



# REAL-TIME SOCIAL DATA SAMPLING

SCOTT HENDRICKSON

JEFF KOLB

BRIAN LEHMAN

JOSHUA MONTAGUE

# 1 Introduction

In the world of real-time social data, we are typically observing a series of activities during some period of time and are interested in identifying significant changes in the corresponding activity rate. Such changes may be signals of emerging events or conversations.

In this work, we intend to quantify our ability to identify these kinds of signals. Figure 1 illustrates (schematically) the three main parameters involved in such calculations: signal size, total activities, and confidence. The three parameters are not simultaneously independent; we can choose or measure two of them – possibly based on our particular use-case – and they will determine the third. If chosen as *ad hoc* parameters, respectively, the signal size is a difference in activity rates between two observation times, the number of total activities is a function of the observation time and underlying instantaneous activity rate, and the confidence is a measure of statistical uncertainty in the activity rate e.g. a 95% confidence interval.

For popular topics, social media streams contain a sufficient rate of activities e.g. blog posts or Tweets to create reliable, high-resolution signals in short observation times. However, less popular topics with infrequent activities require additional effort in order to adequately determine the number of activities, signal sensitivity and confidence level appropriate for the situation.

In Section 1.1, we begin with examples of questions that may arise regard-

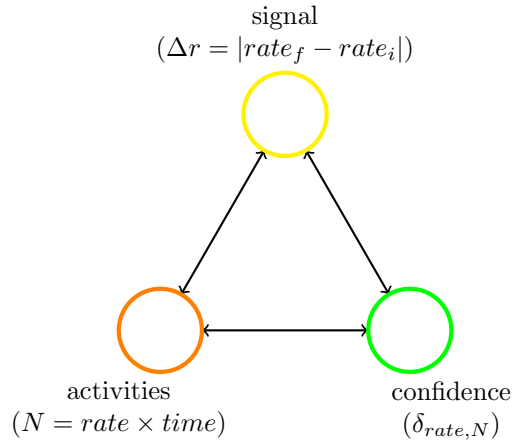


Figure 1: The three parameters used in classifying signal from real-time data: signal size, total activities, and confidence. Signal size is a change in activity rate, the total number of activities observed is a function of observation time, and confidence is that of a reported activity rate. These parameters are not simultaneously independent; we can choose or measure any two and then calculate the third.

ing the sampling of real-time social data. In Section 1.2, we outline some of the mechanisms by which a user can manage their data collection from Gnip, specifically. In Sections 2 and 3, we outline some of the mathematical framework for calculations of activity rate and sampling statistics. Finally, in Section 4, we work through some example calculations.

## 1.1 Motivating Questions

Below are examples of questions regarding activity rate, signal, and confidence level that might motivate the use of this whitepaper. The following Sections and example calculations are intended to answer these kinds of questions.

- The activity rate has doubled from five counts to ten counts between two of my measurement buckets. Is this change significant, or is this expected variation e.g. due to low-frequency events?
- I want to minimize the total number activities that I consume (for reasons of cost, storage, etc). How can I do this while still detecting a factor of two change in activity rate in 1 hour?
- How long should I count activities to detect a change in activity rate of 5%?
- How do I describe the trade-off between signal latency and activity rate uncertainty?
- How do I define confidence levels on activity rate estimates for a time series with only twenty events per day?
- I plan to bucket the data in order to estimate activity rate, how big (i.e. what duration) should the buckets be?
- How many activities should I target to collect in each bucket in order to be have a 95% confidence that my activity rate estimate is accurate for each bucket?

## 1.2 Filtering and Sampling

Rapid growth in the use of social media has led to a large amount of data becoming available from many different sources; Twitter users alone produce approximately 500 million activities per day. In addition to Twitter, Gnip provides access to data from Tumblr, Foursquare, WordPress, Disqus, IntenseDebate, StockTwits, Estimote, NewsGator, as well as easy access to public API data from more than a dozen additional sources. In order to make this large volume of data more manageable, Gnip customers can take advantage of two approaches to sample from our firehoses, reduce overall data consumption, and focus on activities of interest: PowerTrack filtering<sup>1</sup> and sampling<sup>2</sup>.

