# The art and science of working with large datasets

Agnes McFarlin

November 29, 2023

# Abstract

The CDC and Robert Wood Johnson foundation have been partnering for the last few years to provide a huge repository of data and metrics on public health[?]. This data is known as the PLACES dataset. There are hundreds of cities and many disease metrics that are tracked, meaning a vast amount of information can be mined and gained from such a repository. My project hopes to do two things: First would be to navigate such a large dataset successfully, second to answer two questions: For the States with more PLACES data locations, are disease metrics generally different. And do states with more PLACES locations have higher than average metrics?

# 1 Introduction/background

There is a finesse that comes with looking over any dataset and understanding what mathematical concepts to apply, and when to apply them. On the one hand, data that are not normally distributed are treated one way, using nonparametric tests such as the Kruskal-Wallis H. test[10][6]. And then following up with post-hoc testing for example the Dunn test. And nonparametric testing can only explain if two samples are different from each other, no conclusion about the direction or magnitude of the difference is allowed to be drawn. But on the other hand given enough data points, normality can be assumed. And any violations of normality should not cause issues .[2][4]That same dataset can suddenly be assessed using alternative techniques, such as t-tests and more thorough conclusions such as the direction and magnitude of differences can be drawn. Public Health is an important aspect for any governing body. And the United States is no stranger to this,with a population in the hundreds of millions managing any type of health crisis as it emerges is no easy task.That is the overall purpose of the PLACES project, to track emerging disease trends and help with the allocation of resources to areas most in need.[1] The original name of the PLACES project was the 500 Cities Project, which tracked the largest 500 cities and compiled health data for them[1]. It has since grown and become the PLACES project which now tracks 28,000 locations[1], providing a very wide swath of information to work with. What areas have the most well, places in the PLACES dataset? The figure below illustrates the number of locations broken down per state.The three States with the largest numbers of PLACES locations are PA, TX, CA. PA has 1718 locations, TX has 1667, and CA has 1455. Those will be the focus of comparisons for the project.

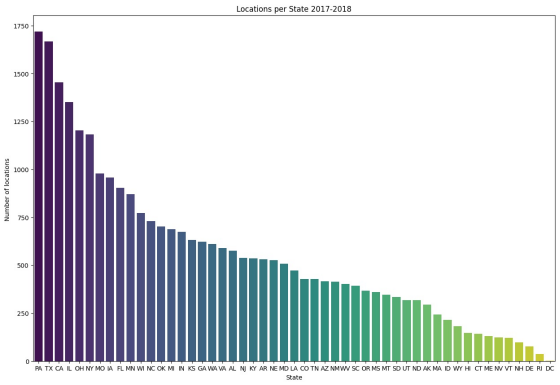


Figure 1: Number of locations by State in the PLACES dataset

# 2 Supportive Work

The CDC PLACES webpage provides a huge repository of data, and also basic ways to look over this data. You can view County comparisons for up to 3 Counties, and there is also an interactive map to provide a straightforward and easy method for interpreting results.[1] But none of the data is year-to-year. There are papers, which can be found in references. But all seem

to focus on a few disease metrics and not changes over time. This may be due to the fact that the datasets are massive to say the least and that certain metrics are obtained by estimation.[3] Despite this, a look into year-to-year changes, even if for information purposes only can yield useful insights.

## 3 Tools, Dataset, Tasks

The datasets employed will be from the Data.gov repository but can also be obtained on the CDC website.[11][1] There are three in total, each spanning two years. Each dataset contains City and State names, Geolocation data, Disease Metric Data and resulting values, reported as percentage values. Python and any libraries needed will be employed. Data will be loaded in, and inspected, cleaned if needed. Data will be visualized and evaluated for distribution fit, and from there the correct type of analysis will be used. Consisting of either parametric or non-parametric testing. The results will then be analyzed, interpreted and a conclusion will be drawn.

## 4 Evaluation Plan

Success for this project was information gain and successful manipulation of the data sets themselves. There were two hypotheses to test. One tested if certain States had different metric results than other States. Specifically the 3 States with the most PLACES locations. (PA,TX,CA) The second tested whether certain States have higher mean or median values compared to other States. The testing procedure split the data into pairs with one part of the pair being the State in question, the other part of the pair consisted of every other State and each of the 30 or so measures was compared between them. Both parametric and non-parametric testing was used.

