Kruskal-Wallis Analysis of Emergency Communications Data

Tony R. Dunsworth

Western Governors University

Abstract

During the SARS-COV2 pandemic, the City of Alexandria, Virginia’s Department of Emergency and Customer Communications implemented measures to ensure the continuity of operations to ensure 911 emergency services were available to the constituents of the city. In order to gauge the efficacy of these efforts and to see how those efforts impacted operational times, Kruskal-Wallis and Scheirer-Ray-Hare tests are run on that data to ensure the samples are different and then the medians are compared to one another and using Dunn and Wilcoxon tests to verify which weeks were significantly different and compared to a timeline of continuity measures to see the impacts of those measures on the various call handling points throughout the 911 process from initial call reception to assignment of the first unit in response. These measures lead to suggestions for the department to ameliorate any service impacts identified and thereby ensure quicker, more accurate service to the constituents of the City of Alexandria.

Keywords: SARS-COV2, Kruskal-Wallis, Scheirer-Ray-Hare, nonparametric analysis, Alexandria, PSAP, DECC. 911, Continuity

Kruskal-Wallis Analysis of Emergency Communications Data

As SARS-COV2 (COVID-19) has impacted every aspect of society in the United States and around the world, emergency communications centers across the country have faced significant challenges and have, from the closure of the 911 center in Puerto Rico (“Puerto Rico shutters 911 call centers amid coronavirus cases,” 2020) to the City of Alexandria, Virginia deploying 911 call takers to work from home while isolating other personnel (Stone 2020), addressed the impact in different ways to ensure the continuity of operations to serve their respective communities. To understand the impact of the decisions made by the City of Alexandria, this paper will employ non-parametric analytical techniques to compare data from 2019 and 2020 and within 2020 to view the changes in the operational times for key metrics in the 911 call process: the time from call pick-up to available to dispatch, the time from available to dispatch to the assignment of the first unit, and the time from call pick-up to release of call. Through the analysis, the impact of operational changes can be viewed and recommendations given to address future major events to the benefit of the community served by the public safety answering point (PSAP).

# Research Question

Is there a statistically significant difference in service call processing times in response to the Alexandria, VA efforts to preserve the continuity of service to the public in their 911 Public Safety Access Point?

## Data Collection

The data needed for this analysis exists in a SQL Server database owned by the City of Alexandria, Virginia and maintained in cooperation between the Department of Emergency and Customer Communication (DECC) which is responsible for the 911 and 311 call centers for the city, and the Information Technology Services (ITS) department which is responsible for the city’s technical services. The data is generated through the CentralSquare Enterprise Computer Aided Dispatch software package and is stored, due to the dates studied, on an archive server in a database named Reporting\_System and in one table; Response\_Master\_Incident. (“Computer-Aided Dispatch | CAD Dispatch Software | CentralSquare,” 2020) This table consists of 119 columns, of which only nine were determined to be necessary for this analysis. Those columns are Response\_Date: the datetime stamp the software determines to be the start time for the incident. Priority\_Number: the numeric value assigned to the call based on definitions given by the agencies served by the Public Safety Answering Point (PSAP) and indicating the relative importance of the call on a scale from one to ten; one being the most important and ten being the least important. Problem: the descriptor of the reason for the service call. (e.g., Traffic Stop, Cardiac Arrest) Agency: the responding agency as defined by DECC in concert with the agencies they serve. MethodOfCallRcvd: the way the call for service was received by the PSAP. Fixed\_Time\_PhonePickUp: the datetime stamp recording the moment the call was officially started per the CAD software. Fixed\_Time\_CallEnteredQueue: the datetime stamp recording the moment the call taker makes the call available for the radio dispatcher to assign responding units. Time\_First\_Unit\_Assigned: the datetime stamp recording the moment the radio dispatcher assigned the first unit to the service call. Fixed\_Time\_CallTakingComplete: the datetime stamp recording the moment the call taker stops contact with the reporter and stops actively working the call for service.

After the appropriate columns were identified, an additional ten columns were created for the dataset. These columns were created in order to present additional analytical opportunities and identify additional significant differences in parts of the PSAP operations. The first six of these columns were created from the Response\_Date column. They are: Year: this column indicates the year portion of the datetime value for the call start. In this study the possible values for this column are 2019 and 2020. Month: this column indicates the month portion of the datetime value for the call start. WeekNo: this column indicates the week number as calculated by SQL Server 2016 from the datetime value of the call start. DOW: this column indicates the day of the week as calculated by SQL Server 2016 from the datetime value of the call start. Shift: this column shows the group which received the call as based upon the hour of the start of the call. In this study, the possible values for this column are “Day” and “Night”. “Day” comprises all calls between 6 a.m. and 6 p.m. with “Night” comprising the opposite. The final four columns in this dataset are calculated from the other datetime columns and reflect elapsed times for different stages in DECC’s handling process. The first is QueueTime which is the time elapsed between the Fixed\_Time\_PhonePickUp and Fixed\_Time\_CallEnteredQueue. This is the time the call taker uses to start processing the call and collect enough information to send the service call to a radio dispatcher for assignment. The second is DispatchTime which is the time elapsed between the Fixed\_Time\_CallEnteredQueue and Time\_First\_Unit\_Assigned. The third is Process time which is the elapsed time between the Fixed\_Time\_PhonePickUp and the Time\_First\_Unit\_Assigned. This is the time the radio dispatcher uses to find the appropriate unit(s) and commit the assignment. The third column is CallTime which is the time elapsed between the Fixed\_Time\_PhonePickUp and Fixed\_Time\_CallTakingComplete. This is the time the call taker uses to process the call from pickup to release. The final additional computed column is ProcessTime which is the time elapsed, in seconds, between the time the phone was picked up and the time the first unit was assigned to the call. While this should be an aggregate of QueueTime and DispatchTime, this can vary in the measures of central tendency slightly from a simple addition of the QueueTime and DispatchTime columns.

The base and most of the derived columns were collected via a T-SQL query from a SQL Server 2016 archive database. The final three columns were created through R Studio after the preliminary dataset was imported for analysis. To create these variables, the timestamp variables are aligned for each variable and then subtracted where the output is given as a numeric value. The advantage to this option is the simplicity of derivation through one mathematical operation between the columns. This ensures that there are fewer opportunities for miscalculations. The disadvantage of this method is that it is done after importation into an R tool. If there are problems identified in the dataset, those problems could be magnified through the alteration of the existing columns. The other option considered would have been to use the CAST and CONVERT functions in T-SQL to turn the datetime columns into big integers and then subtract one value from another. The biggest advantage to using dynamic SQL to grab the columns is the ease of collection at the data source. The biggest disadvantage to this method of collection is the impact on the database from which the data is pulled. If the columns needed for the computation are not indexed, performance can be degraded and the calculations can and results can become unreliable.

Due to the nature of this data, prior to collection, an agreement was reached with DECC for access to the data, provided no identifiable nor restricted data was collected. The data fields used for this analysis were approved by an assistant director and the director prior to collection.

### Data Extraction and Preparation

The data was extracted from a SQL Server database through a query which will be included in the exhibits. After the query was prepared and executed, the results were exported to a comma-separated values (csv) file in Microsoft Azure Data Studio. The data was then inspected for NULL values and other anomalies which could be addressed and ameliorated prior to importation into RStudio. After the first query, adjustments were made to the query to account for the additional data fields which needed to be included as columns in the final csv file. The details of the created columns included in the final query have been discussed earlier in the analysis. To provide additional details and address missing values in the csv file, additional changes were made to the SQL query. To give a better breakdown on calls assigned to the Fire Department between fire calls and medical calls, the SQL query uses a case statement to make a separation based upon the problem nature assigned to the service call. In the first view of the csv file there were numerous NULL values in the field MethodOfCallRcvd; a string indicating the origin of the service call. There were definite patterns in the NULL values keyed to certain problem types. The SQL query was then updated to address those discovered patterns, for example, any calls arriving from Mutual Aid partners are then updated in the query to have a value of ‘MUTUAL AID’. For service calls where there is no discernable pattern to be found, the NULL value was changed to “Not Reported’ to eliminate NULL values from that column. The final column which needs updating to address additional NULL values is the start time for the service call. To address this through the SQL query, there are three clauses to address this. If the problem type assigned to the service call is any one of the Mutual Aid problem types, then we use the ClockStartTime as the start of the service call. If there is no entry in the Fixed\_Time\_PhonePickup column, then we use the Time\_PhonePickup field. Otherwise, we will use the Fixed\_Time\_PhonePickup column.

