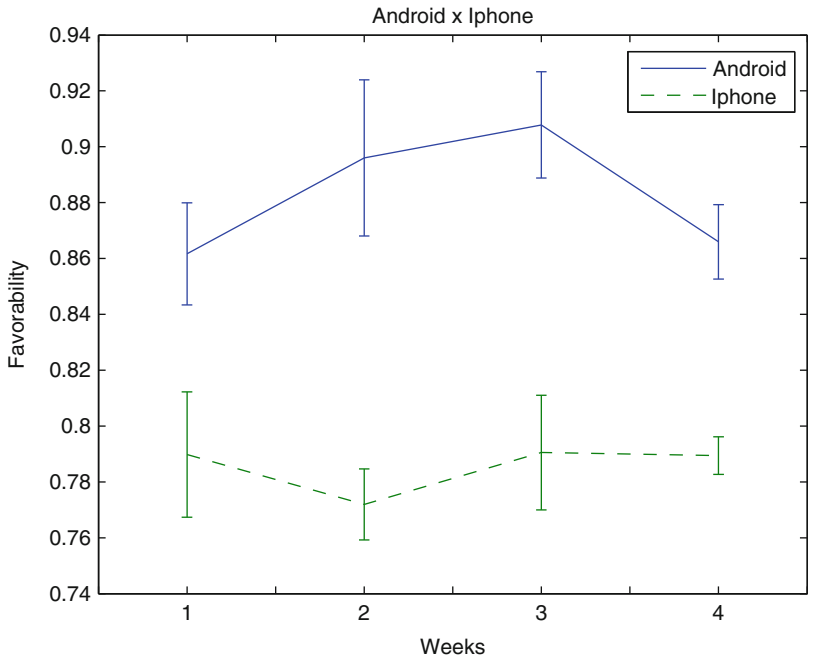
$$AER = \frac{\sum_{c=1}^{N} ER_c}{N},$$

where N is the total number of classes. In this work, $N = 2$, that is, positive class *versus* negative class.

Results and Discussions

As first experiment, we analyzed the system performance in classifying Twitter messages as positive or negative, regardless their subjects. By using $\text{trn}_{(\text{feature})}$ to extract the unigrams and $\text{trn}_{(\text{classif})}$ to train the Naive Bayes (see Table 1), an AER of 17.58 % was reached during test. This represents a satisfactory result, which is comparable to those reported in the literature (Pak and Paroubek 2010; Go et al. 2009; Bifet and Frank 2010). Remind that the test set, tst, is composed of hand-annotated *tweets*.

Then, the proposed system was employed to identify the sentiment of users about the main Smartphone platforms in the market. As mentioned, 71,342 *tweets* were collected during a period of 4 weeks (from February 17, 2013, to March 17, 2013) by looking for the



User Sentiment and Opinion Analysis, Fig. 3 "Top two" Smartphone platforms favorability, obtained by the proposed approach for the period of February 17, 2013–March 17, 2013

hashtags #Android, #Iphone, #WindowsPhone, and #Blackberry.

For each platform, we calculated the percentage of *tweets* classified as positive and used it as a measure of "favorability." Figure 1 shows how favorable are people with respect to each one of the considered Smartphone platforms, and Fig. 2 presents the results obtained by *comScore Mobilens*®. Although their analysis takes into account a greater period of time, our results follow a similar tendency – specially for Android and Iphone, as shows Fig. 3 – where Google corresponds to Android, Apple corresponds to Iphone, and Microsoft corresponds to Windows Phone (Symbian was not considered in our experiments).

Conclusion

In this work, a classification system based on unigrams and Naive Bayes was proposed in order to classify Twitter messages as positive or negative. As real-world application, the system was employed to identify the sentiment of users about the four main Smartphone platforms in the market, namely, Android, Iphone, Windows Phone, and Blackberry. Experimental results showed that this approach may provide useful information about people's opinions and be employed as a marketing research tool.

Cross-References

- ▸ Flickr and Twitter Data Analysis
- ▸ Multi-classifier System for Sentiment Analysis and Opinion Mining
- ▸ Sentiment Analysis in Social Media
- ▸ Twitter Microblog Sentiment Analysis

References

Agarwal A, Xie B, Vovsha I, Rambow O, Passonneau R (2011) Sentiment analysis of Twitter data. In: Workshop on languages in social media, LSM'11, Portland, pp 30–38

Barbosa L, Feng J (2010) Robust sentiment detection on Twitter from biased and noisy data. In: 23rd international conference on computational linguistics: posters, COLING'10, Beijing, pp 36–44

Batista L, Ratte S (2012) A multi-classifier system for sentiment analysis and opinion mining. In: IEEE/ACM international conference on advances in social networks analysis and mining, Istanbul, pp 96–100

Bermingham A, Smeaton A (2010) Classifying sentiment in microblogs: is brevity an advantage? In: Proceedings of the 19th ACM international conference on information and knowledge management, CIKM'10, Toronto, pp 1833–1836

Bifet A, Frank E (2010) Sentiment knowledge discovery in Twitter streaming data. In: 13th international conference on discovery science, Canberra. Springer, pp 1–15

Bird S, Klein E, Loper E (2009) Natural language processing with Python: analyzing text with the natural language toolkit. O'Reilly, Beijing

Bollen J, Mao H, Zeng X (2011) Twitter mood predicts the stock market. J Comput Sci 2:1–8

Davidov D, Tsur O, Rappoport A (2010) Enhanced sentiment learning using Twitter hashtags and smileys. In: 23rd international conference on computational linguistics: posters, COLING'10, Beijing, pp 241–249

Go A, Bhayani R, Huang L (2009) Twitter sentiment classification using distant supervision. Processing 150(12):1–6

Kolb B (2008) Marketing research: a practical approach, 0th edn. Sage, Los Angeles. doi:10.4135/9780857028013

Pak A, Paroubek P (2010) Twitter as a corpus for sentiment analysis and opinion mining. In: 7th conference on international language resources and evaluation, Valletta. European Language Resources Association

Read J (2005) Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In: ACL student research workshop, Ann Arbor, pp 43–48

Tumasjan A, Sprenger T, Sandner P, Welpe I (2010) Predicting elections with Twitter: what 140 characters reveal about political sentiment. Word J Int Linguist Assoc 280(39):178–185

Uses and Gratifications

- ▸ Web Communities Versus Physical Communities