## 5 Current Progress

The three States with the largest numbers of PLACES locations are PA, TX, CA. Those were the focus of comparisons for the project. Otherwise there would be too many variables to be able to display in a manner that would not overwhelm the reader. The datasets were loaded into Python as Pandas DataFrames and preliminary inspection of the columns and data was performed. Each dataset spanned about two years, the over all time being between 2017-2021.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1593206 entries, 0 to 1593205
Data columns (total 15 columns):
 #   Column                Non-Null Count    Dtype
---  ------                --------------    -----
 0   Year                  1593206 non-null  int64
 1   StateAbbr             1593206 non-null  object
 2   LocationName          1593206 non-null  object
 3   DataSource            1593206 non-null  object
 4   Measure               1593206 non-null  object
 5   Data_Value_Unit       1593206 non-null  object
 6   Data_Value_Type       1593206 non-null  object
 7   Data_Value            1593206 non-null  float64
 8   Low_Confidence_Limit  1593206 non-null  float64
 9   High_Confidence_Limit 1593206 non-null  float64
 10  TotalPopulation       1593206 non-null  int64
 11  Geolocation           1593206 non-null  object
 12  LocationID            1593206 non-null  int64
 13  CategoryID            1593206 non-null  object
 14  MeasureId             1593206 non-null  object
dtypes: float64(3), int64(3), object(9)
memory usage: 182.3+ MB
```

Figure 2: Summary of DataFrame

And the total number of columns in each was the same, 21. Some clean up was performed and the number of columns was reduced to 15 in an attempt to load the data into repositories as zipped files. But the resulting data sets were still too big even after data reduction so the changes were not needed. Each dataset contained 51 States (Washington DC was included). The other column which was to be used for analysis was the Measures column which contained between $28 - 31$ differently named measures. There were missing values in certain columns, the "Data Value" column specifically, and those were filtered out in leu of backfilling.

Before any kind of analysis could begin each of the datasets was split into smaller pieces, each chunk being a single State worth of data which took up much less space as shown below.

```
dtypes: float64(2), int64(4), object(9)
memory usage: 4.6+ MB
```

Figure 3: Each smaller DataFrame took up just a few MB at a time

Each smaller DataFrame still contained all needed columns, but was much more manageable. The smaller DataFrames all resided within a larger parent list, each of which were organized by year. The hope was to perform analysis in small chunks instead of iterating over the whole DataFrame over and over. The next step, which was repeated throughout the project, was to ensure the code itself was as efficient as my skillset would allow. For example, I tried to work with the list of DataFrames as often as possible and when I was able to successfully use them to perform the analysis the run times were about 50% of what they were when using the parent DataFrames. And cutting the run times down to a manageable level was a personal goal during the project. Another step taken was trying to perform calculations within each sub DataFrame and completing all of them before moving onto the next portion. For that, ensuring my loops and list comprehensions were indented correctly and performed as they should was imperative.

Next, outliers needed to be dealt with, and for very large data sets it can be quite cumbersome to attempt to use visualization methods. With each State having 28-31 measures for analysis plotting histograms, density plots,or qq plots did not make much sense. That would mean 1,428 histograms if one were plotted at a time. No matter how it was sliced, that was an unrealistic amount of data to go over by hand. Instead, the interquartile range method was applied. Each State and each measure within each State had the interquartile range calculated

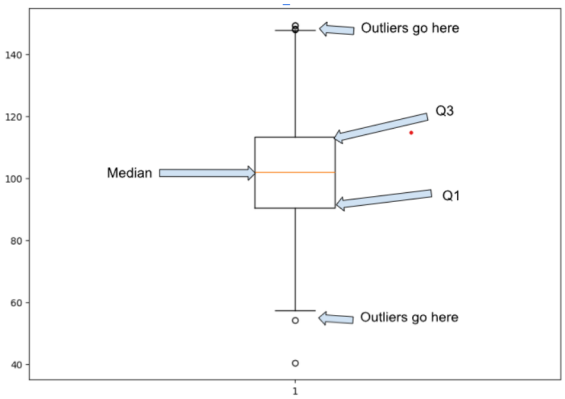


Figure 4: Box Plot to illustrate IQR method

In order to remove outliers we can use the IQR method. (Interquartile range method) Where we set up a 'fence' outside of Q1 and Q3. Anything outside of this fence is considered an outlier. And will be dropped. The formula for calculating the 'fence' is :

$((1.5 * IQR) - Q1)$ OR $((1.5 * IQR) + Q3)$

- * $IQR$ is the interquartile range. Which is obtained by subtracting Q1-Q3.

Following that, mass distribution fitting was performed using the distfit package in Python. Only about 1% of the Data fit a normal distribution. A snippet of the output is below.