After identifying the columns needed and the computed columns required, restrictive WHERE clauses are added to ensure the data collected fits the parameters needed for the final analytical data set. The first restrictive clause limited the data returned to the last two full years; 2019 and 2020. The next clause restricts the returned data to those where the call taking personnel are part of DECC. The table retains the name of the call takers and is joined against the Personnel table where DECC personnel are identified with a four-digit serial number in the four thousand range. This restriction eliminates most of the Mutual Aid calls from the final data set. These calls are restricted from the data set since the operational procedures for these calls bypass the call taker under most circumstances. The next restrictive clause is designed to ensure the call was dispatched and assigned for service. The final restrictive clause is designed to prevent the inclusion of service calls which were never run. After the query was run and exported to a csv file. The csv file was then inspected to look for missing rows with missing data points. With the construction of the query, there were only 25 rows with a missing data column. Because of the volume of rows collected in the extraction, deleting these rows is an appropriate method of data cleansing.

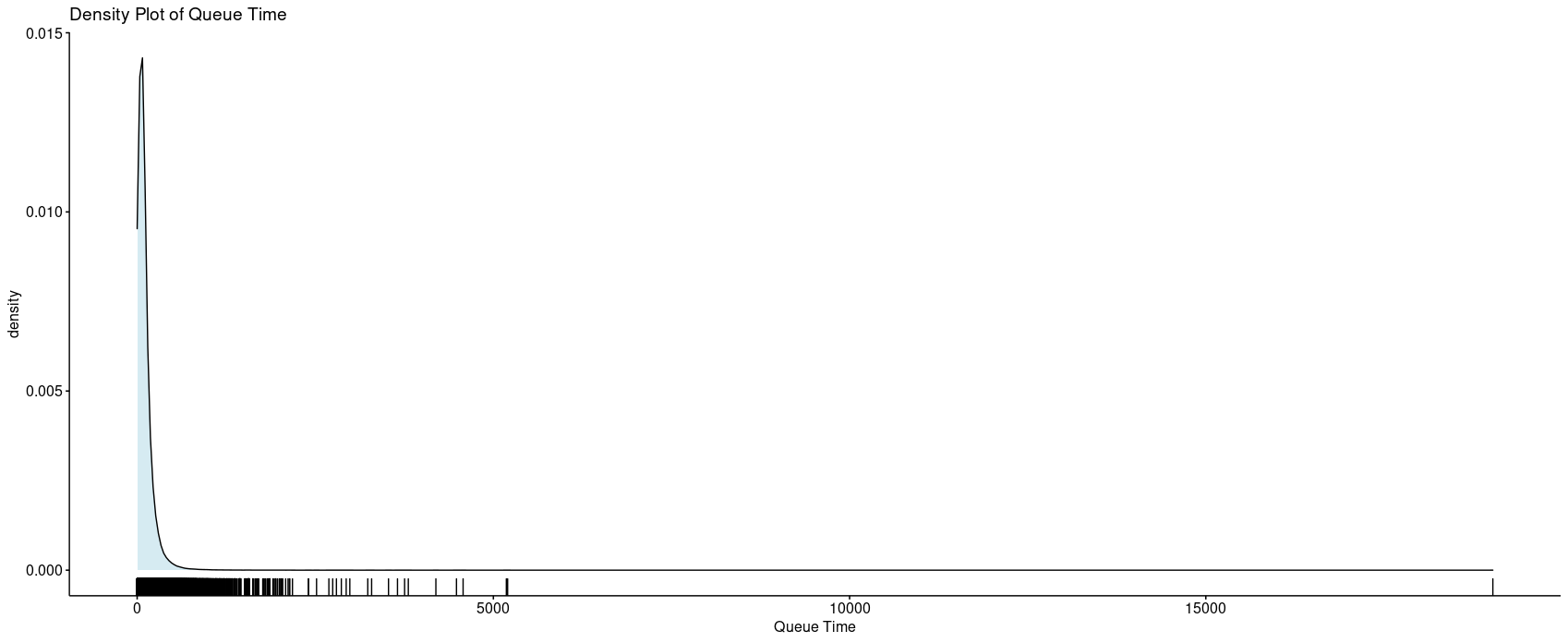
After the data was extracted from the database, it was imported into RStudio for creation of the computed columns and the start of data cleaning. Upon importation into RStudio, the dataset was 15 columns and 150,281 rows. To ensure all data points were correctly imported, instead of using the read.csv function from base R, the readr package was loaded and its read\_csv function was used instead. The advantage to using that specific function is that the datetime columns are properly imported as POSIXct fields. This allows for easier creation of the computed columns discussed above. The character columns are converted into factors using the as.factor function for better univariate analysis. The computed columns are created via subtracting the values of the correct originating columns and the difference converted to a numeric value through the as.numeric function. Now the data frame is 19 columns and 150,281 rows. Next the summary function was run against the computed columns. The minimum values for all of these columns were negative integers. Since these columns are meant to be elapsed times, negative values represent problematic outliers which must be addressed. As there are multiple options using the software as intended which could contribute to a negative value for the difference between two time points, the next step was to count the number of rows where any of the computed columns were a negative value. 571 rows were found to have a computed column with a value less than zero. This represents 0.38% of the rows in the data frame. Since the number of rows is so small, the simplest and safest way to address the negative outliers was to remove the rows from the data frame. Since large positive values for the computed columns could be legitimate values, no rows with large positive values were removed from the data frame. This decision may lead to an increase in the mean of these columns and possibly contribute to an increased right tailed skew in the overall distribution of the remaining data. Comparative normality tests have been run against the data frame as adjusted by the removal of the negative value rows from the data frame and the data frame prior to the negative values removal. Comparative means and medians were also run against both data frames to gauge the impact that a one-sided removal could have on the analyses.

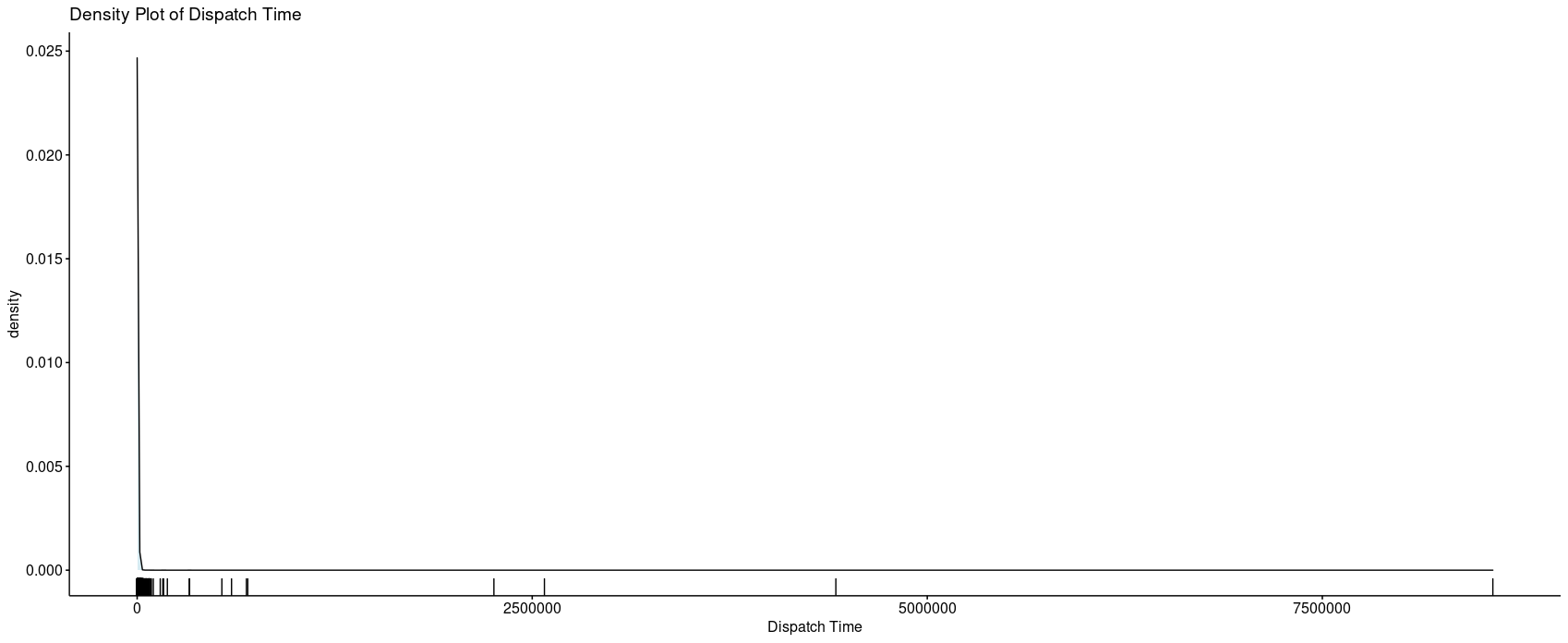
Prior to analysis, after the data has been imported and cleaned, the categorical variables have been converted to factors in order to make univariate and bivariate analyses much easier to perform. Some ordinal numeric variables also had a factor converted version created in R as they are treated, in this analysis, more like categorical variables than numeric variables.

