Gaining insights in datasets in the shade of “garbage in, garbage out” rationale: feature-space distribution fitting

Gürol Canbek (✉ gurol@canbek.com)
https://orcid.org/0000-0002-9337-097X

Research Article

Keywords: data quality, dataset quality, data profiling, machine learning, classification, feature engineering, malware

Posted Date: February 21st, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1369128/v1

Version of Record: A version of this preprint was published at WIREs Data Mining and Knowledge Discovery on March 30th, 2022. See the published version at https://doi.org/10.1002/widm.1456.

Gaining insights in datasets in the shade of “garbage in, garbage out” rationale: feature-space distribution fitting

Gürol Canbek

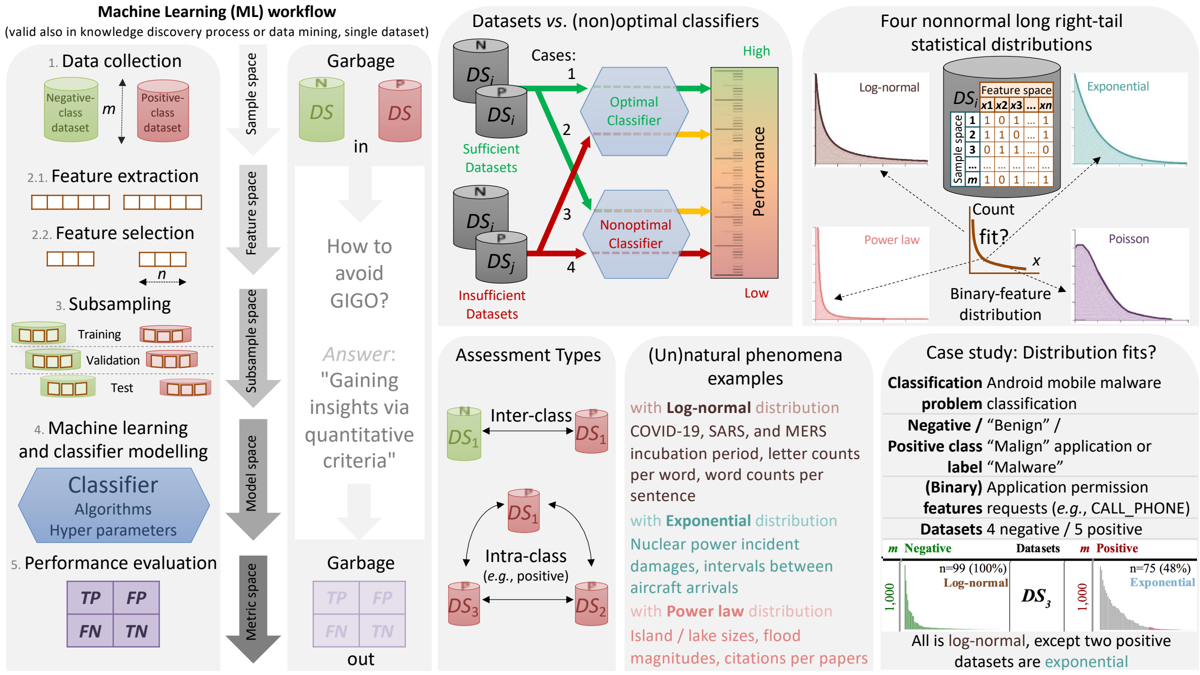
0000-0002-9337-097X, https://gurol.canbek.com, gurol@canbek.com

Abstract

This article emphasises the negative effect of the “Garbage In, Garbage Out” (GIGO) rationale and the importance of ensuring the dataset quality in Machine Learning (ML) based classification applications to achieve high and generalisable performance. Researchers should integrate the insights gained by quantitative analysis of the datasets’ sample and feature spaces into the initial ML workflow. As a specific contribution towards achieving such a goal, a complete approach was suggested to quantify datasets in terms of *feature frequency distribution* characteristics (*i.e.* how the features in the available samples comprising the datasets are frequent?). The approach was demonstrated in eleven benign and malign (malware) Android application datasets belonging to six academic Android mobile malware classification studies. The permissions requested by the applications such as CALL_PHONE compose a relatively high-dimensional binary-feature space. The results have shown that the distributions fit well into two of the *four long right-tail statistical distributions*: log-normal, exponential, power law, and Poisson. Precisely, log-normal was the most exhibited statistical distribution except the two malign datasets that were in exponential. This study also explores statistical distribution fit/unfit feature analysis enhancing the insights in feature space.

Further, the study compiles phenomena examples in the literature exhibiting these statistical distributions that should be considered for interpreting the fitted distributions. In conclusion, conducting well-formed statistical methods provides a clear understanding of the datasets and the intra-class and inter-class differences before proceeding with selecting features and building a classifier model. Feature distribution characteristics should be the one to analyse beforehand.

Graphical/Visual Abstract and Caption



with **Log-normal** distribution
COVID-19, SARS, and MERS incubation period, letter counts per word, word counts per sentence

with **Exponential** distribution
Nuclear power incident damages, intervals between aircraft arrivals

with **Power law** distribution
Island / lake sizes, flood magnitudes, citations per papers

Classification problem	Android mobile malware classification
Negative / Positive class label	“Benign” / “Malign” application or “Malware”
(Binary) features	Application permission requests (*e.g.*, CALL_PHONE)
Datasets	4 negative / 5 positive

All is log-normal, except two positive datasets are exponential

Garbage in garbage out degrades the performance of knowledge discovery process, data mining, and machine-learning workflows requiring optimal classifiers and sufficient datasets. The article suggests quantifying feature-frequency distributions by fitting power law, log-normal, and exponential right-tail distributions.

Submitted: 12 December 2020, First Revision: 6 September 2021

Competing Interest: The author declares that he has no conflict or competing of interest.

1. INTRODUCTION

Classification is a specific problem or task in Machine Learning (ML) at which a computer program (*i.e.* a classifier) improves its performance through learning from experience (Mitchell, 1997, p. 2). The experience is gained by providing labelled examples (*i.e.* training and sometimes validation dataset) of one or more classes that share common properties or characteristics (*i.e.* features) to a classifier that maps the properties into the class labels. The classifier’s success is evaluated on the different sets of labelled examples (*i.e.* a test dataset). After supervised learning and testing phases, the classifier can determine the class of unknown or unlabelled new examples. The definition of classification suggests that classifiers and datasets are the two essential inputs in an ML application.

From a dataset perspective, the dependencies on training, validation, and test datasets imply that the insufficient datasets cause low performance even with optimal (*i.e.* well-modelled, robust) classifiers, which is also called “Garbage In, Garbage Out” (GIGO). In one of the earliest highlights of data dependency of any algorithms, Babbage (1864, p. 67) faced a provoking question “if you put into the machine wrong figures, will the right answers come out?” and was puzzled by the confusion of ideas behind it. We can, now, re-phrase the question as “can garbage in *gold* out possible?” Years later, the literature partly took the attention of such practices expecting “gold.” Tweedie (1994), for example, identified the approaches in observational studies (*e.g.*, the effect of passive smoking) as GIGO, where insufficient data was not a primary consideration. GIGO was also an underlined caution in quantitative data analysis in all scientific inquiry (Arcury & Quandt, 1998, p. 73).

The term GIGO was also used in the literature to

- address the practices in clinical research and question whether the researchers can draw conclusions based on a tiny sample (Heuser, 1998).
- express the unreliability of data where scientific research is conducted over operational or administrative datasets (*e.g.*, epidemiologic research on administrative databases, billing systems, maintained by healthcare providers and institutions) (Grimes, 2010, p. 1018).
- present the criticality of data and stress that the approach and methods without avoiding GIGO will be affectless in critical areas such as cancer detection and staging (O’Hurley et al., 2014, p. 784).

In summary, GIGO has turned out to be a colloquial recognition of poor data entry leading to unreliable data output that leads to the necessity of highly accurate, valid, and complete information collection (Kilkenny & Robinson, 2018, p. 103). Contrary to common belief, the sample size is not necessarily an indication of dataset quality or, in other words, “scientific knowledge is impossible with small-sample classification” (Dougherty & Dalton, 2013, p. 1).

Data quality and the need for insight in data are also a prerequisite for knowledge discovery processes, data mining algorithms, and machine-learning workflows as well as handling big data (Hoerl et al., 2014). Triguero et al. (2019), for example, highlights this in big data scope and certain insights can be helpful to transform big data into smart data. Classification is a focused research field in machine learning where several algorithms are proposed. Researchers try to model a robust classifier based on one of those algorithms to apply it to a specific classification problem. Dougherty et al. (2005) adapted the signal theory of optimal robust filters to classifiers and address two types of robust classifiers: minimax and Bayesian robust classifiers. The former is its worst performance over all the states is better than the other classifiers’ worst performances. The latter has an expected performance better than the other classifiers. In the signal-processing theory of robust filtering, the full distributional knowledge and other factors are central such as feature selection and design via training from sample data. Just like a filter to be applied in non-design settings, a classifier (a filter estimating a class label) should exhibit robust performance across a range of conditions. Dougherty concludes that no matter how precisely the classifier is designed, it may not perform well relative to the actual distributions.

2. GIGO *VERSUS* OPTIMAL CLASSIFIERS IN MACHINE LEARNING

Figure 1, which depicts the ML-based classification workflow with its essential activities conducted by the researchers, highlights the effect of the GIGO rationale. This figure has significantly extended and conceptualised the four combinations of GIGO rationale pictured in (El Naqa et al., 2015, fig. 1.2).

Note that the specification of the whole workflow concerning the introduced “space” concept is a distinctive approach of this study considering the literature. In this manner, the specific scope, transformation of a dataset into different forms (sample space, feature space, and subsample space), and the other input (model space) and output (metric space) could be distinguished easily.