---

<sup>1</sup>e.g. <http://gnip.com/twitter/power-track/>

<sup>2</sup>Twitter's filtered, rate-limited 1% streaming API provides a non-deterministic combination that is not suitable for many analytic tasks. See [Mor13].

Gnip’s PowerTrack operators allow for filtering of a publisher firehose on keywords or fields that are relevant to the topic in which you are interested. For example, if you are interested in tracking the Super Bowl (the American football event), you might start with a broad stream defined by the keywords “superbowl” “super bowl” and “contains:xlvi”, the latter being a substring match of the Roman numeral of the Super Bowl as might be seen in hashtags or short links. This should limit the social data stream to activities that are more closely related to the actual Super Bowl event.

In the case of a major event like the Super Bowl, the keyword-filtered firehose may still represent a very large number of activities – possibly more than we can store or process in real-time, or more than our budget allows. In this case, adding an additional sampling operator will reduce the delivered data to a known fraction of the firehose, upstream of any PowerTrack filtering. For example, in order to apply our previous Super Bowl PowerTrack rules to a 12% sample of the firehose, we would use a rule such as: “(super bowl OR superbowl OR contains:xlvi) sample:12”. Using a sampling filter effectively decreases the number and rate of delivered activities. Some key features of the sampling operator:

1. 1% resolution
2. Stable sampling rate (even on small time scales)
3. Deterministic sampling returns the same activities for near-rule matches. That is, the same Tweets are returned for matches to the “super bowl” portion of the rules “super bowl sample:12” and “(super bowl OR superbowl) sample:12”.
4. Progressively inclusive sampling. That is, a 2% stream e.g. “sample:2” includes the exact activities from the 1% stream, plus an additional 1%.
5. Sampling operator precedes PowerTrack filtering. That is, activities are first selected from the full firehose to reach the desired sampling rate, then filtered by keywords. PowerTrack filtering is thus applied to a subset of activities that is still representative of the full firehose.

Consider the use of both the sampling operator and PowerTrack filtering in the case of this (fictitious) Super Bowl example. Assume our PowerTrack filtering rules would return  $y = 5\%$  of the full firehose over the course of a day. Assume further that we choose to select an  $x = 12\%$  sample of firehose activities to which our PowerTrack filter is applied. Given that the total number of firehose activities (at the time of writing) is about  $N_{fh} = 500$  M per day, our filtering and sampling will leave us with approximately

$$N_{observed} = xyN_{fh} = 0.12(0.05)500 \text{ M} = 3 \text{ M} \quad (1)$$

activities in this day.

The order of sampling and filtering is critical to maintaining the deterministic nature of subsequently filtered activities. Additionally, filtering prior to sampling would likely increase the duration of time needed to obtain an estimate of true activity rate, and would also inhibit any experiments that attempt to quantify a topic as a fractional component of the full firehose.

## 2 Sampling Parameters

In many situations, a simple question is: *“How many events must we observe in order to detect a change in activity rate?”* Answering this question requires an understanding of the trade-offs between sampling time, activity rate, and signal size.



### 2.1 Activity Rate

The average activity rate in a time bucket is calculated as

$$\bar{r} = \frac{N}{T}, \quad (2)$$

where  $N$  is the number of activities in a bucket of time length  $T$ . Due to the statistical variations in the number of activities in any given time interval, a single calculation of the average activity rate will be only an estimate of the true average. Thus there exists an uncertainty in the estimate of this average rate. Figure 2 illustrates this idea that as we continue to observe additional activities, our confidence in the underlying activity rate grows; the bounds shown are those of a 95% confidence interval.