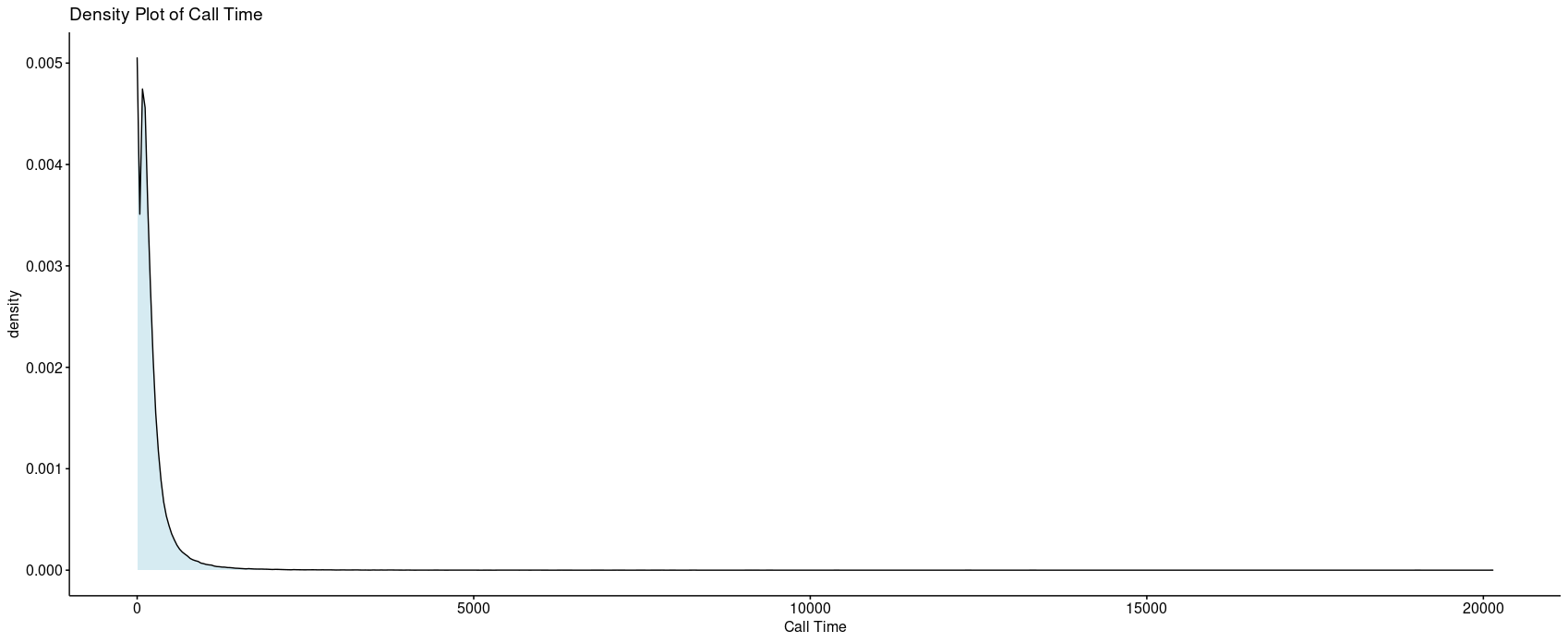
The analyses were performed in RStudio Cloud using the R programming language. R was chosen for the analysis because it can handle the analytical work with fewer add in packages and per Professor Norm Matloff, “R is written by statisticians, for statisticians,” giving it an advantage over Python for detailed statistical analysis (Matloff, 2019).

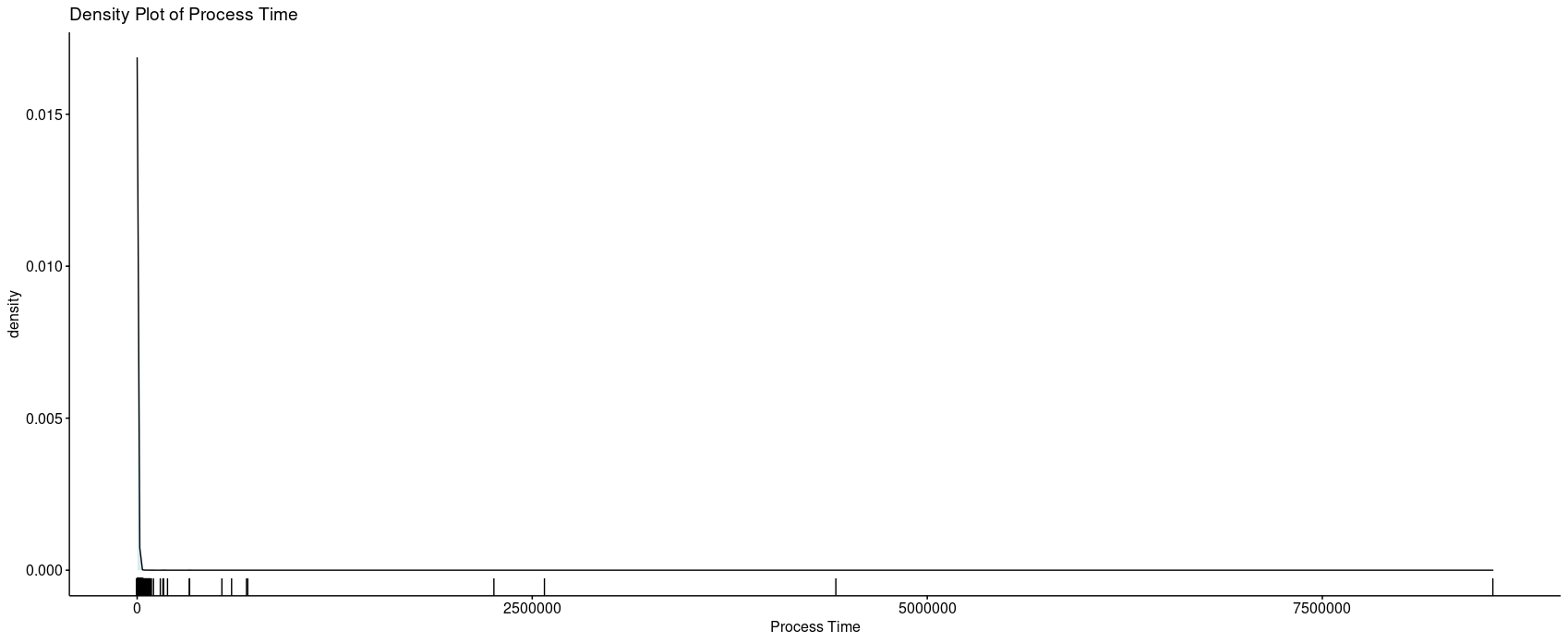
#### Analysis.