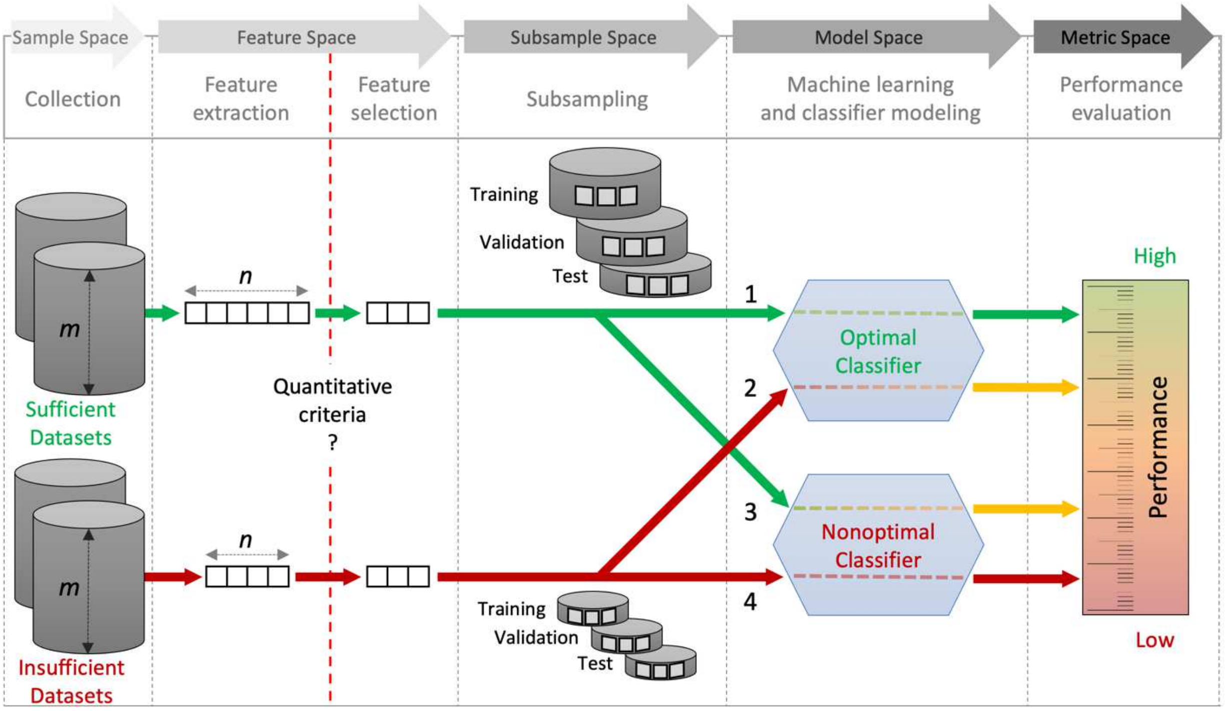


FIGURE 1 I The suggested conceptualisation of ML classification workflow in phases through specific spaces and GIGO rationale (quantitative criteria?)

Figure 1 introduces the four possible combinations of the two inputs: a sufficient/insufficient dataset with an optimal/nonoptimal classifier. It is likely that

- in Case 1 (only win-win scenario): an optimal classifier trained on a sufficient dataset exhibits high performance,
- in Case 2: an optimal classifier trained on an insufficient dataset shows lower performance,
- in Case 3: a nonoptimal classifier trained on a sufficient dataset exhibits low performance, and
- in Case 4 (the worst case): a nonoptimal classifier trained on an insufficient dataset shows the most inadequate performance.

Therefore, dataset sufficiency plays a decisive role. Researchers who build a classifier that is trained and tested on a dataset publish their classification performances in terms of standard metrics such as accuracy, true positive rate, or F1 (Gürol Canbek et al., 2021). The classifiers are compared with other classifiers that are trained and tested on different datasets via the same performance metrics. A few studies compare or analyse the datasets from an ML perspective. For example, Gaugen et al. (2017, fig. 13) compare eight visual image datasets and focus on the distribution of object locations in the image and the ratio of the object size to the image size. They discovered that many dataset labels are centred in the image except datasets having network camera pictures.

Fundamentally, ML research and education focuses on the activities assuming ground-truth or gold standard datasets are already available (Geiger et al., 2021), dataset sufficiency and reliability assurance should be defined, measured, and achieved unconditionally. From a qualitative perspective, several aspects are relevant for assessing the datasets such as their origins, collection methods, the assumptions and conditions (especially for survey datasets), data imputations, and subject matter experts' opinions. Those aspects can differentiate whether the datasets are actively solicited (*e.g.*, surveys or the ones with human-labelling) or passively (*i.e.* as is) acquired (Lew & Schumacher, 2020, pp. 87–89). From a quantitative perspective, ML-based classification, with mostly the supervised-learning approach, first needs to be sure that a dataset exhibits the diversity and representativeness close to reality in the problem domain. Any dataset (*i.e.* sample in statistics) not representing the reality (*i.e.* population in statistics) or the ones capturing only a small part of the reality (biased or not rich datasets) or the ones with errors (low-quality datasets) can be evaluated as a “garbage” in GIGO rationale. Garbage can also be *injected* into datasets unintentionally or deliberately:

- Unintentional garbage, which is occurred not only by injection but also by omission, causes a biased dataset. Such biases can be observed even in widespread ML applications such as face recognition. In a recent study, NIST conducted a test of 189 mostly commercial algorithms on 18 million images of 8.5 million people contained in four large photograph datasets collected in U.S. governmental applications such as visa or border crossing (Grother et al., 2019, pp. 1–3). The results indicating severe bias showed that false positives (erroneous association of samples of two persons) are highest in West and East African and East Asian people and lowest in Eastern European people. False positives are also higher in women than men and oldest/youngest people than middle-aged adults. False negatives (failure to associate one person in two images) are higher in Asian and American Indian people than white and African American individuals in one dataset and higher in African and the Caribbean as well as in older individuals in another dataset. The report clearly shows that the classifiers trained/tested on biased datasets that do not reflect the natural distribution of the real-case instances cause misclassifications, which can be expressed as “garbage” by people who are not well-represented.
- Deliberate injections or techniques are called damages or attacks that are referred to as adversarial ML as a discipline (Joseph et al., 2019, Chapter 3). The attacks that are called poisoning in the training phase and evasion in test or production phases and defence against those attacks have recently been studied in the literature (Biggio & Roli, 2018, p. 318). For example, Mahloujifar et al. (2019) studied those attacks in image classification where a new instance looks similar to an existing instance and suggested that a “concentration of measure” depending on the distribution of the test dataset can indicate the robustness to adversarial perturbations.

Hence, the frequency distribution of binary features is introductory but one of the first mathematical insights of ML known-labelled datasets, a quantitative summary. Therefore, some *cross-checking* assessments of feature distribution might help to sense garbage to examine further. Such exploratory analysis can be conducted in two ways:

- Inter-class: Distribution differences between positive and negative class datasets in a specific classification application, and
- Intra-class: Distribution differences between two or more datasets with the same class (*e.g.*, positive-class datasets) in a specific classification problem domain).

Note that inter-class and intra-class comparisons are known techniques (*i.e.* maximum inter-class deviation and minimum intra-class variation) in clustering and feature selection (Asfour et al., 2021, fig. 1; Sahu et al., 2017, p. 110). However, the goal of these assessments does not help feature selection directly. Because feature selection should take place after ensuring the dataset sufficiency. No matter how effective feature selection is, it should be based on a sufficient dataset. Such exploratory analysis should be conducted before proceeding with selecting features and building a classifier model.

Note that some statistical methods are already used to describe datasets (sample space size, feature space size, class ratios etc.). The statistics related to the shape of the feature distribution, such as skewness, kurtosis, and the number of peaks, can also be analysed (Piringer et al., 2008, p. 242). However, those statistical approaches summarise a dataset based on a single attribute that is usually continuous. Nevertheless, interpreting and comparing statistical figures alone are not convenient; besides, they are generally not suitable for discrete or qualitative features.

Knowing the dataset feature-space distribution characteristics and comparing it with the ones used in other datasets in the same domain (*i.e.* intra-class assessment) can present the nature of the data and provide high-level situational awareness about the datasets, samples, and the contents. Binary features are simple yet common in today’s datasets. Recent practices also emphasize the high effectiveness of binary features (Chen et al., 2021). A researcher who has a specific distribution in her/his dataset that is different from the ones in the same domain should examine further why and how it is different or whether it is a biased dataset. Hence, the dataset could be sufficient. On the other hand, researchers who wish to enrich their datasets usually merge new datasets they acquired from other sources without analysing. They could not be sure how these datasets are different from the existing ones. Proceeding the ML workflow (*e.g.*, feature selection) without analysing such aspects may lead to unrealistic or ungrounded classification models as in the given examples above. Note that because feature count or frequency is needed to find a distribution fit, it is not a costly operation (a single pass in the database with addition operations). Establishing rather generic quantitative analysis can lead to qualitative analysis of the datasets and help to enhance the benchmarking datasets in specific domains (Gürol Canbek et al., 2018).

3. FITTING BINARY-FEATURE FREQUENCIES INTO A STATISTICAL DISTRIBUTION

The feature-space frequency distribution is the frequency of binary features occurrence sorted in decreasing order. Such distributions generally follow size or frequency trends that are intuitively stated as “trivial many and vital few” or “useful many and vital few” (Juran & Godfrey, 1999, pp. 5.27-5.28). Specifically, four nonnormal long right-tail statistical distributions are described in the literature (Joo et al., 2017): log-normal, exponential, power law, and Poisson. Poisson distribution is usually the distribution of “count data” that shows the counts (non-negative) of a single (or combination of) dependent variable(s). Because we have the counts per feature in feature space here, Poisson or similar distributions like binomial or gamma-count distributions should not be expected to fit.