```
('normal', 19)
('not_normal', 1410)
('normal', 12)
('not_normal', 1497)
('normal', 13)
('not_normal', 1513)
```

Figure 5: The output labelled normal is for normal distributions, not normal is for data that does not fit a normal distribution

Due to the lack of normality, Spearman's rank order correlation was used to find and filter out highly correlated variables. This was done to reduce collinearity and improve results. [16] The formula for Spearman's R is as follows:[5]

$$rs = \rho R(X), R(Y) = \frac{cov(R(X),R(Y))}{\sigma_{R(X)}\sigma R(Y)}$$

- $cov(R(X), R(Y))$ =The covariance of the ranked variables
- and $p$ denotes the pearson correlation coefficient.

Even after filtering, there were still several thousand combinations of State pairs and Measures remaining.

For information gain, non-parametric testing was first used in the form of the Kruskal-Wallis H test. This test works on two or more independent samples, testing the null hypothesis that all population means or medians are equal[5]. If the resulting p value was lower than the set alpha (0.05), there was said to be a difference. The formula used is below[6]:

$$H = \frac{12}{n(n+1)} \sum_{i=1}^{k} \frac{R^2}{n_i} - 3(n-1)$$

- Where n$i$ is the sample size for each of groups (1,2,3...k) from 1 to k.

Then the rank is computed (Ri), The resulting statistic is similiar to a chi-squared distribution with k-1 degrees of freedom. If the p value is less than alpha which is 0.05, we reject the null hypothesis. That the means/medians of the states [6] While there were many more failures to reject, the count of values that rejected the null hypothesis warranted the use of post-hoc testing, to see which values differed exactly. The Dunn test uses pairwise comparisons to see which values differ.[15] The formula is given below:

$$z_i = \frac{y_i}{\sigma_i}$$

- where i is one of the 1 to m multiple comparisons
- $\sigma_i$ is the standard deviation of $yi$

A limitation of non-parametric testing is it does not determine whether the compared samples are higher or lower, just that they differ. If there are enough data points(above 30) violations of normality are able to be ignored.[2] After the Dunn test, parametric testing was used to test the second hypothesis: That States with more PLACES locations had higher means for Disease metrics than other States. The Levene's test was used for dividing the data into sections where variance was equal and sections where variance was not equal.

The formula used is below[8]:

$$W = \frac{(N-k)}{(k-1)} \frac{\sum_{i=1}^{k} N_i(Z_i - Z)^2}{\sum_{i=1}^{k} \sum_{j=1}^{N_i} (Z_{ij} - Z_i)^2}$$

- $k$ being the number of groups to which the sampled cases belong
- $N_i$ is the number of cases in the ith group
- $N$ is the total number of cases in all groups
- $Y_{ij}$ is the variable from the jth case from the ith group
- $Z_{ij}$ is either the mean or median of each group
- $Z$ is each individual value

If the resulting p-value of the Levene's tests are less than some significance level (typically 0.05), the obtained differences in sample variances are unlikely to have occurred based on random sampling from a population with equal variances. Thus, the null hypothesis of equal variances is rejected and it is concluded that there is a difference between variances in the population. The scipy stats Levene test library was used to perform the Levene Test. The table below provides the sums of values less than alpha and greater than alpha for each year. Most States did end up having different variances. This influenced the version of the next parametric test that was used.