After the importation of data and the creation of the computed columns in RStudio, the first step was to address and verify the normality of the continuous computed columns. Three tests were performed on the four computed columns to visually and statistically ascertain normality. The first test performed was the creation of a density plot for each of the four variables. The screenshots of these density plots are as below:

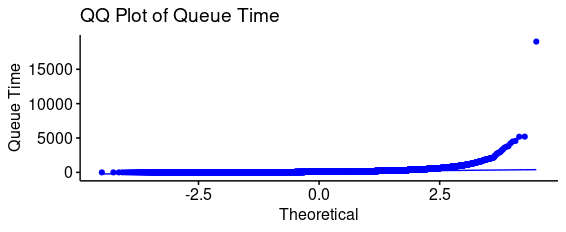
****

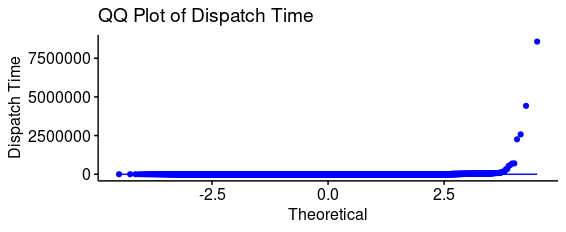
****

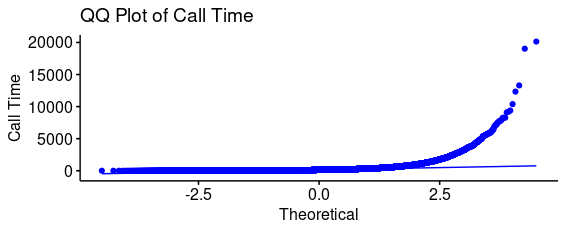
****

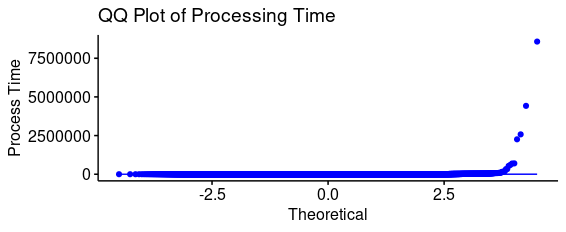
****

As can be seen from these density plots from the ggplot2 package, none of the four continuous variables display a normal distribution. With the length of the tails, all four variables show a significant right skew pattern along with a lack of a normal distribution curve. After the density curves, QQ plots were performed in order to verify the results of the density plots. Those QQ plots are as follows:









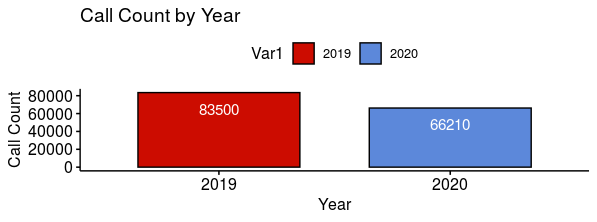
Again, the QQ plots show distributions which are not normal. Together with the density plots, it becomes apparent this data will not conform, with statistical adjustments, to a normal distribution. Finally, as further confirmation an Anderson-Darling normality test was run on each of the computed continuous variables. Because there are ties in the data ranks, the standard Kolmogorov-Smirnov test cannot be run against these variables without throwing errors and warnings. The results of all four variables’ exhibited p values from the Anderson-Darling tests were reported by RStudio as p < 2.2e-16, showing the null hypothesis that the variables are normally distributed can be safely rejected. Since all three tests confirm the variables are not normally distributed, further analysis will use nonparametric methods with comparisons of the population medians rather than using the means which are more impacted by the heavy right skew exhibited by the data.

The medians of the four continuous variables for the entire dataset are as follows:

|  |  |
| --- | --- |
| Variable | Median |
| QueueTime | 76 seconds |
| DispatchTime | 46 seconds |
| CallTime | 127 seconds |
| ProcessTime | 150 seconds |

As the dataset concerns both 2020, the year of the inception of the SARS-COV2 pandemic’s impact on operations and the year prior as a comparator for the impact of the pandemic upon operations, the next step is to start looking at the differences, globally, between the two years included in the dataset.

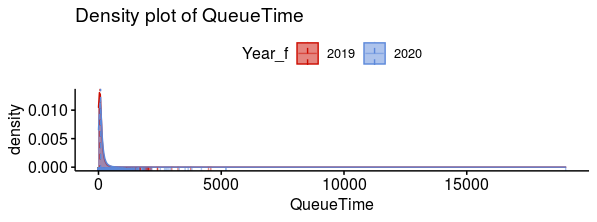
The number of calls for each year are listed in the graphic below:



Overall, there was a decrease in the number of calls from 2019 to 2020 of 17,290 calls which is a decrease of 20.71%. The medians for each continuous variable separated for each of the two years in the dataset are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | 2019 Median | 2020 Median | Difference |
| QueueTime | 68 seconds | 86 seconds | + 18 seconds |
| DispatchTime | 39 seconds | 54 seconds | + 16 seconds |
| CallTime | 108 seconds | 155 seconds | + 47 seconds |
| ProcessTime | 132 seconds | 172 seconds | + 40 seconds |

While there is a decrease in the number of calls between the two years, there is an increase in the medians for each of the continuous variables. Per Profession Salvatore Mangiafico of Rutgers University, as long as the distributions of each group is of similar shape and spread, the Kruskal-Wallis test can be used on the medians to see if there is a significant difference in the population medians Mangiafico (2016). An example of the measure of the distributions, the density plot of the QueueTime for each year of the dataset is produced below:



Looking at the two density plots superimposed on one another, they are both similarly shaped and distributed. Therefore, running the Kruskal-Wallis test can give us a measure of the difference in the population medians. The results of that test are as follows:

Kruskal-Wallis rank sum test

data: QueueTime by Year

Kruskal-Wallis chi-squared = 3231.3, df = 1, p-value < 2.2e-16

As the p-value for this test is less than 0.05, the null hypothesis that there is no significant difference in the medians of the two populations can be rejected. We next need to measure the effect size for the test which, per Steve Draper of the University of Glasgow, measures the degree to which the certainty the results are not an accident but as effect of the difference (Draper, 2020). This is done through the rstatix package using the kruskal\_effsize function against the same parameters.

# A tibble: 1 x 5

.y. n effsize method magnitude

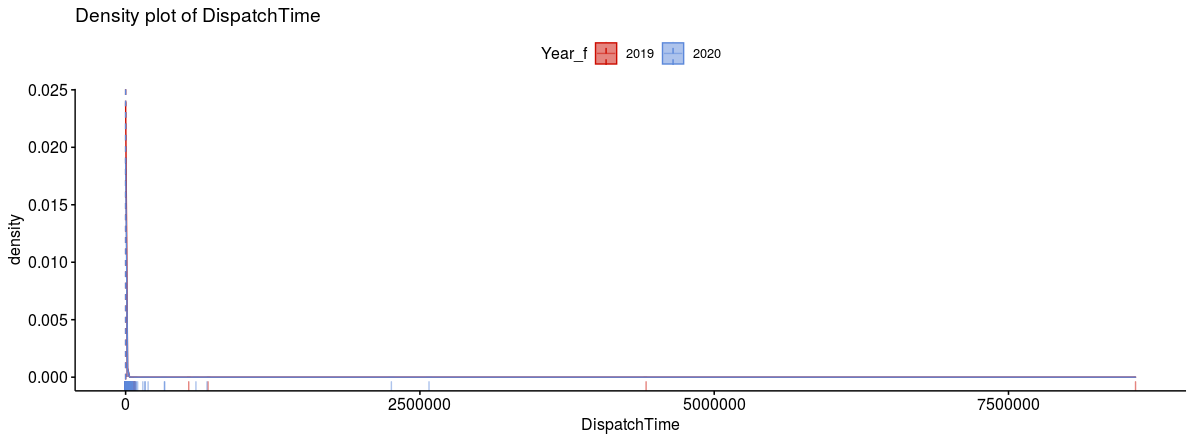
\* *<chr>* *<int>* *<dbl>* *<chr>* *<ord>*

1 QueueTime 149710 0.0216 eta2[H] small

The results of the kruskal\_effsize function indicate the magnitude of the difference of the medians is small.

The analysis of the other continuous variables for differences by year are as follows:

DispatchTime



As with the QueueTime, the density plots are similarly shaped between the two years. Therefore, the Kruskal-Wallis test is applicable for measuring the difference between the means.

Kruskal-Wallis rank sum test

data: DispatchTime by Year

Kruskal-Wallis chi-squared = 679.48, df = 1, p-value < 2.2e-16

In this case, as well, the null hypothesis can be rejected; showing the means come from significantly different populations.