3.1 Example distributions found in natural and unnatural phenomena

Power law, which is also known as the 80:20 rule or Pareto principle, exponential, and log-normal distributions are addressed in the literature that tends to search for a specific statistical distribution to find out the characteristics of many natural and unnatural phenomena. The example phenomena reflecting these distributions are compiled and categorised from (Limpert et al., 2001; Milojević, 2010; Newman, 2004; White et al., 2008) or other resources with given references as follows

- *Natural phenomena fitting log-normal*: elements concentration in the Earth’s crust, latent periods (from infection to the first symptoms) of infectious disease (*e.g.*, the incubation period of Coronavirus disease 2019 (COVID-19), SARS (severe acute respiratory syndrome), and MERS (Middle East Respiratory Syndrome), ((Backer et al., 2020, sec. Table 3)), the abundance of bacteria on plants;

- *Unnatural phenomena fitting log-normal*: number of letters per word, number of words per sentence, age of first marriage in Western;

- *Natural phenomena fitting exponential*: damage in nuclear power incidents and accidents before 1980 (Wheatley et al., 2017, p. 6), moderate-sized disasters (observed sea-level variations, wind velocity, annual river floods) (Pisarenko & Rodkin, 2010, p. 6), the arrival rate of cosmic ray alpha particles or Geiger counter tics (Tobias, 2012, sec. 8.1.6.1);

- *Unnatural phenomena fitting exponential*: time to failure patterns (also in natural phenomena) (Frank, 2009, pt. 5), modelling malware propagation delays (Wang & Murynets, 2013, pp. 43–44), frequency of Korean family names (power law in family names in the world), intervals between aircraft arrivals to major airports (Willemain et al., 2004, p. 5), the inter-arrival times of the 911 calls (Albert, 2011), the time between goals in World Cup soccer matches (Chu, 2003, pp. 65–66), the dispersion of U.S. incomes which was qualified as a kind of thermal equilibrium (Bartels, 2012, p. 17);

- *Natural phenomena fitting power law*: island sizes, lake sizes, flood magnitudes, species body sizes, individual body sizes (White et al., 2008, p. 905); and

- *Unnatural phenomena fitting power law*: author productivity, citations received by papers, scattering of scientific literature (Milojević, 2010, p. 2418).

The above phenomena are provided to introduce the statistical distributions by examples revealing their diversities and to allow the researchers to relate them with the distributions observed in their datasets.

3.2 Methods for testing distribution fits

A statistical distribution could fit a given distribution (*i.e.* truth) for the values (x, binary-features in our case) greater than or equal to a minimum value ($x_{\min}$). For each statistical distribution to be fit the truth, an algorithm first estimates a minimum value by minimising the Kolmogorov-Smirnoff statistics (Clauset et al., 2009, p. 11) and then estimates the statistical distribution parameters. The two tests were used to validate the plausibility of the estimated fits, namely power law, log-normal, and exponential statistical distributions:

- Bootstrap test for each estimated distribution yielding goodness-of-fit test and pl-value (plausibility value) and
- Vuong’s test yielding total likelihood-ratios, pl-value (1-sided), and (2-sided) for comparing the first candidate distribution fit against the second fit with its $x_{\min}$ is equal to the first fit’s $x_{\min}$. For example, a log-normal fit estimated with a specific $x_{\min}$ value is denoted as ln*. In contrast, a power law fit that is calculated based on the same $x_{\min}$ value is represented as pl. The comparison is expressed as pl *vs* ln*.

In the bootstrap test, the pl-value indicates the plausibility of the given statistical distribution by simulating multiple instances of the truth and re-inferring the fitted distribution (Gillespie, 2015, pp. 4–5). In Vuong’s test, the Kullback-Leibler information criterion is used to measure the closeness of the given two statistical distributions to the truth in a likelihood-ratio statistics with testing the hypothesis that they are equally close (Vuong, 1989, p. 308). Besides interpreting the sign of the goodness-of-fit test value specifying which statistical distribution has a better fit (positive for the first and negative for the second statistical distribution,) the following pl-values are provided:

- pl-value (1-sided) indicates the plausibility of the better statistical distribution if it exists, and
- pl-value (2-sided) shows whether both distributions are equally close or far from the truth.

3.3 Online supplementary experimentation platform, software, and datasets

Finding whether a statistical distribution fits into a given distribution should be examined, possibly by verifying different methods. Hence, a software library of the comprehensive set of statistical tests was initially implemented to assess the fit of various statistical distributions in R (a software environment for statistical computing and graphics) based on poweRlaw package (Gillespie, 2015).

The online materials provided are as follows:

3.2.1 Online experimentation platform (https://codeocean.com/capsule/3624528/tree/v1)

The reproducible detailed results can be obtained via an online experimentation capsule in CodeOcean. No programming is required.

3.2.2 The dataset feature-frequency distributions fitting software and datasets (https://github.com/gurol/DsFeatFreqDistFit)

The open-source code implementation, the datasets (in open office spreadsheet and R data format), and other supplementary materials (for example, the charts and tables provided in this study and the complete dump of the distribution fit tests) are available online to review them in detail and use them in your works.

3.4 Case study

Mobile malware classification problem was chosen as a case study domain because it is a critical emerging cyber security field where ML-based classification approaches are highly studied and practised in the literature and the industry to enhance the capacities related to the human factor (Andrade & Yoo, 2019, p. 3). The method is verified by a demonstration that examines and compares negative (benign) datasets and positive (malign) used in various binary classification (malware classification) studies based on binary features (application permission requests) as summarised in Table 1.

TABLE 1 | The aspects of case-study demonstrating dataset comparison

Binary Classification	Case study
Classification problem (domain)	Android mobile malware classification
Examples (samples)	Android mobile applications
Negative class label	"Benign" application
Positive class label	"Malign" application or "Malware"
(Binary) features	Android application permission requests (shortly 'application permissions' or 'permissions')
Example feature	CALL_PHONE: It allows an application to initiate a phone call without going through the Dialler user interface for the user to confirm the call.
Binary feature values	0: The permission is not requested by the application (default) 1: The permission is requested by the application
Missing values	Datasets might have a missing value (*i.e.* they do not have at least one sample (application) with the specific binary feature). Such features are taken as default 0 (not allowed) in dataset comparisons.
Number of features (*n*)	Minimum: 69 and maximum: 118
Compared datasets	Five pairs (negative/positive class) of datasets (DS_0, DS_1, DS_2, DS_3, and DS_5) and one positive-only dataset (DS_4).

Permissions are credentials requested by Android applications before they can use specific system data and facilities such as sending SMS. They are binary flags for Android platforms' primary access control to provide privacy and secure data/information in mobile devices. Many studies in Android malware classification include examining the permission feature frequency distribution in their datasets (Gürol Canbek et al., 2017, pt. Table 3). Although they highlight the dramatic decrease in the frequencies, the distribution that provides valuable insight to qualify the datasets has not been analysed before. Interestingly, the distributions seem to exhibit common characteristics for all the datasets at first glance.

3.5 Case study datasets

Although the literature proposes several approaches to detect Android mobile malware, the datasets are not as diverse as them (Gürol Canbek et al., 2018). Table 2 lists the basic quantitative information for the datasets and introduces the related studies. The two dimensions, namely sample-space size (*m*) and feature-space size (*n*) and prevalence (*PREV*)[1] values are listed. In the related literature, it is observed that authors compare their malware classification performance with others, most of which are based on different benign and malign datasets. The case study can help to gain insights into those datasets.

This study reviewed eleven academic studies providing Android mobile benign and malign datasets listed in Table 2 and selected six datasets for comparison. The DS_0 dataset listed in the first row in Table 2 has not only a higher number of samples but also the highest number of malware (positive-class examples) compared with other datasets. Note that two published datasets were combined, one from 2011 and one from 2012 (Peng et al., 2012) into one dataset (DS_5).

[1] The proportion of total positive samples (m_P), *e.g.*, having a malign characteristic, in total sample size [$m_P + m_N$]

Introduction to Android Mobile-Malware Classification Problem:

Android is a mobile platform that provides a large number and a wide range of mobile applications. Android applications are developed by anyone and released on third-party application markets besides the official market named Google Play. Despite this diversity, the platform could be the target of malicious people who develop or make injections into existing applications that exposes some risks against end-users. Malware authors develop and use different techniques in those applications appearing as legitimate to overcome the platform’s security or exploit human factors. Therefore, mobile malware detection, which is labelling a given application as ‘benign’ (‘negative’) or ‘malign’ (‘positive’, also known as ‘malware’), is one of the urging areas to be studied by the security sector and academia. Experts examine the applications manually with the help of specialised tools (*e.g.*, reverse engineering software) and decide whether they are benign or malign. This human involved process is called malware analysis. In addition to dynamic malware analysis that concentrates on applications’ behaviours observed at run-time, static malware analysis examines binaries, files, and codes to classify Android malware from benign applications.

Android’s permission mechanism limits the specific operations performed by applications or provides *ad hoc* access to particular data at the end-user's discretion. Suppose an application is required to initiate a phone call without going through the standard dialler user interface for the user to confirm the call, for example. In that case, it must manifest or request CALL_PHONE permissions. Please, refer to Android API (Application Programming Interface) documentation for the list of the permissions and their descriptions at https://developer.android.com/reference/android/Manifest.permission.html. More information can also be found in (Gurol Canbek, 2021, sec. Appendix B)

The six datasets (DS_6 – DS_{11}) encountered in the literature were excluded from this study due to the following reasons. The DS_9 dataset (Peiravian & Zhu, 2013) is the same as the original DS_4 dataset (Jiang & Zhou, 2013). The datasets DS_8 (Canfora et al., 2013), DS_{10} (Felt et al., 2011) have missed one class. Only the top ten permissions were published for DS_6 (Hoffmann et al., 2013), and only the top 20 permissions were published for DS_7 (Sarma et al., 2012), but the whole feature space could not be obtained for this study.