The t-test compares means between two groups and determines whether differences between them are due to chance.[9] For each type

| Year | p-value <0.05 | p-value >0.05 |
|---|---|---|
| 2017-2018 | 3213 | 892 |
| 2018-2019 | 3239 | 1043 |
| 2019-2020 | 3414 | 956 |

of result a variation of the Independent two sample t-test was used. Where equal variances were found the below formula was used[9]:

$t = \frac{\bar{X_1} - \bar{X_2}}{s_p * \sqrt{(\frac{1}{n_1} + \frac{1}{n_2})}}$

- $\bar{X_1}$ is the mean of sample 1
- $\bar{X_2}$ is the mean of sample 2
- $n_1$ is the sample size of sample 1
- $n_2$ is the sample size of sample 2
- $s_p$ is the pooled variance calculated by:

  $s_p = \sqrt{\frac{(n_1-1)s^2_{X_1} + (n_2-1)s^2_{X_2}}{n_1+n_2-2}}$
- $s^2_{X_1}$ is the variance for sample 1
- $s^2_{X_2}$ is the variance for sample 2
- $n_1 + n_2 - 2$ calculates the degrees of freedom

If the Levene's test showed that variances were not equal, Welch's t-test was used. Instead of a pooled variance the difference between means is used. [9]

The formula for Welch's t-test is below:

$t = \frac{\bar{X_1} - \bar{X_2}}{s_{\bar{\Delta}}}$

with $s_{\bar{\Delta}} = \sqrt{\frac{s^2_1}{n_1} + \frac{s^2_2}{n_2}}$

- $s_{\bar{\Delta}}$ being the difference between the means and is not a pooled variance in this case
- $n_1$ is the sample size of sample 1
- $n_2$ is the sample size of sample 2
- and the degrees of freedom are calculated using: $d.f. = \frac{(\frac{s^2_1}{n_1} + \frac{s^2_2}{n_2})^2}{\frac{(s^2_1/n_1)^2}{n_1-1} + \frac{(s^2_2/n_2)^2}{n_2-1}}$
- $s^2_1$ is the unbiased estimator of the variance of sample 1
- $s^2_2$ is the unbiased estimator of the variance of sample 2

# 6 Results, Conclusion, Discussion

For the non-parametric tests the results for the Kruskal-Wallis H test were as follows: While there were many failures to reject there were still a number of significant results, which warranted the use of the post-hoc testing.

| Year | p-value <0.05 | p-value >0.05 |
|---|---|---|
| 2017-2018 | 233 | 3872 |
| 2019-2019 | 217 | 4065 |
| 2019-2020 | 226 | 4144 |

Table 1: Kruskal-Wallis H.testing results

The Dunn test was then performed and conversely, using pairwise comparisons there was a much larger number of differences than for the Kruskal-Wallis test. In fact the values reversed.

| | p-value <0.05 | p-value >0.05 |
|---|---|---|
| 2017-2018 | 3732 | 552 |
| 2018-2019 | 3685 | 599 |
| 2019-2020 | 3942 | 648 |

Table 2: Dunn test results

For parametric testing no matter which test was used, the results were compiled and counted. At first a general concensus was visualized/obtained by summing up all of the times the States in question resulted in rejecting the null hypothesis vs the sum of failiures to reject. The results obtained were quite cumbersome to handle. An example visualization. The blue marks are where the State of Texas had lower p values than the State to which it was compared on a specific measure. (rejecting the null) The orange denotes there was a failiure to reject comparing