### 2.2 Signal Sensitivity

Higher underlying activity rates lead naturally to more certain estimates of said rates than lower ones. For a low activity rate, it is possible that small changes in our estimated rate (calculated from one bucket to the next) will be inconclusive; the statistics of infrequent events lead to some amount of variation. In order to declare a valid signal, the variation due to e.g. infrequent events must be smaller than the thing we define as ‘signal.’ Therefore, we observe a valid signal in a time series when the activity rate between buckets has changed by more than the rate signal sensitivity,  $\Delta r$ , defined as

$$|r(t_f) - r(t_i)| \geq \Delta r. \quad (3)$$

Each bucket size is defined by the difference between the variables  $t_f$  and  $t_i$ , the times at which the activity rate is measured. The associated time duration  $T_l = t_f - t_i$  is the signal latency.

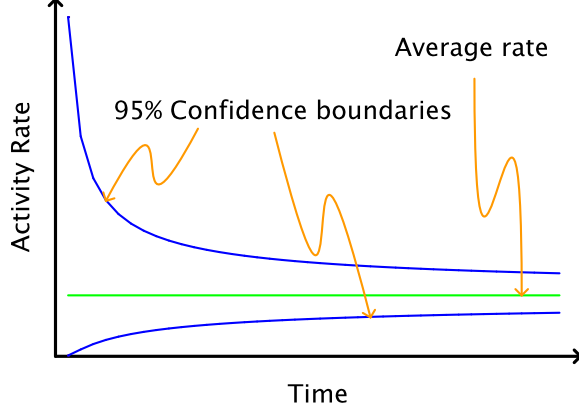


Figure 2: The 95% confidence interval (blue) representing the uncertainty in the estimated activity rate (green) decreases in size as we observe additional activities.



### 2.3 Signal Sensitivity–Confidence Criteria

If we assume we are interested in estimating the activity rate (c.f. Equation 2) of some form of steady-state process, the observed activities in any given period will be distributed about the long term mean. As for a typical Poisson process (discussed further in Section 3.1), the span of fixed-percentage confidence intervals decreases with an increasing number of observed activities, and as mentioned in Section 2.2, this uncertainty also inversely scales with the underlying activity rate.

Referring to the signal definition in Equation 3, we can establish a rough criteria for confidence in terms of the signal uncertainty,  $\delta r$ :

$$\frac{\delta r}{\bar{r}_{low}} << \frac{\Delta r}{\bar{r}_{low}}. \quad (4)$$

where  $\bar{r}_{low}$  will be the lower activity rate estimates of the two observations. The duplicate denominator exists to emphasize the fact that although the span of the confidence interval increases with more activities, the relative interval size decreases.

To make this inequality a bit more concrete, we can introduce a criteria factor,  $\eta$ , which specifies the relative size difference between our observed rate change (i.e. potential signal) and the relative uncertainty interval,

$$\frac{\delta r}{\bar{r}_{low}} \eta = \frac{\Delta r}{\bar{r}_{low}}, \quad (5)$$

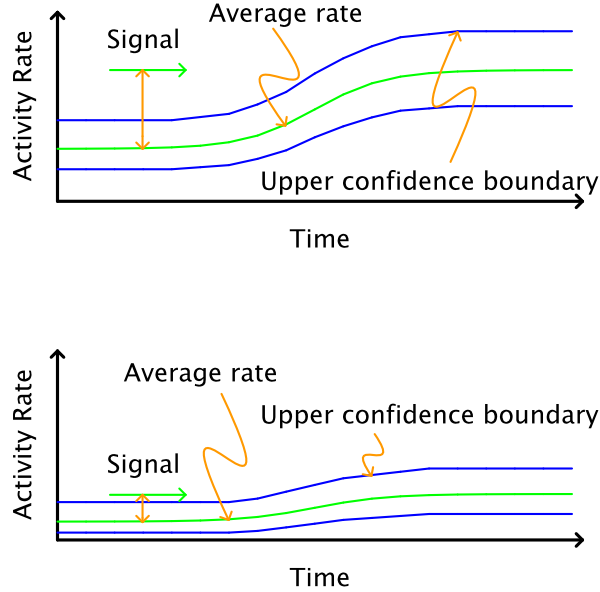


Figure 3: A change in rate from the start time (left) to the end time (right) is established when the change is equal to – or greater than – the uncertainty in the earlier rate estimate. The upper image shows the point at which we cross this threshold. Before this condition is met, the change in rate remains within the uncertainty and observation of a signal is indeterminate. The lower image shows this situation.

where  $\eta \geq 1$ .