TABLE 2 | *Case study datasets: Summary of sample and feature spaces of the benign (negative) and malign (positive) dataset.*

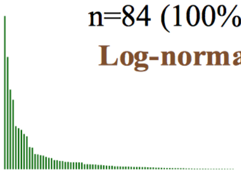
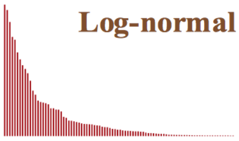
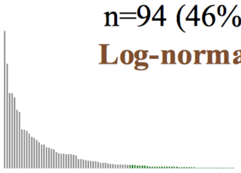
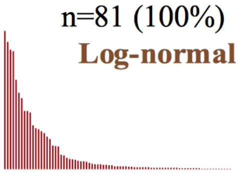
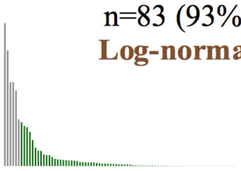
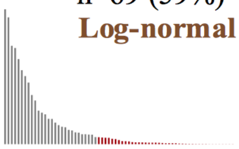
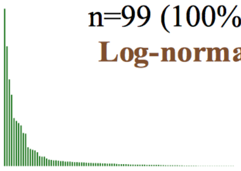
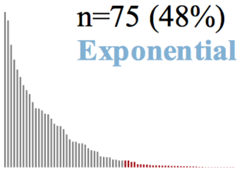
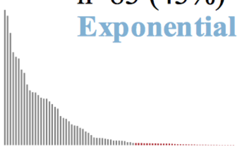
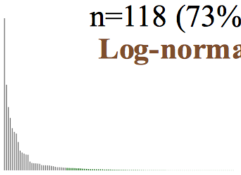
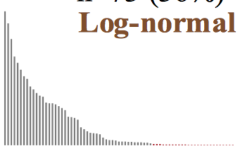
			Sample space			Feature space	
Dataset	Name	Authors and reference	m_N	*PREV*	m_P	n_N	n_P
DS_0	ANDRUBIS	(Lindorfer et al., 2014)	264,303	60%	399,353	84	90
DS_1	Contagio	(Aswini & Vinod, 2014)	254	52%	280	94	81
DS_2		(Wang et al., 2014)*	310,926	2%	4,868	83	69
DS_3		(Yerima et al., 2014)*	1,000	50%	1,000	99	75
DS_4	Android Malware Genome Project	(Jiang & Zhou, 2013)		100%	1,260		83
DS_5		(Peng et al., 2012)	207,865	0.2%	378	118	73
-DS_6		(Hoffmann et al., 2013)	136,603		6,187		
-DS_7	Contagio	(Sarma et al., 2012)	158,062		121		
-DS_8		(Canfora et al., 2013)			400		
-DS_9		(Peiravian & Zhu, 2013)*	1,250		1,260		
-DS_{10}		(Felt et al., 2011)	900				

* The positive-class datasets contain AMGP samples.

3.6 Initial analysis of the feature-space frequency distributions

Table 3 shows the feature-space frequency distribution graph per dataset for each class and the most plausible fit distributions with (*ntail* ratio, *i.e.* fitted features ratio[2]), which are described below. As seen in the related mini graphs for each dataset in Table 3, all the samples demonstrate a long right tail. The right tail holds rarely requested permissions (low-amplitude in graphs) while dominating the short-left part holds frequently requested permissions (high-amplitude in graphs).

TABLE 3 I Comparison of eleven negative and positive-class datasets’ feature-space frequency distributions with mini graphs and the name of the most plausible distribution fits (*n* is the feature-space size and the values in braces are *ntail*).

m	Negative	Datasets	*m*	Positive
264,303	 n=84 (100%) Log-normal	DS_0	399,353	 n=90 (100%) Log-normal
254	 n=94 (46%) Log-normal	DS_1	280	 n=81 (100%) Log-normal
310,926	 n=83 (93%) Log-normal	DS_2	4,868	 n=69 (59%) Log-normal
1,000	 n=99 (100%) Log-normal	DS_3	1,000	 n=75 (48%) Exponential
		DS_4	1,260	 n=83 (43%) Exponential
207,865	 n=118 (73%) Log-normal	DS_5	378	 n=73 (36%) Log-normal

Another interesting finding in Table 3 is related to inter-class analysis (*i.e.*, analysing positive-class *versus* negative-class datasets). Comparing permission frequency distribution curves per dataset per

[2] The long right-tail statistical distribution is evident only in the tail ($x_{\min}$ > x). Poisson decays exponentially, power law decays polynomially. The sharp drop is not surprising whereas the long right-tail is surprising.

class, we can see that the frequencies of malign dataset' curves get flattered later (decay is small) comparing the benign dataset (sharp drop). The reason behind this sharp drop in benign dataset regular applications uses a few specific permissions, whereas malware needs to request a more comprehensive set of permissions. Figure 2 shows the distributions for all the datasets in one graph per class, where the *y*-axis is transformed into a logarithmic scale. Be aware that the *x*-axis is not the same feature sequence naturally.

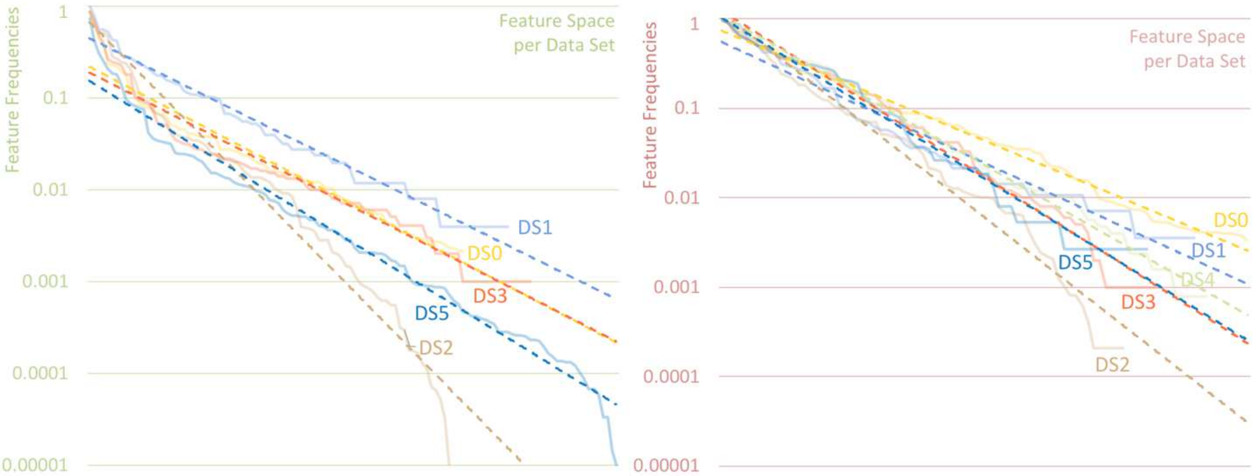


FIGURE 2 | Frequency distribution in a log scale (*y*-axis) of the feature space (*x*-axis) per dataset for each class. Dashed lines are added to see the linear trend.

However, the inferences from the graphs may be misleading. They should be verified from a statistical point of view (Newman, 2004, sec. II) because the distributions could –or actually could not be– one of the statistical distributions, namely power law, log-normal, Poisson, and exponential distributions. If a dataset's features fit into a statistical distribution, the distribution defines the characteristics or nature of the feature space in the dataset. It provides more analysis possibilities on inter-class, intra-class, and extra-class because each statistical distribution has its own theoretical and practical implications for interpreting (Joo et al., 2017, pt. Table 1).

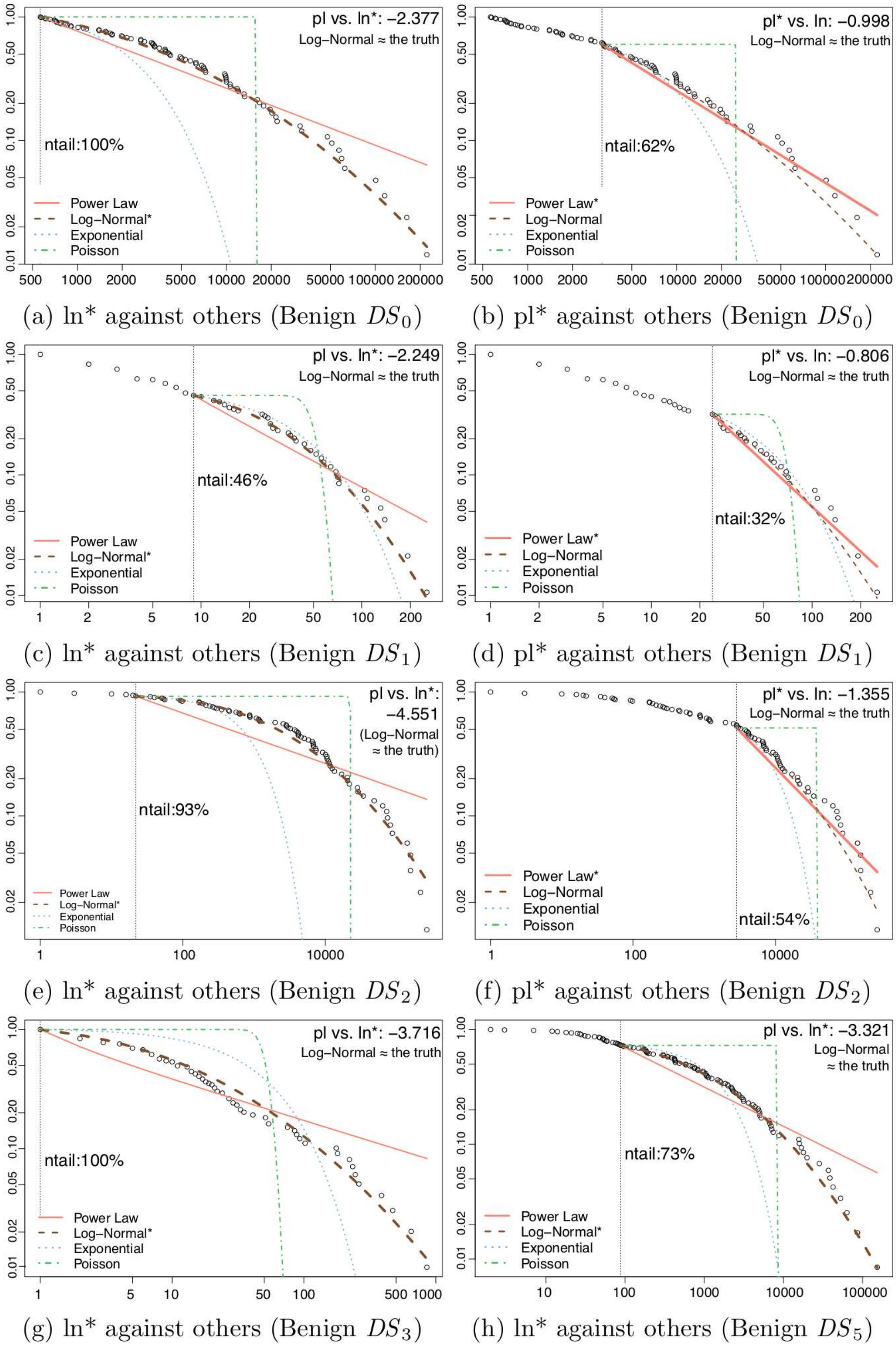
4. RESULTS

Figure 3 shows the charts for benign datasets, whereas Figure 4 for malign datasets, all of which are generated by an R script provided online (DsFeatFreqDistFit.R, see subsection 3.2.2). For your comparison, the charts for the best plausible statistical distribution fit and the second plausible one are provided for the same dataset. Figure 3 (a) and Figure 3 (b) are the log-normal distribution as the best fit and the power law distribution as the second-best fit, respectively, for benign DS_0.