the States on a specific measure. And the brown helped illustrate how large the differences were. While it is nice to look at, a more efficient way to sort through the data was used.

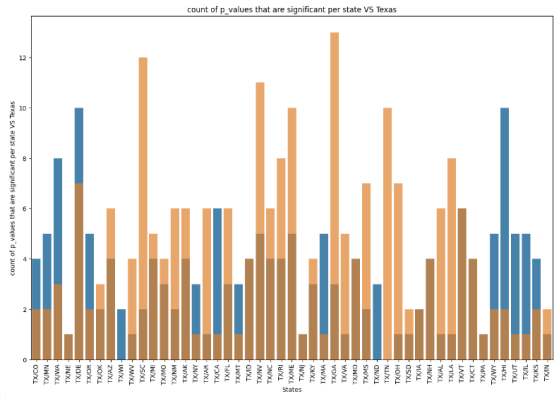


Figure 6: There is some clarity to this method but it requires a lot of time and attention.

The data for CA,PA,TX was summed based on whether or not the result rejected the null hypothesis or failed to reject it and further grouped by equal variance testing or unequal variance testing. The below tables(see supplements I) did not take into account the Measure itself, only whether the resulting p-value was higher or not providing an overall count table. And looking at these results paints a very different picture than looking at the more detailed results.

If the question is as simple as : Are mean values higher for States with more PLACES data, the conclusion is to reject the null hypothesis in favor of the alternative. In this case it might be better to err on the side of caution if a blanket decision is needed, since the alternative would mean a possible shortage of resources. In order to answer the question more in depth and have results be easily interpretable formatting was used on the result tables which will be in the supplementary section following the references. The columns consist of the propotions of each measure where the results were greater than or less than each alpha value. While there may be many results in the tables, this was the most straightforward way to provide results to the reader that were clear, concise and easy to interpret quickly. A simple calculation is provided below:

$$prop = \frac{val}{\sum p\ >0.05 + \sum p<0.05}$$

- Val is the value in question(sum $<0.05$ or sum $>0.05$)
- The sums of each occurrence
- Prop is the resulting proportion

For California in terms of both equal and unequal variances about 75% of measures were significantly different between all 3 years. (supplements A and B)

For Texas the total proportion of measures that were greater than average ended up being about 60% for all 3 year spans.(Supplements C and D)

For Pennsylvania the proportions were about even, 45% for the first year, 40% for the second year and 50% for the third year.(Supplements E and F)

# 7 Bibliography

"Places: Local Data for Better Health." Centers for Disease Control and Prevention, Centers for Disease Control and Prevention, 13 July 2023, www.cdc.gov/places/index.html/. [1]

"Free Textbooks Online with No Catch ." OpenStax, openstax.org/details/books/introductory-statistics. Accessed 29 Nov. 2023. [2]

"Places: Local Data for Better Health.The paper" Centers for Disease Control and Prevention, Centers for Disease Control and Prevention, 16 June 2022, www.cdc.gov/pcd/issues/2022/21_0459.htm. [3]

Ghasemi, Asghar, and Saleh Zahediasl. "Normality Tests for Statistical Analysis: A Guide for Non-Statisticians." International Journal of Endocrinology and Metabolism, U.S. National Library of Medicine, 2012, www.ncbi.nlm.nih.gov/pmc/articles/PMC3693611/. [4]

"Scipy.Stats.Kruskal." Scipy.Stats.Kruskal SciPy v1.11.4 Manual, docs.scipy.org/doc/scipy/reference/generated/scipy.stats.kruskal.html. Accessed 29 Nov. 2023. [5]

"Kruskal Wallis." Kruskal-Wallis Test,

www.itl.nist.gov/div898/software/dataplot/refman1/auxillar/kruskwal.htm. Accessed 29 Nov. 2023. [6]