We will sometimes refer to the right-hand side of Equation 5 as our ‘signal sensitivity’, and the left-hand side as a ‘relative (confidence) interval size’. Flexibility in the choice of  $\eta$  allows us to prioritize high-certainty classification of signal (larger  $\eta$ , greater separation of observed signal and uncertainty bounds, typically requiring longer observation time), or lower-certainty classification (smaller  $\eta$ , less signal-uncertainty separation, typically requiring shorter observation time).

### 3 Statistics of Time Series of Activities

In this section, we explore some of the underlying mathematics and statistics involved in activity rate estimation. The goal is to demonstrate a method for calculating confidence intervals for rate estimates, and how to consider the available tradeoffs inherent in such a measurement.



### 3.1 Poisson Activity Probability

Because social activities (e.g. Tweets) are timed approximately randomly and have inter-activity times which follow an exponential distribution, we can classify such a process as a Poisson process. As such, we can model the probability of observation times  $t$  between events as

$$p_{activity}(t) = re^{-rt}. \quad (6)$$

This leads to the probability of observing  $n$  activities in time  $t$ , and with activity rate  $r$ , following a Poisson distribution:

$$P(n) = \frac{e^{-rt}(rt)^n}{n!}. \quad (7)$$

The expected value of this distribution  $E[n] = n = rt$ . The mean and variance of the Poisson distribution are both equal to  $r$ [Dev99].



### 3.2 Poisson Confidence Intervals

We are counting activities within a defined time interval in order to estimate the activity rate  $\bar{r}$ . Confidence intervals for the Poisson distribution with confidence level equal to  $100\%(1 - \alpha)$  are given by [Geo12],

$$\frac{1}{2T}\chi^2(\alpha/2; 2n) \leq r \leq \frac{1}{2T}\chi^2(1 - \alpha/2; 2n + 2) \quad (8)$$

where  $\chi^2$  is the inverse cumulative distribution function,  $CDF^{-1}(p; n)$ , of the  $\chi^2$  distribution.<sup>3</sup> Note that with this definition of  $\alpha$ , a confidence interval of 90% corresponds to  $\alpha = 0.1$ . Example 90% confidence intervals and their relative sizes are shown in Table 1.

To determine the parameters satisfying our data collection goals, we can find the value of  $n$  for which the time interval and confidence level match our requirements for signal detection. That is, we can now calculate any one of signal sensitivity, signal latency, activity rate, or confidence level given the other parameters. Example calculations for various design choices are illustrated in the last section of this paper.

### 3.3 Normal Approximations and Bucketed Counts

For a sufficiently large number of observed activities (equivalently, if the activity rate is sufficiently large), the Poisson distribution can be approximated by a normal distribution. For example, the common 95% confidence interval is symmetric about the mean and given by

$$\bar{r} - 1.96\sqrt{\bar{r}/n} \leq \hat{r} \leq \bar{r} + 1.96\sqrt{\bar{r}/n}. \quad (9)$$

<sup>3</sup>A useful approximation to the exact interval is given by:  
 $[n(1 - \frac{1}{9n} - \frac{z_\alpha}{3\sqrt{n}})^3, (n+1)(1 - \frac{1}{9(n+1)} + \frac{z_\alpha}{3\sqrt{n+1}})^3]$



$n$	Interval Bounds	Interval Size ( $\delta n$ )	Relative Interval
1	0.0513, 4.744	4.693	4.693
2	0.3554, 6.296	5.940	2.970
3	0.8177, 7.754	6.936	2.312
4	1.366, 9.154	7.787	1.947
5	1.970, 10.51	8.543	1.709
6	2.613, 11.84	9.229	1.538
7	3.285, 13.15	9.863	1.409
8	3.981, 14.43	10.45	1.307
9	4.695, 15.71	11.01	1.223
10	5.426, 16.96	11.54	1.154
20	13.25, 29.06	15.81	0.7904
30	21.59, 40.69	19.10	0.6366
40	30.20, 52.07	21.87	0.5468
50	38.96, 63.29	24.32	0.4864
60	47.85, 74.39	26.54	0.4423
70	56.83, 85.40	28.57	0.4082
80	65.88, 96.35	30.47	0.3809
90	74.98, 107.2	32.25	0.3584
100	84.14, 118.1	33.94	0.3394
200	177.3, 224.9	47.55	0.2378
300	272.1, 330.1	58.00	0.1933
400	367.7, 434.5	66.82	0.1670
500	463.8, 538.4	74.58	0.1492
750	705.5, 796.6	91.11	0.1215
1000	948.6, 1054.	105.0	0.1050