Because frequency distributions of a wide variety of phenomena tend to be, at least approximately, power law distribution (White et al., 2008, p. 905), it was expected that feature frequencies would fit power law (at least benign/negative datasets due to their naturalness comparing the malign/positive datasets that have features required by malicious purposes). However, the results were different.

Both Figure 3 and Table 4 shows that all the benign datasets exhibit log-normal distribution. The benign fits are valid in high *ntail* percentages except for the benign DS_1 dataset with only 254 samples. The second plausible fits are power law distribution for all the datasets. As stated above, Poisson distribution is not a plausible fit for the feature frequency distribution of any datasets no matter what the class is. The feature frequency distributions of malign datasets are different from each other. Unlike benign datasets, malign DS_3 and DS_4 exhibit exponential distribution, and DS_2 are also close to exponential distribution (higher *ntail* ratio but lower pl-value).

This should not be considered as a generalised rule statement. However, an exciting finding revealed that considering the example phenomena above, log-normal and partly power law (except earthquakes, solar flares, and war intensities, for example) distributions represent the stable phenomena like benign applications’ permission request distribution.



(a) ln* against others (Benign DS_0)

(b) pl* against others (Benign DS_0)

(c) ln* against others (Benign DS_1)

(d) pl* against others (Benign DS_1)

(e) ln* against others (Benign DS_2)

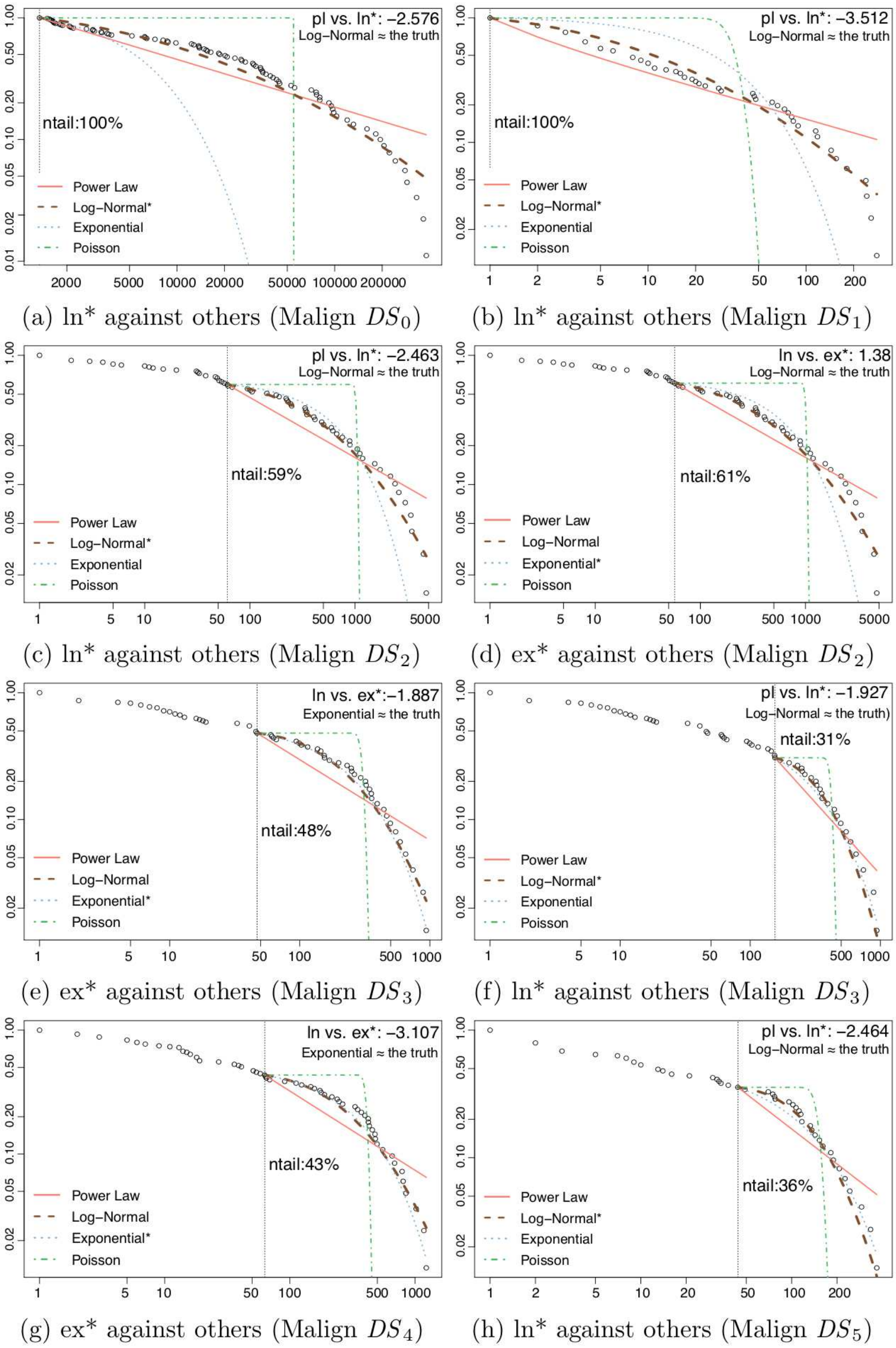
(f) pl* against others (Benign DS_2)

(g) ln* against others (Benign DS_3)

(h) ln* against others (Benign DS_5)

FIGURE 3 | Feature frequency distribution and power law, log-normal, exponential, and Poisson fits per benign dataset (for *DS*$_0$, *DS*$_1$, and *DS*$_2$: left charts: the best fit, right charts: the second-best fit). *X*-axis: feature counts, *y*-axis: ranks (generated via the DsFeatFreqDistFit.R script).

In contrast, exponential describes somewhat chaotic phenomena like malign applications’ permission request distribution. Another finding is that the high number of samples tends to exhibit log-normal distribution.



(a) ln* against others (Malign DS_0)

(b) ln* against others (Malign DS_1)

(c) ln* against others (Malign DS_2)

(d) ex* against others (Malign DS_2)

(e) ex* against others (Malign DS_3)

(f) ln* against others (Malign DS_3)

(g) ex* against others (Malign DS_4)

(h) ln* against others (Malign DS_5)

FIGURE 4 | Feature frequency distribution and power law, log-normal, exponential, and Poisson fit per malign dataset (for DS_2 and DS_3: left charts: the best fit, right charts: the second-best fit). *X*-axis: feature counts, *y*-axis: ranks (generated via the DsFeatFreqDistFit.R script).

Although it is not focused on in the literature, the number of fitted features were also taken into account because the fitted features and not-fitted features provide more insight into the datasets and classes. Table 4 shows the summary of the analysis of plausibility of permission feature frequency distribution fits into log-normal, exponential, and power law statistical distributions. “*ntail*” indicates the ratio of fitted features to the number features (*n*) as a percentage in the dataset. Examining the results, benign datasets have higher *ntail* ratios compared the malign datasets.

TABLE 4 I Plausibility of permission feature frequency fits into log-normal (ln), exponential (ex), and power law (pl) statistical distributions