"3.2 - Identifying Outliers: IQR Method: Stat 200." PennState: Statistics Online Courses, online.stat.psu.edu/stat200/lesson/3/3.2. Accessed 29 Nov. 2023. [7]

"Levene's Test." Wikipedia, Wikimedia Foundation, 28 June 2023, en.wikipedia.org/wiki/Levene%27s_test. [8]

"Student's t-Test." Wikipedia, Wikimedia Foundation, 28 Nov. 2023, en.wikipedia.org/wiki/Student%27s_t-test. [9]

"Scipy.Stats.Kruskal." Scipy.Stats.Kruskal - SciPy v1.11.4 Manual, docs.scipy.org/doc/scipy/reference/generated/scipy.stats.kruskal.html. Accessed 29 Nov. 2023. [10]

"Places: Local Data for Better Health, Place Data 2022 Release." Catalog, Publisher Centers for Disease Control and Prevention, 26 Aug. 2023, catalog.data.gov/dataset/places-local-data-for-better-health-place-data-2022-release. [11]

Nonparametric Pairwise Multiple Comparisons in Independent Groups Using Dunn's Test, NIST, 1 Jan. 2015, journals.sagepub.com/doi/pdf/10.1177/1536867X1501500117. [15]

"12.3 - Highly Correlated Predictors: Stat 501." PennState: Statistics Online Courses, online.stat.psu.edu/stat501/lesson/12/12.3. Accessed 29 Nov. 2023. [16]

# 8 parametric testing supplemental figures I

| equal_variance | p-value >0.05 2017-2018 | p-value <0.05 2017-2018 | p-value >0.05 2018-2019 | p-value <0.05 2018-2019 | p-value >0.05 2019-2020 | p-value <0.05 2019-2020 |
|---|---|---|---|---|---|---|
| TX | 199 | 116 | 212 | 168 | 204 | 107 |
| CA | 213 | 35 | 201 | 61 | 243 | 50 |
| PA | 134 | 195 | 185 | 216 | 151 | 201 |

Table 3: Totals for rejecting or failing to reject the null hypothesis (means of paired values being compared were the same). Texas showed fairly even splitting throughout. For California The failiure to reject hovered around 1/3rd or so. Pennsylvania however did have more than half of the results fall into the fail to reject category.

| unequal_variance | p-value >0.05 2017-2018 | p-value <0.05 2017-2018 | p-value >0.05 2018-2019 | p-value <0.05 2018-2019 | p-value >0.05 2019-2020 | p-value <0.05 2019-2020 |
|---|---|---|---|---|---|---|
| TX | 568 | 484 | 576 | 468 | 550 | 595 |
| CA | 740 | 383 | 802 | 363 | 789 | 373 |
| PA | 552 | 486 | 498 | 532 | 547 | 560 |

Table 4: For unequal variances the split was much closer between the counts of rejecting or failing to reject the null hypothesis. A few categories were quite close, with an almost even split.

| | p-value_>0.05_2017-2018 | p-value_<0.05_2017-2018 | p-value_>0.05_2018-2019 | p-value_<0.05_2018-2019 | p-value_>0.05_2019-2020 | p-value_<0.05_2019-2020 |
|---|---|---|---|---|---|---|
| 0 | nan | 1.000000 | nan | 1.000000 | 0.250000 | 0.750000 |
| 1 | nan | 1.000000 | nan | 1.000000 | nan | 1.000000 |
| 2 | 0.631579 | 0.368421 | 0.500000 | 0.500000 | 0.400000 | 0.600000 |
| 3 | nan | nan | 0.111111 | 0.888889 | nan | 1.000000 |
| 4 | 0.500000 | 0.500000 | nan | 1.000000 | nan | 1.000000 |
| 5 | 1.000000 | nan | 1.000000 | nan | 1.000000 | nan |
| 6 | 0.076923 | 0.923077 | nan | 1.000000 | 0.100000 | 0.900000 |
| 7 | nan | 1.000000 | nan | 1.000000 | nan | 1.000000 |
| 8 | nan | 1.000000 | nan | 1.000000 | 0.111111 | 0.888889 |
| 9 | 0.428571 | 0.571429 | nan | 1.000000 | nan | 1.000000 |
| 10 | 1.000000 | nan | nan | nan | 0.333333 | 0.666667 |
| 11 | 0.076923 | 0.923077 | nan | 1.000000 | 0.066667 | 0.933333 |