Table 1: 90% ( $\alpha = 0.1$ ) confidence intervals around the number of events counted,  $n$ , in unit time  $T$ . Rate interval size is  $\delta r = \delta n/T$ . Note that while the absolute size of the interval increases, the relative interval uncertainty decreases.

Depending on our particular use-case or desire for accuracy, use of a normal confidence interval may be sufficient – or at least a fast approximation.

For various reasons, activity counts may be collected in buckets of some pre-defined time length. Estimation of the activity rate may be more naturally calculated by bucket than by total time  $T$  (e.g. as required by our confidence requirements). In general, the relationship between total time  $T$  and (constant) bucket size  $\Delta t$  is

$$\Delta t = \frac{T}{k} \quad (10)$$

where  $k$  is the number of buckets. This parameter can also be used to calculate a corresponding signal latency – in units of ‘buckets’ –  $k_l = T_l/\Delta t$ .

In the case where  $\Delta t \ll T$ , resolution times are typically interchangeable

with the number of buckets. However, in general, the bucket resolution time will not be an even multiple of the bucket size. In this case, the need for a calculation of average activity rate per bucket  $\bar{r} = n/\Delta t$  adds another layer of variability and is beyond the scope of this work.



### 3.4 Summary of Trade-Offs and Parameters

When underlying activity rates are high, we can make highly-certain estimates of the rate in a relatively short time. This also allows us to detect small changes in a previous activity rate (high signal sensitivity). Similarly, lower activity rates lead to larger uncertainty in our estimates and require that we observe for a longer period of time to establish the existence of a signal at the same level of confidence. Along with some example use-case goal/action suggestions, these model trade-offs are summarized in Table 2.

For additional reference, Table 3 includes a summary of the model parameters introduced in this work.

Goal	Possible Actions	Example
Minimize activities (i.e. decrease $n$ )	increase $\Delta r$ (decrease signal sensitivity); decrease confidence factor ( $\alpha$ ); increase $T_l$ (wait longer for the signal)	Section 4.1 (large signal latency)
Increase signal sensitivity (i.e. decrease $\Delta r$ )	increase $T$ (increase number of buckets ( $k$ ); increase bucket size ( $\Delta t$ )); increase activity rate ( $r$ ) by broadening filter or increase PowerTrack sampling	Section 4.3 (sensitivity with high rate)
Decrease signal latency (i.e. decrease $T_l$ )	decrease signal sensitivity $\Delta r$ ; decrease confidence factor ( $\alpha$ ); increase activity rate ( $r$ ) by broadening filter or increase PowerTrack sampling	Section 4.2 (large signal latency)
Decrease signal uncertainty (i.e. decrease $\eta$ )	increase $T$ [increase number of buckets ( $k$ ); or increase bucket size ( $\Delta t$ ); increase activity counts (increase $n$ , $r$ ) by broadening filter; increase PowerTrack sampling	Section 4.2 (small $\eta \leq 1$ )

Table 2: Summary of model trade-offs.