Class	Datasets	*m*	*Plausibility Degree*	Plausible (1st) and Alternative (2nd, 3rd) Distribution Fits			Bootstrap Test		Vuong's Test on Two Distributions			
		n		Distribution	*ntail*	Parameter(s)	GoF	pl-value	1st *vs.* 2nd	Statistics	pl-value (1-sided)	pl-value (2-sided)
Negative (Benign)	*DS*$_0$	264K	1st	Log-normal	100%	μ=7.57, σ=2.04	0.05	0.92	pl *vs.* ln*	-2.38	0.99	0.02
			3rd	Exponential	7%	λ=0.00001	0.35	0.00	ln *vs.* ex*	0.34	0.37	0.73
		84	2nd	Power law	62%	α=1.75 (typical)	0.12	0.15	pl* *vs.* ln	-1.00	0.84	0.32
	DS$_1$	254	1st	Log-normal	46%	μ=3.23, σ=1.08	0.07	0.20	pl *vs.* ln*	-2.25	0.99	0.02
			3rd	Exponential	34%	λ=0.02079	0.09	0.49	ln *vs.* ex*	0.42	0.34	0.67
		94	2nd	Power law	32%	α=2.22 (typical)	0.10	0.49	pl* *vs.* ln	-0.81	0.79	0.42
	DS$_2$	311K	1st	Log-normal	93%	μ=7.82, σ=2.52	0.06	0.50	pl *vs.* ln*	-4.55	1.00	0.00
			3rd	Exponential	19%	λ=0.00002	0.83	0.00	ln *vs.* ex*	0.49	0.31	0.63
		83	2nd	Power law	54%	α=1.59 (typical)	0.10	0.43	pl* *vs.* ln	-1.36	0.91	0.18
	DS$_3$	1K	1st	Log-normal	100%	μ=2.05, σ=2.00	0.06	0.24	pl *vs.* ln*	-3.72	1.00	0.00
			3rd	Exponential	12%	λ=0.00434	0.14	0.58	ln *vs.* ex*	-0.83	0.80	0.40
		99	2nd	Power law	54%	α=1.71 (typical)	0.08	0.24	pl* *vs.* ln	-0.93	0.82	0.35
	DS$_5$	208K	1st	Log-normal	73%	μ=6.81, σ=2.20	0.04	0.86	pl *vs.* ln*	-3.32	1.00	0.00
			3rd	Exponential	5%	λ=0.00002	0.63	0.00	ln *vs.* ex*	0.53	0.30	0.60
		118	2nd	Power law	36%	α=1.66 (typical)	0.09	0.41	pl* *vs.* ln	-1.14	0.87	0.25
Positive (Malign)	*DS*$_0$	399K	1st	Log-normal	100%	μ=8.94, σ=2.20	0.06	0.69	pl *vs.* ln*	-2.58	1.00	0.01
			3rd	Exponential	20%	λ=0.00001	0.47	0.00	ln *vs.* ex*	0.48	0.31	0.63
		90	2nd	*Power law*	99%	α=1.40 (typical)	0.14	0.74	pl* *vs.* ln	-2.34	0.99	0.02
	DS$_1$	280	1st	Log-normal	100%	μ=1.78, σ=2.03	0.07	0.10	pl *vs.* ln*	-3.51	1.00	0.00
			3rd	Exponential	30%	λ=0.01130	0.08	0.91	ln *vs.* ex*	-1.46	0.93	0.14
		81	2nd	*Power law*	86%	α=1.47 (typical)	0.09	0.11	pl* *vs.* ln	-2.18	0.99	0.03
	DS$_2$	4.9K	1st	Log-normal	59%	μ=5.99, σ=1.44	0.06	0.79	ex *vs.* ln*	-1.35	0.91	0.18
			2nd	*Exponential*	61%	λ=0.00125	0.13	0.12	ln *vs.* ex*	1.38	0.08	0.17
		69	3rd	Power law	46%	α=1.76 (typical)	0.12	0.19	pl* *vs.* ln	-1.22	0.89	0.22
	DS$_3$	1K	2nd	Log-normal	31%	μ=5.83, σ=0.57	0.07	0.93	pl *vs.* ln*	-1.93	0.97	0.05
			1st	Exponential	48%	λ=0.00396	0.08	0.81	ln *vs.* ex*	-1.89	0.97	0.06
		75	3rd	Power law	20%	α=3.32 (atypical)	0.13	0.63	pl* *vs.* ln	-0.53	0.70	0.60
	DS$_4$	1.3K	2nd	Log-normal	30%	μ=6.07, σ=0.59	0.08	0.66	pl *vs.* ln*	-2.02	0.98	0.04
			1st	Exponential	43%	λ=0.00293	0.09	0.47	ln *vs.* ex*	-3.11	1.00	0.00
		83	3rd	Power law	22%	α=3.06 (atypical)	0.13	0.46	pl* *vs.* ln	-0.78	0.78	0.43
	DS$_5$	378	1st	Log-normal	36%	μ=4.83, σ=0.58	0.07	0.93	ex *vs.* ln*	-0.50	0.69	0.61
			2nd	*Exponential*	32%	λ=0.01127	0.09	0.85	ln *vs.* ex*	-0.34	0.63	0.74
		73	3rd	Power law	26%	α=3.01 (typical)	0.11	0.74	pl* *vs.* ln	-0.76	0.78	0.45

The tabular presentation is useful to see the corresponding values all at once, but how the distributions fit into the truth may not be sensed easily. Therefore, compact charts were prepared to show the original feature frequency distribution (the truth) and the plausible statistical distribution.

5. FITTED/UNFITTED FEATURES ANALYSIS

In this study, further analysis was conducted on the fitted where $x \geq x_{\min}$ and unfitted features where $x < x_{\min}$. Figure 5 shows the result of our analysis of permission feature spaces for all the benign and malign datasets.

There are three sets in Figure 5:

- The green one is the intersection of fitted features of the benign datasets with 38 common features
- The red one is the same for malign datasets with 22 common features

- The orange one is the intersection of unfitted features of the malign datasets with 14 common features

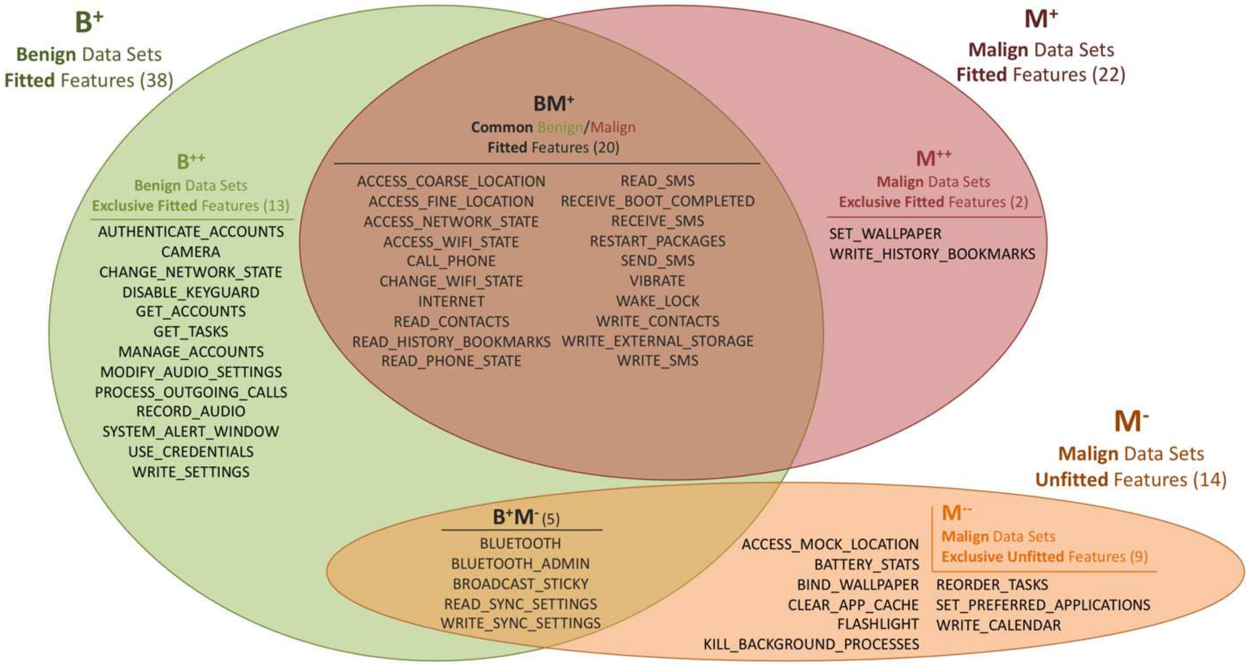


FIGURE 5 | Venn diagram for fitted/unfitted features. The numbers in braces show the feature counts. “+” superscript denotes fitted features whereas “-” denotes unfitted ones.

Note that the intersection of unfitted features of the benign datasets is empty in our case. Fitted or unfitted features are determined in the most plausible statistical distribution per each dataset. The DsFeatFreqDistFit.R script finds and displays the fitted and unfitted features per dataset. The features are then intersected per class by using another functionality (getCommonFeatures) in the provided package. The features are also provided in the extra materials provided online at https://github.com/gurol/dsfeatfreqdist.

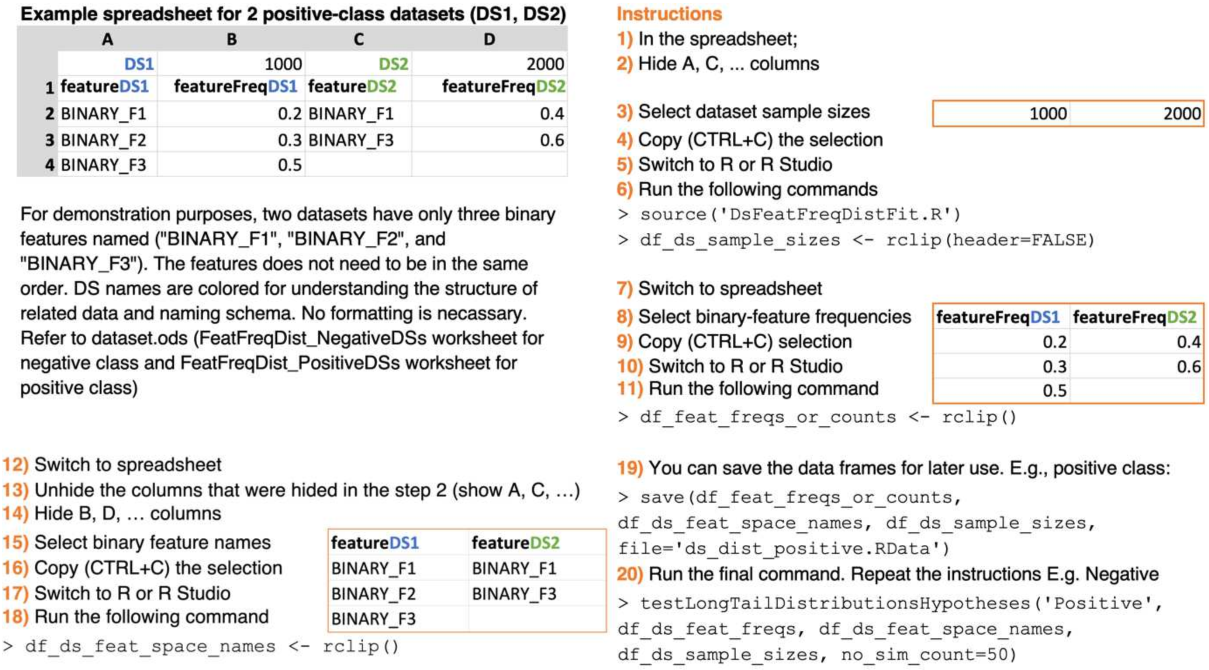
This approach could be useful for gaining insight into feature space, especially regarding the inter dataset and inter-class analyses, which could be valuable for feature selection and dataset comparison activities. For binary classification that is classifying benign and malign Android mobile applications, the discriminative features having inter-class differences should be taken into account while performing the activities. Therefore, the features in M++, B++, and M-- parts could be discriminative or be evaluated on a new type of malware.