| 12 | 0.500000 | 0.500000 | nan | nan | 0.500000 | 0.500000 |
| 13 | 0.166667 | 0.833333 | nan | 1.000000 | 0.083333 | 0.916667 |
| 14 | nan | 1.000000 | nan | nan | 0.500000 | 0.500000 |
| 15 | nan | 1.000000 | nan | nan | 1.000000 | nan |
| 16 | nan | 1.000000 | 0.071429 | 0.928571 | 0.166667 | 0.833333 |
| 17 | 0.350000 | 0.650000 | 0.411765 | 0.588235 | 0.400000 | 0.600000 |
| 18 | 0.333333 | 0.666667 | 0.444444 | 0.555556 | 0.500000 | 0.500000 |
| 19 | 0.400000 | 0.600000 | nan | 1.000000 | 0.428571 | 0.571429 |
| 20 | 0.250000 | 0.750000 | nan | 1.000000 | nan | 1.000000 |
| 21 | nan | 1.000000 | nan | 1.000000 | 0.100000 | 0.900000 |
| 22 | nan | 1.000000 | 0.111111 | 0.888889 | 0.111111 | 0.888889 |
| 23 | nan | 1.000000 | nan | 1.000000 | nan | 1.000000 |
| 24 | nan | 1.000000 | 0.666667 | 0.333333 | 0.333333 | 0.666667 |
| 25 | 0.384615 | 0.615385 | nan | 1.000000 | 0.500000 | 0.500000 |
| 26 | nan | 1.000000 | 0.125000 | 0.875000 | 0.285714 | 0.714286 |
| 27 | 0.142857 | 0.857143 | 0.111111 | 0.888889 | 0.142857 | 0.857143 |
| 28 | 0.333333 | 0.666667 | 0.750000 | 0.250000 | 0.363636 | 0.636364 |
| 29 | 0.117647 | 0.882353 | 0.214286 | 0.785714 | 0.400000 | 0.600000 |

Figure 7: California t-test results for equal variances

# 9 parametric testing supplemental figures II

| | p-value_>0.05_2017-2018 | p-value_<0.05_2017-2018 | p-value_>0.05_2018-2019 | p-value_<0.05_2018-2019 | p-value_>0.05_2019-2020 | p-value_<0.05_2019-2020 |
|---|---|---|---|---|---|---|
| 0 | 0.219512 | 0.780488 | 0.209302 | 0.790698 | 0.195122 | 0.804878 |
| 1 | 0.166667 | 0.833333 | 0.032258 | 0.967742 | 0.027027 | 0.972973 |
| 2 | 0.666667 | 0.333333 | 0.702703 | 0.297297 | 0.526316 | 0.473684 |
| 3 | 0.081633 | 0.918367 | 0.076923 | 0.923077 | 0.081081 | 0.918919 |
| 4 | 0.446809 | 0.553191 | 0.319149 | 0.680851 | 0.250000 | 0.750000 |
| 5 | 0.604651 | 0.395349 | 0.604651 | 0.395349 | 0.595238 | 0.404762 |
| 6 | 0.611111 | 0.388889 | 0.658537 | 0.341463 | 0.657895 | 0.342105 |
| 7 | 0.432432 | 0.567568 | 0.242424 | 0.757576 | 0.166667 | 0.833333 |
| 8 | 0.322581 | 0.677419 | 0.212121 | 0.787879 | 0.166667 | 0.833333 |
| 9 | 0.439024 | 0.560976 | 0.279070 | 0.720930 | 0.166667 | 0.833333 |
| 10 | 0.531915 | 0.468085 | 0.551020 | 0.448980 | 0.589744 | 0.410256 |
| 11 | 0.027778 | 0.972222 | 0.055556 | 0.944444 | 0.030303 | 0.969697 |
| 12 | 0.025641 | 0.974359 | nan | nan | 0.125000 | 0.875000 |
| 13 | 0.488372 | 0.511628 | 0.431818 | 0.568182 | 0.527778 | 0.472222 |
| 14 | 0.645833 | 0.354167 | nan | nan | 0.547619 | 0.452381 |
| 15 | nan | 1.000000 | 0.653061 | 0.346939 | 0.652174 | 0.347826 |
| 16 | 0.111111 | 0.888889 | 0.171429 | 0.828571 | 0.111111 | 0.888889 |
| 17 | 0.259259 | 0.740741 | 0.125000 | 0.875000 | 0.214286 | 0.785714 |
| 18 | 0.194444 | 0.805556 | 0.625000 | 0.375000 | 0.647059 | 0.352941 |
| 19 | 0.465116 | 0.534884 | 0.340426 | 0.659574 | 0.200000 | 0.800000 |
| 20 | 0.288889 | 0.711111 | 0.200000 | 0.800000 | 0.204545 | 0.795455 |
| 21 | 0.204545 | 0.795455 | 0.111111 | 0.888889 | 0.078947 | 0.921053 |
| 22 | 0.078947 | 0.921053 | 0.300000 | 0.700000 | 0.307692 | 0.692308 |
| 23 | 0.058824 | 0.941176 | 0.690476 | 0.309524 | 0.666667 | 0.333333 |
| 24 | 0.590909 | 0.409091 | 0.586957 | 0.413043 | 0.400000 | 0.600000 |
| 25 | 0.305556 | 0.694444 | 0.225000 | 0.775000 | 0.173913 | 0.826087 |