# A tibble: 1 x 5

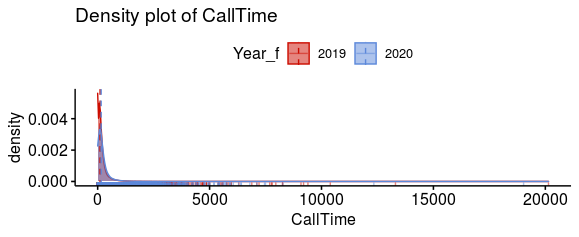
.y. n effsize method magnitude

\* *<chr>* *<int>* *<dbl>* *<chr>* *<ord>*

1 QueueTime 149710 0.0216 eta2[H] small

The effect size shows, like QueueTime, the magnitude of the difference of the means is small.

CallTime:



As with the prior two continuous variables, the density plots are similarly shaped between the two years. Therefore, the Kruskal-Wallis test is applicable for measuring the difference between the means.

Kruskal-Wallis rank sum test

data: CallTime by Year

Kruskal-Wallis chi-squared = 3998.8, df = 1, p-value < 2.2e-16

Since the p-value of the Kruskal-Wallis test is less than 0.05, we can reject the null hypothesis here as well and accept the means come from separate populations. Following this with the effect size test:

# A tibble: 1 x 5

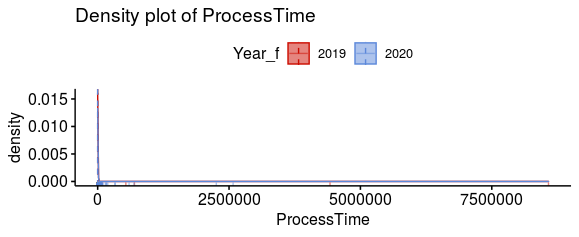
.y. n effsize method magnitude

\* *<chr>* *<int>* *<dbl>* *<chr>* *<ord>*

1 CallTime 149710 0.0267 eta2[H] small

The effect size here, is also small in magnitude, similar to the other continuous variables.

The final continuous variable, ProcessTime, has the following results for the test to use the Kruskal-Wallis test and the results of that test.



This density plot shows a similar distribution to DispatchTime and allows us to continue forward with the Kruskal-Wallis test.

Kruskal-Wallis rank sum test

data: ProcessTime by Year

Kruskal-Wallis chi-squared = 1239.6, df = 1, p-value < 2.2e-16

This variable also shows a p result from the Kruskal-Wallis test which is less than 0.05 and, once again, we can reject the null hypothesis and accept that the medians are drawn from separate populations. The effect size for this is the same as the previous three.

# A tibble: 1 x 5

.y. n effsize method magnitude

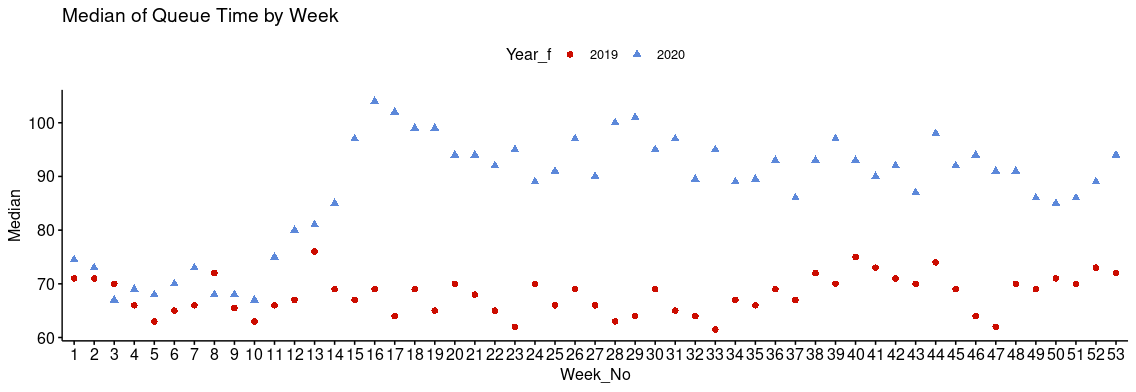
\* *<chr>* *<int>* *<dbl>* *<chr>* *<ord>*

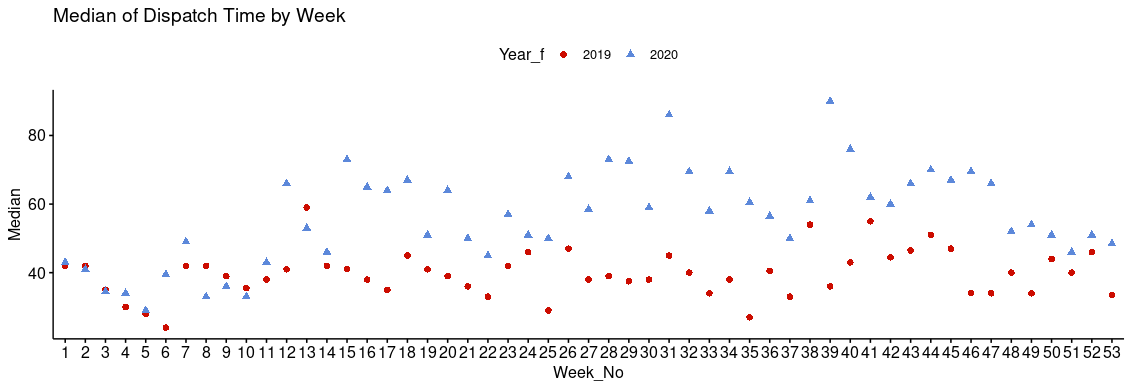
1 ProcessTime 149710 0.00827 eta2[H] small

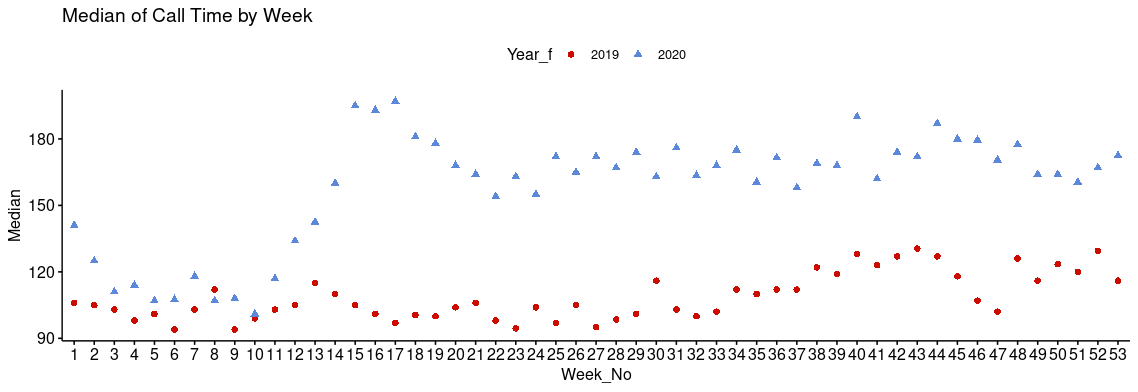
The effects of year on the continuous variables are small. So, while there is a statistically significant difference between the means for the years 2019 and 2020, there may be larger effects upon the means with other variables.

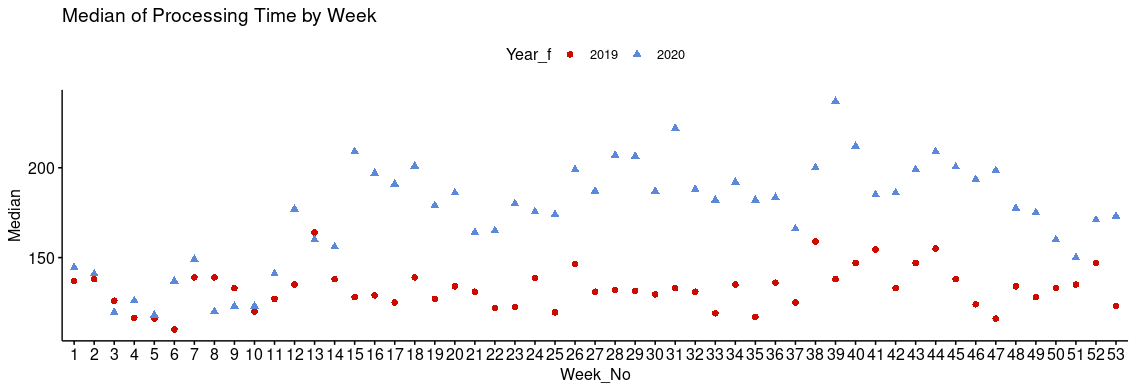
In order to better ascertain significant differences in the medians which can point to the impact of continuity of operation measures taken by DECC, the analysis of the computed continuous variables will be done by combination of independent variables with the ordinal variable Year. This will allow for the illumination of trends between the two years of the study. In support of this, the study is utilizing the Scheirer-Ray-Hare extension to the Kruskal-Wallis test to determine if a statistically significant difference exists between population medians. This method was chosen over Aligned Rank Transformation ANOVA despite the reservations of Professor Mangiafico, (Baharom, Nuawi, Priyandoko, & Mangiafico, 2020) as communicated in a researchgate.com forum because all attempts to perform this test in RStudio have encountered errors or warnings that the ART ANOVA test is not suitable for this dataset. The Scheirer-Ray-Hare was performed on the same dataset with success in each attempt.