Considering malware detection, the M++ and M-- features should be examined first among 122 permission features. For the sake of saving space, our initial interpretations highlighted the following features to report in this study: in M++: WRITE HISTORY BOOKMARKS is dangerous type permission, in M--: ACCESS MOCK LOCATION, CLEAR APP CACHE, and WRITE CALENDAR are dangerous, SET PREFERRED APPLICATIONS has been deprecated since Android API 7, BATTERY STATS and BIND WALLPAPER is a signature or system type permissions. The other features are standard type permissions.

Determining the distribution of the features could be useful for other activities. For instance, the geometric mean should be used to determine the central tendency of a log-normal distribution instead of arithmetic means.

Prepare your data and run your experiment using the provided script

To reproduce the results presented in this manuscript or conduct a similar experiment for your datasets. Please, prepare a spreadsheet and run the commands in R or RStudio according to the instructions given below. For more information and downloading the script files (DsFeatFreqDistFit.R and utils.R) visit https://github.com/gurol/DsFeatFreqDistFit.



Example spreadsheet for 2 positive-class datasets (DS1, DS2)

	A	B	C	D
	DS1	1000	DS2	2000
1	featureDS1	featureFreqDS1	featureDS2	featureFreqDS2
2	BINARY_F1	0.2	BINARY_F1	0.4
3	BINARY_F2	0.3	BINARY_F3	0.6
4	BINARY_F3	0.5		

For demonstration purposes, two datasets have only three binary features named ("BINARY_F1", "BINARY_F2", and "BINARY_F3"). The features does not need to be in the same order. DS names are colored for understanding the structure of related data and naming schema. No formatting is necassary. Refer to dataset.ods (FeatFreqDist_NegativeDSs worksheet for negative class and FeatFreqDist_PositiveDSs worksheet for positive class)

Instructions

1) In the spreadsheet;
2) Hide A, C, ... columns

3) Select dataset sample sizes

1000	2000

4) Copy (CTRL+C) the selection
5) Switch to R or R Studio
6) Run the following commands

```
> source('DsFeatFreqDistFit.R')
> df_ds_sample_sizes <- rclip(header=FALSE)
```

7) Switch to spreadsheet
8) Select binary-feature frequencies
9) Copy (CTRL+C) selection
10) Switch to R or R Studio
11) Run the following command

featureFreqDS1	featureFreqDS2
0.2	0.4
0.3	0.6
0.5	

```
> df_feat_freqs_or_counts <- rclip()
```

12) Switch to spreadsheet
13) Unhide the columns that were hided in the step 2 (show A, C, …)
14) Hide B, D, … columns
15) Select binary feature names
16) Copy (CTRL+C) the selection
17) Switch to R or R Studio
18) Run the following command

featureDS1	featureDS2
BINARY_F1	BINARY_F1
BINARY_F2	BINARY_F3
BINARY_F3	

```
> df_ds_feat_space_names <- rclip()
```

19) You can save the data frames for later use. E.g., positive class:

```
> save(df_feat_freqs_or_counts,
df_ds_feat_space_names, df_ds_sample_sizes,
file='ds_dist_positive.RData')
```

20) Run the final command. Repeat the instructions E.g. Negative

```
> testLongTailDistributionsHypotheses('Positive',
df_ds_feat_freqs, df_ds_feat_space_names,
df_ds_sample_sizes, no_sim_count=50)
```

Note that `no_sim_count` dramatically increases the time to complete the code. Use 3, for example, for the first attempts. The commands above and comments can be found in DsFeatFreqDistFit.R file.

Conclusion

In the shade of the Garbage In, Garbage Out (GIGO) rationale, high classification-performance could be possible only when an optimal (well-modelled or robust) classifier is trained on sufficient datasets. This requirement is especially crucial in domains where proper benchmark datasets are not available. This study suggests that one of the initial insights (before conducting an ML workflow, *e.g.*, feature selection) into the sufficiency of datasets in ML-based classification studies is quantifying binary-feature space distribution of datasets. Hence, the distribution of binary features is essential to gain initial insight.

The approach has been tested on eleven Android malware/benign application datasets in the literature, and the results are interpreted. This study with an in-depth look at the feature space distribution provides insight into datasets by examining their similarity to various statistical distributions. Interestingly, it was observed that the features in our example benign/malign application datasets exhibit a long right tail (holding rare features while dominating the short left part keeping frequent features). Therefore, we can look for the log-normal, exponential, power law, and Poisson statistical distributions fit. In eleven experimental datasets, all the benign datasets exhibit log-normal distribution against exponential, power law, and Poisson. In malign datasets, the higher plausibility of exponential distributions was observed. The two malign datasets are fit to exponential distributions. Considering the findings, the distribution of feature space among the samples in a dataset should also be analysed to see whether it is close to precisely one of the probability distributions. If there is, the fitted statistical distribution should be provided as an informative meta-feature with the distribution parameters. The parameters are *ntail* ratio for all types of distribution fits, mean and standard deviation parameters for log-normal fit, rate parameter for exponential fit, or alpha parameter (also known as the exponent or scaling parameter) for power-law fit. The plausibility of the test results like in Table 4 could also be provided for further information that would be a good habit of avoiding publication bias.

This study is also a reference for ML studies in terms of providing the natural and unnatural phenomena exhibiting the power law, log-normal, and exponential statistical distributions. The compiled examples give hints about the observed feature distribution fits. In this regard, it is found that benign application’s permission requests follow log-normal distributions, sort of stable phenomena. In contrast, malware’s permission requests tend to follow exponential distributions, relatively chaotic phenomena. This similarity found in the Android mobile platform could be looked for the benign/malign software on other mobile and desktop platforms. The initial findings represented in Figure 5 could be evaluated further by Android malware domain experts as well as ML researchers in feature selection, and the approach could be followed in other domains. Taken together, this study highlights that such exploratory analyses should be involved more in ML studies. It demonstrates the method along with the ready-to-use open-source scripts and comprehensive accompanying materials to the researchers who come from different disciplines and are uninformed about possible statistical usage.

Acknowledgements

The author thanks the researchers for providing the permission frequencies in their datasets.

References

Albert, L. (2011). *Is anything really exponentially distribute?* https://punkrockor.com/2011/02/10/is-anything-really-exponentially-distributed/

Andrade, R. O., & Yoo, S. G. (2019). Cognitive security: A comprehensive study of cognitive science in cybersecurity. *Journal of Information Security and Applications*, *48*, 1–13. https://doi.org/10.1016/j.jisa.2019.06.008

Arcury, T. A., & Quandt, S. A. (1998). Qualitative methods in arthritis research: sampling and data analysis. *Arthritis & Rheumatology*, *11*(1), 66–74. https://doi.org/10.1002/art.1790110111

Asfour, M., Menon, C., & Jiang, X. (2021). A machine learning processing pipeline for reliable hand gesture classification of fmg signals with stochastic variance. *Sensors*, *21*(4), 1–16. https://doi.org/10.3390/s21041504

Aswini, A. M., & Vinod, P. (2014). Droid permission miner: Mining prominent permissions for Android malware analysis. *The 5th International Conference on the Applications of Digital Information and Web Technologies (ICADIWT)*, 81–86. https://doi.org/10.1109/ICADIWT.2014.6814679

Babbage, C. (1864). *Passages from the Life of a Philosopher*. Longman, Green, Longman, Roberts, & Green.

Backer, J. A., Klinkenberg, D., & Wallinga, J. (2020). Incubation period of 2019 novel coronavirus (2019- nCoV) infections among travellers from Wuhan, China, 20 28 January 2020. *Eurosurveillance*, *25*(5), 1–6. https://doi.org/10.2807/1560-7917.ES.2020.25.5.2000062

Bartels, L. M. (2012). *The New Gilded Age*. Princeton University Press.

Biggio, B., & Roli, F. (2018). Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, *84*, 317–331. https://doi.org/10.1016/j.patcog.2018.07.023

Canbek, Gurol. (2021). Gaining new insight into machine-learning datasets via multiple binary-feature frequency ranks with a mobile benign/malware apps example. *Hittite Journal of Science and Engineering*, *8*(2), 103–121. https://doi.org/10.17350/HJSE19030000221

Canbek, Gürol, Baykal, N., & Sagiroglu, S. (2017). Clustering and visualization of mobile application permissions for end users and malware analysts. *The 5th International Symposium on Digital Forensic and Security (ISDFS)*, 1–10. https://doi.org/10.1109/ISDFS.2017.7916512

Canbek, Gürol, Sagiroglu, S., & Taskaya Temizel, T. (2018). New techniques in profiling big datasets for machine learning with a concise review of Android mobile malware datasets. *2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT)*, 117–121. https://doi.org/10.1109/ibigdelft.2018.8625275

Canbek, Gürol, Taskaya Temizel, T., & Sagiroglu, S. (2021). BenchMetrics: a systematic benchmarking method for binary-classification performance metrics. *Neural Computing and Applications*. https://doi.org/10.1007/s00521-021-06103-6

Canfora, G., Mercaldo, F., & Visaggio, C. A. (2013). A classifier of malicious Android applications. *The 8th International Conference on Availability, Reliability and Security (ARES)*, 607–614. https://doi.org/10.1109/ARES.2013.80

Chen, J., Dubut, F., Li, J. (Zengzhong), & Majumder, R. (2021). *Make Every feature Binary (MEB): A 135B parameter sparse neural network for massively improved search relevance*.

Chu, S. (2003). Using soccer goals to motivate the Poisson process. *INFORMS Transactions on Education*, *3*(October 2015), 64–70. https://doi.org/10.1287/ited.3.2.64

Clauset, A., Shalizi, C. R., & Newman, M. E. J. (2009). Power-law distributions in empirical data. *SIAM Review*, *51*, 661–703. https://doi.org/10.1137/070710111

Dougherty, E. R., & Dalton, L. A. (2013). Scientific knowledge is possible with small-sample classification. In *EURASIP Journal on Bioinformatics and Systems Biology* (Issue 10).