Parameter	Symbol	Definition
Activity count	$n$	Number of activities in time $T$
Sample time	$T$	Duration of observation
Activity rate	$r$	Number of activities per time $T$
Avg. activity rate	$\bar{r} = n/T$	Our estimate of average activity rate
Rate uncertainty	$\delta r$	Uncertainty of our rate estimate
Confidence factor	$\alpha$	Confidence level is 100%(1 - $\alpha$ )
Signal sensitivity	$\Delta r = r_f - r_i$	Detectable change in activity rate
Signal latency	$T_l$	Time required to detect $\Delta r$
Criteria factor	$\eta$	Rate signal criteria multiplier factor (i.e. $\eta = 3$ means relative signal is 3 $\times$ random variations in sample)
Bucket size	$\Delta t$	Predetermined time scale for estimating rate (possibly already determined in your system)
Number of buckets	$k = T/\Delta t$	Observation duration expressed in number of buckets
Sampling operator	$S$	PowerTrack sampling operator (e.g. "sample: $S$ ")

Table 3: Summary of model parameters.

## 4 Example Calculations

### 4.1 Estimate PowerTrack Sampling Operator Value

Recall that the sampling operator allows us to reduce the full firehose of activities to a representative subset of user-defined size. Selecting the value of  $S$  is a process that often starts at  $S = 100\%$ . By monitoring the number of activities,  $n$ , that are filtered through the rules, we get an estimate for  $\bar{r}$ .

Assume that using 100% of the firehose for one minute, we observe  $n = 10$  activities. Further, assume that we would like to detect a change in activity rate from 10 to 20 activities per minute using  $\eta = 3$ . What value of  $S$  should we choose to sample from the firehose?

Imagine for this example that we are comfortable with a signal latency of two days – i.e. our system needs to react to signals in about two days. Given that we expect 10 activities per minute, we should see on average 14,400 activities per day in the full firehose. To calculate the relative interval size, we use Equation 5 and our knowledge of the desired signal sensitivity:

$$\frac{\delta r}{\bar{r}} = \text{Relative Interval Size} = \frac{1}{3} \frac{(20 - 10) \text{ activities}}{\text{min}} \frac{1 \text{ min}}{10 \text{ activities}} = \frac{1}{3}. \quad (11)$$

For a relative interval size of  $1/3 \approx 33\%$ , Table 1 requires about 100 activities (total, over the two days). Hence, instead of using 100% of the firehose, we can

use  $S = \frac{100}{2^{*14,400}} \ll 1\% \rightarrow 1\%^4$ .

## 4.2 Estimate Signal Latency

Imagine we observe rate of 10 activities per minute and we want to detect a change in activity rate from 10 to 20 activities per minute. If we again use a value of  $\eta = 3$ , how long does it take to identify such a change in the activity rate as a signal with 90% confidence level? To calculate an answer, we use the signal sensitivity–Confidence Criteria, Equation 5 and Confidence Interval Sizes from Table 1.

With a criteria factor of 3, our signal sensitivity is

$$\frac{1}{\eta} \frac{\Delta r}{\bar{r}} = \frac{1}{3} \frac{(20 - 10) \text{ activities}}{\text{min}} \frac{1 \text{ min}}{10 \text{ activities}} = \frac{1}{3} \approx 33\%, \quad (12)$$

and with  $n = 10$ , our 90% confidence interval size is about 11.54 (cf. Table 1).

Comparing the confidence interval size to our signal sensitivity,

$$\frac{\delta r}{\bar{r}} = \frac{(11.54)}{10} \approx 115.4\% \not\leq 33\%, \quad (13)$$

we can see that we cannot detect an increase in rate of 10 activities per minute after just one minute.

To determine our signal latency,  $T_l$ , we use the signal sensitivity calculated above and Table 1 to find the approximate number of activities we must observe to meet our criteria. To ensure our relative confidence interval matches our signal sensitivity at 33%, we see that we must observe approximately 100 activities. Since our previous rate,  $\bar{r}_{low}$  was 10 activities per minute, we determine that it will require  $T_l = 10$  minutes of observation to detect our desired change in activity rate.

## 4.3 Estimate Signal Sensitivity

Suppose we would like to determine the magnitude of a change in activity rate needed to classify it as significant. As shown in Equation 5, classifying a signal  $\Delta r$  as significant depends on the choice of criteria factor  $\eta$  and the observation parameters that determine the uncertainty  $\delta r$ . Specifically, we will need to choose a criteria factor  $\eta$  and confidence level  $(1 - \alpha)$ , and our observation will be characterized by total activity count  $n$  and total time  $T$ .

