Big Data Set Minimization and Inference

Using Sufficient Statistics to Capture Data

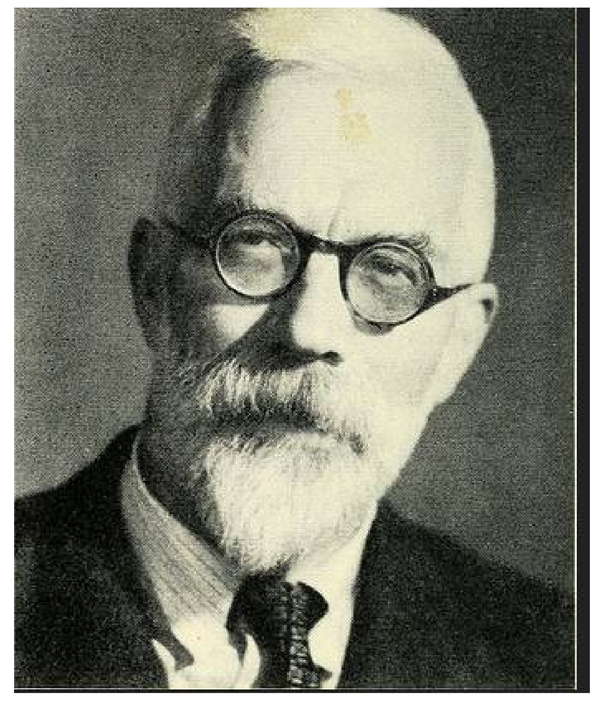
Josh Rendon & Micah A. Thornton
Southern Methodist University

Contact Information:
Department of CSE
Southern Methodist University
6425 Boaz Lane Dallas TX 75205

Phone: +1 (214) 564 9637
Email: `mathornton@smu.edu`

SMU. Darwin Deason Institute for Cyber Security
LYLE SCHOOL OF ENGINEERING

Introduction



As we move into an era where dataset sizes are increasing exponentially with the advent of the internet, and other technologically advanced surveying systems, it becomes impossible to store all of the relevant data. This is especially the case when it comes to storing data relevant to cryptographic functions for analysis. Luckily there exist methods whereby we can reduce the necessary storage through reduction techniques such as the analysis and storage of only sufficient statistics (first introduced by the marvelous Ronald Fisher). In this poster we discuss one such reduction technique used to store and analyze large collections of bit-strings from a statistical perspective.

Background

A Bit-string is a collection of discretized voltage levels that are either 1 or 0. For example consider the bit-string below:

010011001011011101001100101101110100110010110111
010011001011011101001100101101110100110010110111
010011001011011101001100101101110100110010110111
010011001011011101001100101101110100110010110111
010011001011011101001100101101110100110010110111
0100110010110111

This is a collection of 256 bits, in order to store this exact value on a computer the computer would accordingly require 256 bits of memory. there are 2^{256} possible bit strings of this size. Say we wanted to store 1% of them.

$$(0.01) \cdot 2^{256} = 1.15 \cdot 10^{75}$$

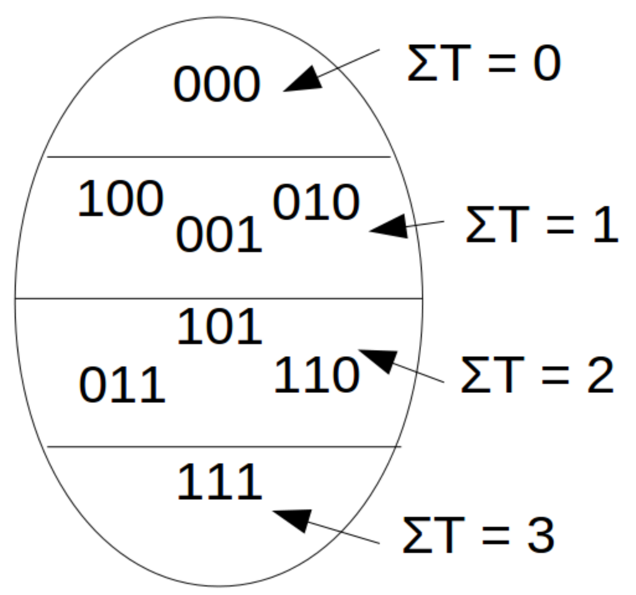
$$1.15 \cdot 10^{75} \text{ strings } \cdot 256 \text{ bits/string } = 2.96 \cdot 10^{77} \text{ bits}$$

$$2.96 \cdot 10^{77} \text{ bits } \cdot \frac{1 \text{ EB}}{2^{68} \text{ bits}} = 1.004 \cdot 10^{57} EB$$

For comparison 1 EB (Exa-byte) is 1048576 Tera-bytes. It is estimated that there is about 295 EB of data stored in computers on Earth.

