Discussion

The results of this test showed that the 4th version of the Twitter Generator was the most random according to the Chi Squared p-value (Figure 5). Because the p-value was so high (56235223621678.4) it means that there is a miniscule chance that there is something other than chance at work. The p-value of the Twitter Generator Version 4 was within 12.5% of the PHP rand() function. It is important to point out that the 2nd version of the Twitter Generator did not get included into results because of an error when collecting data. Each data set is supposed to have one column with the random number, and another column with the exact time and date in which it is generated, unfortunately the time in which it was generated was not logged. This is needed in order to not include any more than the first three million rows that were generated that were included in the results. There is also a version 1.5 that was never tested because of a major error in the programming that could not be resolved, this prevented me from testing it at all.

One other developer wrote an application to generate random numbers utilizing tweets from twitter: “Twitter was full of information, and yet full of entropy. Since random numbers need high-quality sources of entropy, I had the idea to harness Twitter as an entropy source.” (Adams Manor Consulting). Their application had a similar function as mine did, but did not produce integers, it produced random strings simply by generating the MD5 hash of a tweets gained through an RSS feed. They also stated that it had problems with the same tweet being used multiple times in a row, causing the random numbers to be the same many times in the row, and offered that a salt (pseudo-random string) should be added on so the same number would not show up many times in a row. I utilized that concept in the 4th version of the twitter generator.