Since there is a significant difference in the medians between 2019 and 2020, further research between the two years using an additional independent variable can show further details into the overall impact DECC’s continuity decisions had on the call handling times between the two years and if there are specific areas to analyze within 2020 to view that impact at a more granular level. Since the reporting currently supplied to DECC is compiled and submitted weekly, further research into the differences between weeks can show those impacts most clearly. The medians for each week and the differences between them in numeric form is included at the end of this analysis. The graphical versions of these comparisons are immediately below.









In three of the four variables, there is a noticeable separation of the medians between 2019 and 2020 starting around week 12. Per the former DECC Systems Administrator, Robert Bloom, in an interview with FirstNet (Stone 2020), the use of remote call takers working from home and fielding non-emergency calls started on March 06, 2020. Beginning at the end of week 10, the impact on the medians of the computed variables can be monitored starting with week 11. With the remote call takers restricted to handling non-emergency calls, these call takers could not field inbound 911 calls which now added to the work load of the remaining staff on site. The three computed variables which demonstrate a greater separation all center around the call taking procedures. The remaining variable which doesn’t show the same degree of separation concerns the radio dispatch procedures. In DECC’s workflows, the Fired and EMS services utilize algorithmic assistance for the assignment of resources while the Police services use the algorithmic recommendations as a guide. Because of this workflow difference, the dispatch times for the Fire and EMS services are much lower overall and consistent between the two years. Combining the observation of the median separations with the Kruskal-Wallis and Scheirer-Ray-Hare test, demonstrates the significance and continuity measures had on call handling weekly between 2019 and 2002,

Queue Time:

DV: QueueTime

Observations: 149710

D: 0.9949724

MS total: 1867769484

Df Sum Sq H p.value

Year\_f 1 6.0049e+12 3231.3 0

Week\_No 52 2.1694e+12 1167.3 0

Year\_f:Week\_No 52 2.1590e+12 1161.8 0

Residuals 149604 2.6788e+14

Dispatch Time:

DV: DispatchTime

Observations: 149710

D: 0.9953849

MS total: 1867769484

Df Sum Sq H p.value

Year\_f 1 1.2633e+12 679.48 0

Week\_No 52 1.1995e+12 645.20 0

Year\_f:Week\_No 52 8.1188e+11 436.70 0

Residuals 149604 2.7506e+14

Call Time:

DV: CallTime

Observations: 149710

D: 0.9949851

MS total: 1867769484

Df Sum Sq H p.value

Year\_f 1 7.4314e+12 3998.8 0

Week\_No 52 3.1080e+12 1672.4 0

Year\_f:Week\_No 52 1.8937e+12 1019.0 0

Residuals 149604 2.6579e+14

Process Time:

DV: ProcessTime

Observations: 149710

D: 0.9973837

MS total: 1867769484

Df Sum Sq H p.value

Year\_f 1 2.3093e+12 1239.63 0

Week\_No 52 1.6704e+12 896.69 0

Year\_f:Week\_No 52 9.5507e+11 512.68 0

Residuals 149604 2.7396e+14

As the p-values for each of the computed variables are recorded as p < 0.05, the null hypothesis can be rejected; the weekly medians between years in the data set do come from significantly different populations. After this, Dunn tests were run against the Week\_No variable to determine which weeks exhibited statistically significant medians within all 53 weeks. Due to the size of the output of the Dunn test, the data will be included as a separate spreadsheet with this analysis. That spreadsheet consists of four worksheets, one for each of the computed variables. In brief, week 12 shows significant differences with most of the remaining weeks. In contract, weeks 13 through 15 do not show the same significant differences with other weeks. Approximately half of the week combinations in the Dunn test show some measure of significance and confirm there are differences between populations medians from the same week number in the different years. With these results, analysis can be shifted into 2020 to examine the impact of continuity efforts.

**Continuity of Operations**

Per interviews with Jeff Wobbleton, DECC’s Assistant Director for IT and HR, beginning with the 6th of March, the end of week 10, DECC began the implementation of measures to ensure the continuity of 911 operations as the SARS-COV2 pandemic worsened in the region. The first of these measures was the assignment of some call-takers to work from home using an emergency dispatch kit. When the program started, the remote call-takers were only taking calls from the non-emergency lines. This continued through week 11. At the beginning of week 12, the 16th of March, those call-takers started receiving 911 calls remotely as well. This continuity measure has continued for the remainder of the year. As the situation in the community worsened, DECC created an isolation bubble in the primary dispatch center, accepting volunteers on ten-day assignments to live within the isolation bubble. This started at the end of week 13 and continued through week 22. Three weeks into this program, DECC created a second isolation bubble for one ten-day assignment in weeks 16 and 17; from the 16th of April through the 24th of April. This isolation bubble was set up a local hotel away from any city sites. During these isolation bubbles, operations continued normally at the secondary call center. The last additional continuity measure taken was contracting tele-nurse provider to provide additional screening for certain basic medical calls. This additional screening allows for calls to be moved away from the call-takers quicker and conserve responding resources. This started in week 19 and continued through the remainder of 2020 (J. Wobbleton, personal communication, May 27, 2021).

Analyzing the effects these efforts demonstrated on the medians of 2020 and how those compared with the medians for 2019, as demonstrated above, the separation in medians started being more pronounced in week 12. The highest medians and differences between 2019 and 2020 for the time from call pickup to when it is dispatchable can be seen at weeks 16 and 17 when the isolation bubble at the hotel was deployed. The time to queue remained elevated in 2020 as opposed to 2019 for the remainder of the year. In contrast, neither the addition of the tele-nurse service nor the isolation bubble at the primary call center showed impacts on the service time for call handling. In the time frame from the call entering the queue for dispatch to the time the first unit was assigned by the radio dispatcher, the medians show some separation, but not as marked for most of the year. The largest median and the greatest separation both appear at week 39. Looking at the volume of calls for that week, the call volume does not appear to contribute to an increase in the dispatch times for that week, nor do call volumes between the two years account for that either. The call volume for that week for each of the two years is as follows:

|  |  |  |
| --- | --- | --- |
|  | 2019 | 2020 |
| Call Volume | 1637 | 1265 |
| Median for Dispatch Time | 36.0 | 90.0 |

Call volume fell 22.72% from 2019 to 2020. At the same time, the median time for a dispatcher to assign a call to a unit rose by 150% from 2019 to 2020. Further details will need to be

examined to see what factors contributed to that increase.

##### **Summary and Implications**

Based on the analysis of the dataset, it has been shown that the use of call-takers working remotely contributed significantly to the increase in medians for most of the measured time points in 2020 over the previous year. As noted, the difference in the medians for the time from initial call pickup to the call being sent to the dispatching queue rose significantly in week 12 of 2020 and remainder significantly higher through the remainder of the year. The other continuity effort which showed an impact on this time point was the use of the isolation bubble at an off-site facility. In both cases, parts of the process require an indirect connection to the city’s network and includes data transmission outside of that network. One suggestion would be to utilize the NetMotion solution used by the Police Department which allow connections to be persisted on slower networks and maintain a session even with the loss of connection. (NetMotion Software, 2021) Another recommendation is to include on-duty supervisors on all reports DECC currently receives so they can use that data when response times elevate to develop and implement amelioration strategies to address those elevated response times. It is clear that DECC is creative and willing to employ unique strategies to ensure operational continuity and when that same creativity and willingness to innovate to serve the community are applied, they find solutions which benefit the community’s and their public safety partners’ needs while continuing to be a leader in their field.