So data minimization is an absolute must when dealing with large sets of bit strings. The question arises, how can we store this data without losing critical information? Turns out, Fisher answered the question for us, with his analysis of sufficient statistics.

Sufficient Statistics



Think of the bit string before not as a collection of voltage levels, but instead a collection of Bernoulli Trials (IE two possible outcomes Y/N). Now imagine a set of three random variables X_1, X_2, X_3 each of which evaluates to 0, or 1. It has been proven (to see a proof ask!) that the sum of Bernoulli trials is a minimal sufficient statistic(IE minimizes the number of partitions in the data set). A classical result from statistical analysis is that:

$$(X_1, X_2, ..., X_n) \sim Bernoulli(p) \implies (X_1 + X_2 + ... + X_n) \sim bin(n, p)$$

When dealing with a large collection of bit-strings as we are in our research, we can save simply the sum of all the ones in the bit string (known colloquially as the Hamming Weight) of the bit-string. We are also saving another brand of descriptive statistic which we have labeled the ones count, the table below depicts both the one's count as well as the Hamming-Weight storage procedure for a small example string.

Figure 1: Hamming-Weight and Ones-Count Calculation

bit position	7	6	5	4	3	2	1	0	HW
sample A	0	1	0	0	1	0	0	1	*3*
sample B	1	1	0	0	1	0	1	1	*5*
sample C	0	0	1	1	0	1	1	0	*4*
sample D	1	0	1	0	1	0	1	0	*4*
sample E	0	1	0	1	1	1	0	1	*5*
sample F	1	0	1	1	0	1	1	1	*6*
sample G	1	1	1	0	1	0	1	1	*6*
1's count	*4*	*4*	*4*	*3*	*5*	*3*	*5*	*5*	

In essence we are taking data which would produce a uniform distribution on $[0, 2^{256}]$ and reducing it to a data set which is binomial on [0,256] (Hamming Weight). In other words:

$$(X_1, X_2, ...X_{256}) \sim Bernoulli(p) \implies HW \sim Bin(256, p)$$

Now consider if we wanted to compare our collection of bit strings to one which was truly random, in the case that they are truly random there is equal probability they will evaluate to a one or a zero.

$$(X_1, X_2, ...X_{256}) \sim Bernoulli(0.5) \implies HW \sim Bin(256, p)$$

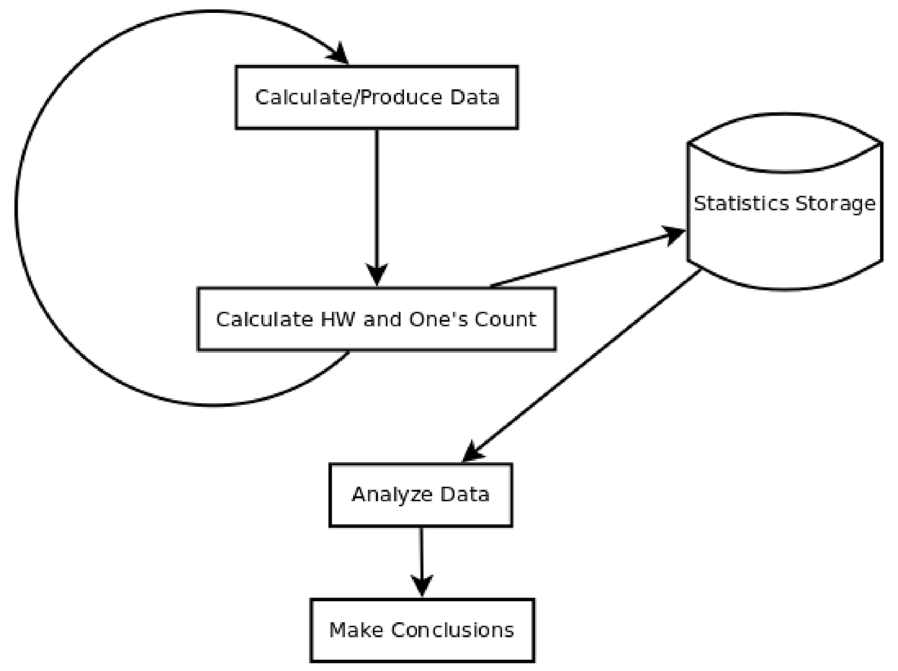
$$(X_1, X_2, ...X_{256}) \sim Bernoulli(0.5) \implies OC \sim DUnif(0, 256)$$

We can now use tests such as goodness of fit tests to compare our data to a theoretical model.