Let us assume we have decided to classify as significant a signal with  $\eta = 10$ , or  $\Delta r = 10 \times \delta r$ . Furthermore, we have chosen a 90% confidence interval and observed  $n=10,000$  activities over a period of  $T=1$  minute (60 seconds) for an estimated rate of  $\bar{r} = 167$  activities per second. We use Equation 8 to calculate the interval of activities for our 90% confidence level, and divide by observation period  $T$  to obtain the corresponding minimum significant rate  $\delta r = 5$  activities per second. Recall, however, that we have also specified a criteria factor  $\eta = 10$ .

---

<sup>4</sup>Recall from Section 1.2 that the sampling operator has a precision of 1%

Therefore, in this example, in order to classify the change in rate as significant, we must observe a change at the level of  $\Delta r = \eta \times \delta r = 10 \times 5 = 50$  activities per second. For an increasing activity rate, this corresponds to a total activity rate of  $167 + 50 = 217$  activities per second. For a decreasing rate, 117 activities per second.

## 5 Conclusion

This work is intended to enable more confident analysis and understanding of the social data streams available through Gnip. A better understanding of the parameter trade-offs involved in any sort of measurement will hopefully empower you to use these data more efficiently in your own environment.

The latest version of this document and supporting code for creating figures and tables can be found at:

<https://github.com/DrSkippy/Gnip-Realtime-Social-Data-Sampling>.

If you find errors in this work, or have comments, please email shendrickson@gnip.com. This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License:

[http://creativecommons.org/licenses/by-sa/3.0/deed.en\\_US](http://creativecommons.org/licenses/by-sa/3.0/deed.en_US).

## References

- [Mor13] F. Morstatter, J. Pfeffer, J. Liu, K. Carley, *Is the Sample Good Enough? Comparing Data from Twitters Streaming API with Twitters Firehose*, <http://www.public.asu.edu/~fmorstat/paperpdfs/icwsm2013.pdf> 2013.
- [Geo12] F. George B. Golam Kibria, *Confidence Intervals for Signal to Noise Ratio of a Poisson Distribution*, <http://thescipub.com/abstract/10.3844/amjbsp.2011.44.55> 2013.
- [Dev99] J. Devore, *Probability and Statistics for Engineering and the Sciences*, Duxbury, 1999

# ABOUT THE AUTHORS



**Scott Hendrickson, PhD**  
Data Scientist

Scott currently leads the Gnip Data Science team at Twitter. He has a deep interest in discovering patterns, creating useful models and sharing a deeper understanding of how people can leverage social data from publishers like Twitter. Prior to Gnip, Scott worked with startups focused on data analysis, machine learning, data visualization and data-centric strategy projects.



**Josh Montague, PhD**  
Data Scientist

Josh is a Gnip Data Scientist at Twitter where he conducts research in support of enterprise customers. He is passionate about designing, building, and sharing data-driven projects that solve problems. Before starting his career in software, Josh spent many years as an experimental condensed matter physicist. He spends his free time enjoying the outdoors in Colorado.



**Jeff Kolb, PhD**  
Data Scientist

Jeff is a Gnip Data Scientist at Twitter. He guides people through the process of deriving genuinely useful conclusions from large-scale Twitter data analysis. In a previous career, he studied the origins of mass at CERN.



**Brian Lehman**  
Data Scientist

Brian Lehman is a Gnip Data Scientist at Twitter. Counting his quiver of race bikes usually results in a value near the number of frisbees in his dog's arsenal of retrieval toys. He previously taught mathematics at the Colorado School of Mines.





303.997.7488 | [info@gnip.com](mailto:info@gnip.com) | [gnip.com](http://gnip.com) | [@gnip](https://twitter.com/gnip)

©2015 Twitter, Inc., or its affiliates.

All rights reserved. TWITTER, TWEET and the Bird Logo are trademarks of Twitter, Inc., or its affiliates.

Figures and statistics in this white paper are all as of June 2015, except as noted otherwise.