# References

Puerto Rico shutters 911 call centers amid coronavirus cases. (2020, October 22). Retrieved April 15, 2021, from AP NEWS website: https://apnews.com/article/virus-outbreak-puerto-rico-latin-america-c939e01656d5c0d2ce490ae9dbe8db46

Stone, L. (2020, May 5). City of Alexandria, VA Utilizes FirstNet for 911 Remote Call Taking During Pandemic | First Responder Network Authority. Retrieved April 15, 2021, from www.firstnet.gov website: <https://www.firstnet.gov/newsroom/blog/city-alexandria-va-utilizes-firstnet-911-remote-call-taking-during-pandemic>

Computer-Aided Dispatch | CAD Dispatch Software | CentralSquare. (2020). Retrieved March 10, 2021, from www.centralsquare.com website: https://www.centralsquare.com/public-safety/cad

Matloff, N. (2019, October 14). *R-vs.-Python-for-Data-Science*. GitHub

Mangiafico, S. S. (2016). R Handbook: Kruskal–Wallis Test. Retrieved May 8, 2021, from <https://rcompanion.org/handbook/F_08.html>

Draper, S. (2020, July 7). Effect size. Retrieved May 8, 2021, from https://www.psy.gla.ac.uk/%7Esteve/best/effect.html

Baharom, M. Z., Nuawi, M. Z., Priyandoko, G., & Mangiafico, S. S. (2020, December 5). Parameter Analysis of Electromagnetic Braking Using Fully Nested and Two Way ANOVA. Retrieved May 12, 2021, from https://www.researchgate.net/post/Is\_there\_a\_non-parametric\_equivalent\_of\_a\_two\_way\_ANOVA

NetMotion Software. (2021, May 20). *Digitally transforming public safety*. <https://www.netmotionsoftware.com/publicsafety>

Wobbleton, J (2021, May 27). Personal communication [email, online chats, and text messages]