Procedure

Below is a flow diagram representing the procedure that is used in the analysis of our data.

Figure 2: Flow Diagram of Proposed Procedure



After each data generation/calculation cycle we delete the unnecessary data and save off a sample set of the statistics that are used for our analysis. This dramatically cuts the necessary storage for our generated data.

When is this Approach Useful

1. Sampling Quickly and Accumulating Data is not a Problem.
2. Enough Desired Sufficient Statistics for proper Analysis are known
3. Sufficient Statistics can easily be calculated without depending on prior data

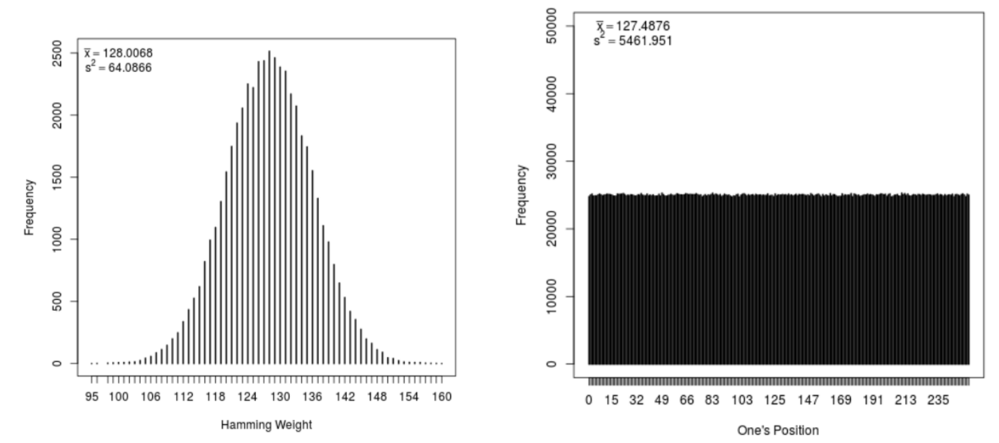
Uses for Sufficient Statistics

- Comparison to idealistic models ie. (Chi Square Goodness of Fit Tests).
- Comparison to other models ie. (Student's T test, Wilcoxon Rank Sum/Signed Rank, ANOVA)
- Modeling/Multipl Regression among many different models.

Results

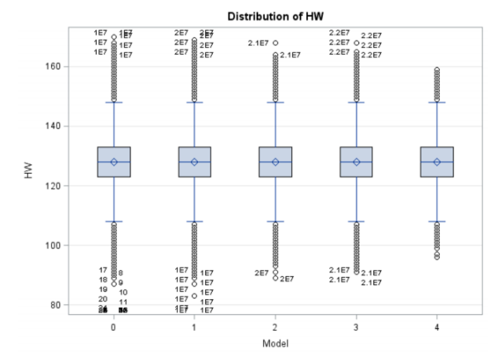
During our research, we produced 50,000 bit strings from several different intentional processes, we wished to compare them to the idealistic model of truly random bit strings and to one another. Histograms of some of our data are provided below:

Figure 3: Sufficient Statistic Histograms



As we can see, they fit the idealized model fairly well! We also used our descriptive statistics from several of our processes to perform ANOVA among the processes themselves.

Figure 4: ANOVA among Processes



Source	DF	Sum of Squares	Mean Square	F Value	Pr>F
Model	4	154	39	0.60	0.6605
Error	2.2E7	1,411,566,164	64	n/a	n/a
Corrected Total	2.2E7	1,411,566,319	n/a	n/a	n/a

We found there was no statistical difference in our processes in the above analysis.

Conclusions

- Data storage can be a huge burden when dealing with big data sets
- Data set minimization can occur by storing only sufficient statistics
- Be cautious of storing the right data for the analysis you desire
- Running analysis and Inference over sufficient statistics can give great insight on the actual data, but is not a replacement for small data sets.

Forthcoming Research

An actual analysis of the memory savings and disadvantages to this method is forthcoming, in addition we are always seeking out new ways to use these sufficient statistics to make inference on our data. Any input would be greatly appreciated. We are also adapting methods for using higher order cummulants such as kurtosis and skewness to our methodology.

Acknowledgements

This work would not have been possible without the key contributions of Ronald Fisher to the field of Statistics.