Dougherty, E. R., Hua, J., Xiong, Z., & Chen, Y. (2005). Optimal robust classifiers. *Pattern Recognition*, *38*(10), 1520–1532. https://doi.org/10.1016/j.patcog.2005.01.019

El Naqa, I., Li, R., & Murphy, M. J. (Eds.). (2015). *Machine Learning in Radiation Oncology*. Springer. https://doi.org/10.1007/978-3-319-18305-3

Felt, A. P., Chin, E., Hanna, S., Song, D., & Wagner, D. (2011). Android permissions demystified. *Proceedings of the 18th ACM Conference on Computer and Communications Security (CCS)*, 627. https://doi.org/10.1145/2046707.2046779

Frank, S. A. (2009). The common patterns of nature. *Journal of Evolutionary Biology*, *22*(8), 1563–1585. https://doi.org/10.1111/j.1420-9101.2009.01775.x

Gauen, K., Dailey, R., Laiman, J., Zi, Y., Asokan, N., Lu, Y. H., Thiruvathukal, G. K., Shyu, M. L., & Chen, S. C. (2017). Comparison of visual datasets for machine learning. *2017 IEEE International Conference on Information Reuse and Integration (IRI)*, 346–355. https://doi.org/10.1109/IRI.2017.59

Geiger, R. S., Cope, D., Ip, J., Lotosh, M., Shah, A., Weng, J., & Tang, R. (2021). “Garbage in, garbage out” revisited: What do machine learning application papers report about human-labeled training data? *Quantitative Science Studies*, 1–32. https://doi.org/10.1162/qss_a_00144

Gillespie, C. S. (2015). Fitting heavy tailed distributions: The poweRlaw package. *Journal of Statistical Software*, *64*(2). https://doi.org/10.18637/jss.v000.i00

Grimes, D. A. (2010). Epidemiologic research using administrative databases: Garbage in, garbage out. *Obstetrics & Gynecology*, *116*(5), 1018–1019. https://doi.org/10.1097/AOG.0b013e3181f98300

Grother, P., Ngan, M., & Hanaoka, K. (2019). *Face Recognition Vendor Test (FVRT): Part 3: Demographic Effects*. https://doi.org/10.6028/NIST.IR.8280

Heuser, R. R. (1998). Editorial comment: garbage in, garbage out. *Catheterization and Cardiovascular Diagnosis*, *43*(4), 402–402. https://doi.org/10.1002/(SICI)1097-0304(199804)43:4<402::AID-CCD8>3.0.CO;2-C

Hoerl, R. W., Snee, R. D., & De Veaux, R. D. (2014). Applying statistical thinking to ‘Big Data’ problems. *Wiley Interdisciplinary Reviews: Computational Statistics*, *6*(4), 222–232. https://doi.org/10.1002/wics.1306

Hoffmann, J., Ussath, M., Holz, T., & Spreitzenbarth, M. (2013). Slicing droids: Program slicing for smali code. *SAC ’13 Proceedings of the 28th Annual ACM Symposium on Applied Computing*, 1844–1851.

Jiang, X., & Zhou, Y. (2013). *Android Malware*. Springer.

Joo, H., Aguinis, H., & Bradley, K. J. (2017). Not all nonnormal distributions are created equal:

Improved theoretical and measurement precision. *Journal of Applied Psychology*, *102*(7), 1022–1053. https://doi.org/10.1037/apl0000214

Joseph, A. D., Nelson, B., Benjamin, I. P. R., & Tygar, J. D. (2019). *Adversarial Machine Learning*. Cambridge University Press. https://doi.org/10.1017/9781107338548

Juran, J. M., & Godfrey, A. B. (1999). *Juran’s Quality Handbook* (R. E. Hoogstoel & E. G. Schilling (Eds.); 5th ed.). McGraw-Hill.

Kilkenny, M. F., & Robinson, K. M. (2018). Data quality: “Garbage in – garbage out”. In *Health Information Management Journal* (Vol. 47, Issue 3, pp. 103–105). SAGE Publications Inc. https://doi.org/10.1177/1833358318774357

Lew, G., & Schumacher, R. M. (2020). *AI and UX: Why Artificial Intelligence Needs User Experience*. Apress. https://doi.org/10.1007/978-1-4842-5775-3

Limpert, E., Stahel, W. a., & Abbt, M. (2001). Log-normal distributions across the sciences: Keys and clues. *BioScience*, *51*(5), 341. https://doi.org/10.1641/0006-3568(2001)051[0341:LNDATS]2.0.CO;2

Lindorfer, M., Neugschwandtner, M., Weichselbaum, L., Fratantonio, Y., Veen, V. Van Der, & Platzer, C. (2014). ANDRUBIS - 1,000,000 apps later: a view on current Android malware behaviors. *3rd International Workshop on Building Analysis Datasets and Gathering Experience Returns for Security (BADGERS)*, 3–17.

Mahloujifar, S., Diochnos, D. I., & Mahmoody, M. (2019). The curse of concentration in robust learning: Evasion and poisoning attacks from concentration of measure. *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, *33*(01), 4536–4543. https://doi.org/10.1609/aaai.v33i01.33014536

Milojević, S. (2010). Power law distributions in information science: Making the case for logarithmic binning. *Journal of the American Society for Information Science and Technology*, *61*(12), 2417–2425. https://doi.org/10.1002/asi.21426

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill Science/Engineering/Math.

Newman, M. E. J. (2004). Power laws, Pareto distributions and Zipf’s law. *Contemporary Physics*, *46*, 323–351. https://doi.org/10.1016/j.cities.2012.03.001

O’Hurley, G., Sjöstedt, E., Rahman, A., Li, B., Kampf, C., Pontén, F., Gallagher, W. M., & Lindskog, C. (2014). Garbage in, garbage out: A critical evaluation of strategies used for validation of immunohistochemical biomarkers. *Molecular Oncology*, *8*(4), 783–798. https://doi.org/10.1016/j.molonc.2014.03.008

Peiravian, N., & Zhu, X. (2013). Machine learning for Android malware detection using permission and API calls. *IEEE 25th International Conference on Tools with Artificial Intelligence (ICTAI)*, 300–305. https://doi.org/10.1109/ICTAI.2013.53

Peng, H., Gates, C., Sarma, B., Li, N., Qi, Y., Potharaju, R., Nita-Rotaru, C., & Molloy, I. (2012). Using probabilistic generative models for ranking risks of Android apps. *19th Conference on Computer and Communications Security (CCS)*, 241–252. https://doi.org/10.1145/2382196.2382224

Piringer, H., Berger, W., & Hauser, H. (2008). Quantifying and comparing features in high-dimensional datasets. *Proceedings of the International Conference on Information Visualisation*, 240–245. https://doi.org/10.1109/IV.2008.17

Pisarenko, V., & Rodkin, M. (2010). Distributions of characteristics of natural disasters: Data and classification. In *Heavy-Tailed Distributions in Disaster Analysis* (p. 190). Springer, Dordrecht. https://doi.org/10.1007/978-90-481-9171-0

Sahu, B., Dehuri, S., & Jagadev, A. K. (2017). Feature selection model based on clustering and ranking in pipeline for microarray data. *Informatics in Medicine Unlocked*, *9*(July), 107–122. https://doi.org/10.1016/j.imu.2017.07.004

Sarma, B., Li, N., Gates, C., Potharaju, R., Nita-Rotaru, C., & Molloy, I. (2012). Android permissions: A perspective combining risks and benefits. *17th Symposium on Access Control Models and*

Technologies (SACMAT), 13–22. https://doi.org/10.1145/2295136.2295141

Tobias, P. (2012). What are the basic lifetime distribution models used for non-repairable populations? In *e-Handbook of Statistical Methods*. NIST/SEMATECH. https://doi.org/10.18434/M32189

Triguero, I., García-Gil, D., Maillo, J., Luengo, J., García, S., & Herrera, F. (2019). Transforming big data into smart data: An insight on the use of the k-nearest neighbors algorithm to obtain quality data. In *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* (Vol. 9, Issue 2, pp. 1–24). Wiley-Blackwell. https://doi.org/10.1002/widm.1289

Tweedie, R. L., Mengersen, K. L., & Eccleston, J. A. (1994). Garbage in, garbage out: Can statisticians quantify the effects of poor data? *CHANCE*, *7*(2), 20–27. https://doi.org/10.1080/09332480.1994.11882492

Vuong, Q. H. (1989). Likelihood ratio tests for model selection and non-nested hypotheses. *Econometrica*, *57*(2), 307–333.

Wang, W., & Murynets, I. (2013). What you see predicts what you get - Lightweight agent based malware detection. *Security and Communication Networks*, *6*(1), 33–48. https://doi.org/10.1002/sec

Wang, W., Wang, X., Feng, D., Liu, J., Han, Z., & Zhang, X. (2014). Exploring permission-induced risk in Android applications for malicious application detection. *IEEE Transactions on Information Forensics and Security*, *9*(11), 1828–1842. https://doi.org/10.1109/TIFS.2014.2353996

Wheatley, S., Sovacool, B., & Sornette, D. (2017). Of disasters and dragon kings: A statistical analysis of nuclear power incidents and accidents. *Risk Analysis*, *37*(1), 99–115. https://doi.org/10.1111/risa.12587

White, E. P., Enquist, B. J., & Green, J. L. (2008). On Estimating the exponent of power-law frequency distributions. *Ecology*, *89*(4), 905–912. https://doi.org/doi.org/10.1890/07-1288.1

Willemain, T., Fan, H., & Ma, H. (2004). *Statistical analysis of intervals between projected airport arrivals*. https://doi.org/10.1.1.198.4710

Yerima, S. Y., Sezer, S., & McWilliams, G. (2014). Analysis of Bayesian classification-based approaches for Android malware detection. *IET Information Security*, *8*(1), 25–36. https://doi.org/10.1049/iet-ifs.2013.0095

Further Reading

Shalizi, C. (2010). *So, you think you have a power law, do you? Well, isn’t that special?* http://www.stat.cmu.edu/~cshalizi/2010-10-18-Meetup.pdf