Tables

Weekly Numbers:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Queue Time | 2019 | 2020 | Difference | Pct Difference |
| Week 1 | 71 | 74.5 | 3.5 | 4.93% |
| Week 2 | 71 | 73 | 2 | 2.82% |
| Week 3 | 70 | 67 | -3 | -4.29% |
| Week 4 | 66 | 69 | 3 | 4.55% |
| Week 5 | 63 | 68 | 5 | 7.94% |
| Week 6 | 65 | 70 | 5 | 7.69% |
| Week 7 | 66 | 73 | 7 | 10.61% |
| Week 8 | 72 | 68 | -4 | -5.56% |
| Week 9 | 65.5 | 68 | 2.5 | 3.82% |
| Week 10 | 63 | 67 | 4 | 6.35% |
| Week 11 | 66 | 75 | 9 | 13.64% |
| Week 12 | 67 | 80 | 13 | 19.40% |
| Week 13 | 76 | 81 | 5 | 6.58% |
| Week 14 | 69 | 85 | 16 | 23.19% |
| Week 15 | 67 | 97 | 30 | 44.78% |
| Week 16 | 69 | 104 | 35 | 50.72% |
| Week 17 | 64 | 102 | 38 | 59.38% |
| Week 18 | 69 | 99 | 30 | 43.48% |
| Week 19 | 65 | 99 | 34 | 52.31% |
| Week 20 | 70 | 94 | 24 | 34.29% |
| Week 21 | 68 | 94 | 26 | 38.24% |
| Week 22 | 65 | 92 | 27 | 41.54% |
| Week 23 | 62 | 95 | 33 | 53.23% |
| Week 24 | 70 | 89 | 19 | 27.14% |
| Week 25 | 66 | 91 | 25 | 37.88% |
| Week 26 | 69 | 97 | 28 | 40.58% |
| Week 27 | 66 | 90 | 24 | 36.36% |
| Week 28 | 63 | 100 | 37 | 58.73% |
| Week 29 | 64 | 101 | 37 | 57.81% |
| Week 30 | 69 | 95 | 26 | 37.68% |
| Week 31 | 65 | 97 | 32 | 49.23% |
| Week 32 | 64 | 89.5 | 25.5 | 39.84% |
| Week 33 | 61.5 | 95 | 33.5 | 54.47% |
| Week 34 | 67 | 89 | 22 | 32.84% |
| Week 35 | 66 | 89.5 | 23.5 | 35.61% |
| Week 36 | 69 | 93 | 24 | 34.78% |
| Week 37 | 67 | 86 | 19 | 28.36% |
| Week 38 | 72 | 93 | 21 | 29.17% |
| Week 39 | 70 | 97 | 27 | 38.57% |
| Week 40 | 75 | 93 | 18 | 24.00% |
| Week 41 | 73 | 90 | 17 | 23.29% |
| Week 42 | 71 | 92 | 21 | 29.58% |
| Week 43 | 70 | 87 | 17 | 24.29% |
| Week 44 | 74 | 98 | 24 | 32.43% |
| Week 45 | 69 | 92 | 23 | 33.33% |
| Week 46 | 64 | 94 | 30 | 46.88% |
| Week 47 | 62 | 91 | 29 | 46.77% |
| Week 48 | 70 | 91 | 21 | 30.00% |
| Week 49 | 69 | 86 | 17 | 24.64% |
| Week 50 | 71 | 85 | 14 | 19.72% |
| Week 51 | 70 | 86 | 16 | 22.86% |
| Week 52 | 73 | 89 | 16 | 21.92% |
| Week 53 | 72 | 94 | 22 | 30.56% |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dispatch Time | 2019 | 2020 | Difference | Pct Difference |
| Week 1 | 42 | 43 | 1 | 2.38% |
| Week 2 | 42 | 41 | -1 | -2.38% |
| Week 3 | 35 | 34.5 | -0.5 | -1.43% |
| Week 4 | 30 | 34 | 4 | 13.33% |
| Week 5 | 28 | 29 | 1 | 3.57% |
| Week 6 | 24 | 39.5 | 15.5 | 64.58% |
| Week 7 | 42 | 49 | 7 | 16.67% |
| Week 8 | 42 | 33 | -9 | -21.43% |
| Week 9 | 39 | 36 | -3 | -7.69% |
| Week 10 | 35.5 | 33 | -2.5 | -7.04% |
| Week 11 | 38 | 43. | 5 | 13.16% |
| Week 12 | 41 | 66 | 25 | 60.98% |
| Week 13 | 59 | 53 | -6 | -10.17% |
| Week 14 | 42 | 46 | 4 | 9.52% |
| Week 15 | 41 | 73 | 32 | 78.05% |
| Week 16 | 38 | 65 | 27 | 71.05% |
| Week 17 | 35 | 64 | 29 | 82.86% |
| Week 18 | 45 | 67 | 22 | 48.89% |
| Week 19 | 41 | 51 | 10 | 24.39% |
| Week 20 | 39 | 64 | 25 | 64.10% |
| Week 21 | 36 | 50 | 14 | 38.89% |
| Week 22 | 33 | 45 | 12 | 36.36% |
| Week 23 | 42 | 57 | 15 | 35.71% |
| Week 24 | 46 | 51 | 5 | 10.87% |
| Week 25 | 29 | 50 | 21 | 72.41% |
| Week 26 | 47 | 68 | 21 | 44.68% |
| Week 27 | 38 | 58.5 | 20.5 | 53.95% |
| Week 28 | 39 | 73 | 34 | 87.18% |
| Week 29 | 37.5 | 72.5 | 35 | 93.33% |
| Week 30 | 38 | 59 | 21 | 55.26% |
| Week 31 | 45 | 86 | 41 | 91.11% |
| Week 32 | 40 | 69.5 | 29.5 | 73.75% |
| Week 33 | 34 | 58 | 24 | 70.59% |
| Week 34 | 38 | 69.5 | 31.5 | 82.89% |
| Week 35 | 27 | 60.5 | 33.5 | 124.07% |
| Week 36 | 40.5 | 56.5 | 16 | 39.51% |
| Week 37 | 33 | 50 | 17 | 51.52% |
| Week 38 | 54 | 61 | 7 | 12.96% |
| Week 39 | 36 | 90 | 54 | 150.00% |
| Week 40 | 43 | 76 | 33 | 76.74% |
| Week 41 | 55 | 62 | 7 | 12.73% |
| Week 42 | 44.5 | 60 | 15.5 | 34.83% |
| Week 43 | 46.5 | 66 | 19.5 | 41.94% |
| Week 44 | 51 | 70 | 19 | 37.25% |
| Week 45 | 47 | 67 | 20 | 42.55% |
| Week 46 | 34 | 69.5 | 35.5 | 104.41% |
| Week 47 | 34 | 66 | 32 | 94.12% |
| Week 48 | 40 | 52 | 12 | 30.00% |
| Week 49 | 34 | 54 | 20 | 58.82% |
| Week 50 | 44 | 51 | 7 | 15.91% |
| Week 51 | 40 | 46 | 6 | 15.00% |
| Week 52 | 46 | 51 | 5 | 10.87% |
| Week 53 | 33.5 | 48.5 | 15 | 44.78% |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Call Time | 2019 | 2020 | Difference | Pct Difference |
| Week 1 | 106 | 141 | 35 | 33.02% |
| Week 2 | 105 | 125 | 20 | 19.05% |
| Week 3 | 103 | 111 | 8 | 7.77% |
| Week 4 | 98 | 114 | 16 | 16.33% |
| Week 5 | 101 | 107 | 6 | 5.94% |
| Week 6 | 94 | 107.5 | 13.5 | 14.36% |
| Week 7 | 103 | 118 | 15 | 14.56% |
| Week 8 | 112 | 107 | -5 | -4.46% |
| Week 9 | 94 | 108 | 14 | 14.89% |
| Week 10 | 99 | 101 | 2 | 2.02% |
| Week 11 | 103 | 117 | 14 | 13.59% |
| Week 12 | 105 | 134 | 29 | 27.62% |
| Week 13 | 115 | 142.5 | 27.5 | 23.91% |
| Week 14 | 110 | 160 | 50 | 45.45% |
| Week 15 | 105 | 195 | 90 | 85.71% |
| Week 16 | 101 | 193 | 92 | 91.09% |
| Week 17 | 97 | 197 | 100 | 103.09% |
| Week 18 | 100.5 | 181 | 80.5 | 80.10% |
| Week 19 | 100 | 178 | 78 | 78.00% |
| Week 20 | 104 | 168 | 64 | 61.54% |
| Week 21 | 106 | 164 | 58 | 54.72% |
| Week 22 | 98 | 154 | 56 | 57.14% |
| Week 23 | 94.5 | 163 | 68.5 | 72.49% |
| Week 24 | 104 | 155 | 51 | 49.04% |
| Week 25 | 97 | 172 | 75 | 77.32% |
| Week 26 | 105 | 165 | 60 | 57.14% |
| Week 27 | 95 | 172 | 77 | 81.05% |
| Week 28 | 98.5 | 167 | 68.5 | 69.54% |
| Week 29 | 101 | 174 | 73 | 72.28% |
| Week 30 | 116 | 163 | 47 | 40.52% |
| Week 31 | 103 | 176 | 73 | 70.87% |
| Week 32 | 100 | 163.5 | 63.5 | 63.50% |
| Week 33 | 102 | 168 | 66 | 64.71% |
| Week 34 | 112 | 175 | 63 | 56.25% |
| Week 35 | 110 | 160.5 | 50.5 | 45.91% |
| Week 36 | 112 | 171.5 | 59.5 | 53.13% |
| Week 37 | 112 | 158 | 46 | 41.07% |
| Week 38 | 122 | 169 | 47 | 38.52% |
| Week 39 | 119 | 168 | 49 | 41.18% |
| Week 40 | 128 | 190 | 62 | 48.44% |
| Week 41 | 123 | 162 | 39 | 31.71% |
| Week 42 | 127 | 174 | 47 | 37.01% |
| Week 43 | 130.5 | 172 | 41.5 | 31.80% |
| Week 44 | 127 | 187 | 60 | 47.24% |
| Week 45 | 118 | 180 | 62 | 52.54% |
| Week 46 | 107 | 179.5 | 72.5 | 67.76% |
| Week 47 | 102 | 170.5 | 68.5 | 67.16% |
| Week 48 | 126 | 177.5 | 51.5 | 40.87% |
| Week 49 | 116 | 164 | 48 | 41.38% |
| Week 50 | 123.5 | 164 | 40.5 | 32.79% |
| Week 51 | 120 | 160.5 | 40.5 | 33.75% |
| Week 52 | 129.5 | 167 | 37.5 | 28.96% |
| Week 53 | 116 | 172.5 | 56.5 | 48.71% |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Processing Time | 2019 | 2020 | Difference | Pct Difference |
| Week 1 | 137 | 144.5 | 7.5 | 5.47% |
| Week 2 | 138 | 141 | 3 | 2.17% |
| Week 3 | 126 | 119.5 | -6.5 | -5.16% |
| Week 4 | 116.5 | 126 | 9.5 | 8.15% |
| Week 5 | 116 | 118 | 2 | 1.72% |
| Week 6 | 110 | 137 | 27 | 24.55% |
| Week 7 | 139 | 149 | 10 | 7.19% |
| Week 8 | 139 | 120 | -19 | -13.67% |
| Week 9 | 133 | 123 | -10 | -7.52% |
| Week 10 | 120 | 123 | 3 | 2.50% |
| Week 11 | 127 | 141 | 14 | 11.02% |
| Week 12 | 135 | 177 | 42 | 31.11% |
| Week 13 | 164 | 160 | -4 | -2.44% |
| Week 14 | 138 | 156 | 18 | 13.04% |
| Week 15 | 128 | 209 | 81 | 63.28% |
| Week 16 | 129 | 197 | 68 | 52.71% |
| Week 17 | 125 | 191 | 66 | 52.80% |
| Week 18 | 139 | 201 | 62 | 44.60% |
| Week 19 | 127 | 179 | 52 | 40.94% |
| Week 20 | 134 | 186 | 52 | 38.81% |
| Week 21 | 131 | 164 | 33 | 25.19% |
| Week 22 | 122 | 165 | 43 | 35.25% |
| Week 23 | 122.5 | 180 | 57.5 | 46.94% |
| Week 24 | 138.5 | 175.5 | 37 | 26.71% |
| Week 25 | 119.5 | 174 | 54.5 | 45.61% |
| Week 26 | 146.5 | 199 | 52.5 | 35.84% |
| Week 27 | 131 | 187 | 56 | 42.75% |
| Week 28 | 132 | 207 | 75 | 56.82% |
| Week 29 | 131.5 | 206.5 | 75 | 57.03% |
| Week 30 | 129.5 | 187 | 57.5 | 44.40% |
| Week 31 | 133 | 222 | 89 | 66.92% |
| Week 32 | 131 | 188 | 57 | 43.51% |
| Week 33 | 119 | 182 | 63 | 52.94% |
| Week 34 | 135 | 192 | 57 | 42.22% |
| Week 35 | 117 | 182 | 65 | 55.56% |
| Week 36 | 136 | 183.5 | 47.5 | 34.93% |
| Week 37 | 125 | 166 | 41 | 32.80% |
| Week 38 | 159 | 200 | 41 | 25.79% |
| Week 39 | 138 | 237 | 99 | 71.74% |
| Week 40 | 147 | 212 | 65 | 44.22% |
| Week 41 | 154.5 | 185 | 30.5 | 19.74% |
| Week 42 | 133 | 186 | 53 | 39.85% |
| Week 43 | 147 | 199 | 52 | 35.37% |
| Week 44 | 155 | 209 | 54 | 34.84% |
| Week 45 | 138 | 200.5 | 62.5 | 45.29% |
| Week 46 | 124 | 193.5 | 69.5 | 56.05% |
| Week 47 | 116 | 198.5 | 82.5 | 71.12% |
| Week 48 | 134 | 177.5 | 43.5 | 32.46% |
| Week 49 | 128 | 175 | 47 | 36.72% |
| Week 50 | 133 | 160 | 27 | 20.30% |
| Week 51 | 135 | 150 | 15 | 11.11% |
| Week 52 | 147 | 171 | 24 | 16.33% |
| Week 53 | 123 | 173 | 50 | 40.65% |