| 26 | 0.428571 | 0.571429 | 0.292683 | 0.707317 | 0.264706 | 0.735294 |
| 27 | 0.058824 | 0.941176 | 0.032258 | 0.967742 | 0.058824 | 0.941176 |
| 28 | 0.413043 | 0.586957 | 0.377778 | 0.622222 | 0.405405 | 0.594595 |
| 29 | 0.031250 | 0.968750 | 0.114286 | 0.885714 | 0.119048 | 0.880952 |

Figure 8: California t-test results for unequal variances

| | p-value_>0.05_2017-2018 | p-value_<0.05_2017-2018 | p-value_>0.05_2018-2019 | p-value_<0.05_2018-2019 | p-value_>0.05_2019-2020 | p-value_<0.05_2019-2020 |
|---|---|---|---|---|---|---|
| 0 | 0.219512 | 0.780488 | 0.209302 | 0.790698 | 0.195122 | 0.804878 |
| 1 | 0.166667 | 0.833333 | 0.032258 | 0.967742 | 0.027027 | 0.972973 |
| 2 | 0.666667 | 0.333333 | 0.702703 | 0.297297 | 0.526316 | 0.473684 |
| 3 | 0.081633 | 0.918367 | 0.076923 | 0.923077 | 0.081081 | 0.918919 |
| 4 | 0.446809 | 0.553191 | 0.319149 | 0.680851 | 0.250000 | 0.750000 |
| 5 | 0.604651 | 0.395349 | 0.604651 | 0.395349 | 0.595238 | 0.404762 |
| 6 | 0.611111 | 0.388889 | 0.658537 | 0.341463 | 0.657895 | 0.342105 |
| 7 | 0.432432 | 0.567568 | 0.242424 | 0.757576 | 0.166667 | 0.833333 |
| 8 | 0.322581 | 0.677419 | 0.212121 | 0.787879 | 0.166667 | 0.833333 |
| 9 | 0.439024 | 0.560976 | 0.279070 | 0.720930 | 0.166667 | 0.833333 |
| 10 | 0.531915 | 0.468085 | 0.551020 | 0.448980 | 0.589744 | 0.410256 |
| 11 | 0.027778 | 0.972222 | 0.055556 | 0.944444 | 0.030303 | 0.969697 |
| 12 | 0.025641 | 0.974359 | nan | nan | 0.125000 | 0.875000 |
| 13 | 0.488372 | 0.511628 | 0.431818 | 0.568182 | 0.527778 | 0.472222 |
| 14 | 0.645833 | 0.354167 | nan | nan | 0.547619 | 0.452381 |
| 15 | nan | 1.000000 | 0.653061 | 0.346939 | 0.652174 | 0.347826 |
| 16 | 0.111111 | 0.888889 | 0.171429 | 0.828571 | 0.111111 | 0.888889 |
| 17 | 0.259259 | 0.740741 | 0.125000 | 0.875000 | 0.214286 | 0.785714 |
| 18 | 0.194444 | 0.805556 | 0.625000 | 0.375000 | 0.647059 | 0.352941 |
| 19 | 0.465116 | 0.534884 | 0.340426 | 0.659574 | 0.200000 | 0.800000 |
| 20 | 0.288889 | 0.711111 | 0.200000 | 0.800000 | 0.204545 | 0.795455 |
| 21 | 0.204545 | 0.795455 | 0.111111 | 0.888889 | 0.078947 | 0.921053 |
| 22 | 0.078947 | 0.921053 | 0.300000 | 0.700000 | 0.307692 | 0.692308 |
| 23 | 0.058824 | 0.941176 | 0.690476 | 0.309524 | 0.666667 | 0.333333 |
| 24 | 0.590909 | 0.409091 | 0.586957 | 0.413043 | 0.400000 | 0.600000 |
| 25 | 0.305556 | 0.694444 | 0.225000 | 0.775000 | 0.173913 | 0.826087 |
| 26 | 0.428571 | 0.571429 | 0.292683 | 0.707317 | 0.264706 | 0.735294 |
| 27 | 0.058824 | 0.941176 | 0.032258 | 0.967742 | 0.058824 | 0.941176 |
| 28 | 0.413043 | 0.586957 | 0.377778 | 0.622222 | 0.405405 | 0.594595 |
| 29 | 0.031250 | 0.968750 | 0.114286 | 0.885714 | 0.119048 | 0.880952 |

Figure 9: Texas t-test results for equal variances

| | p-value_>0.05_2017-2018 | p-value_<0.05_2017-2018 | p-value_>0.05_2018-2019 | p-value_<0.05_2018-2019 | p-value_>0.05_2019-2020 | p-value_<0.05_2019-2020 |
|---|---|---|---|---|---|---|
| 0 | 0.717949 | 0.282051 | 0.657895 | 0.342105 | 0.600000 | 0.400000 |
| 1 | 0.088235 | 0.911765 | 0.096774 | 0.903226 | 0.033333 | 0.966667 |
| 2 | 0.677419 | 0.322581 | 0.342857 | 0.657143 | 0.487805 | 0.512195 |
| 3 | 0.022222 | 0.977778 | 0.029412 | 0.970588 | 0.030303 | 0.969697 |
| 4 | 0.045455 | 0.954545 | 0.100000 | 0.900000 | 0.066667 | 0.933333 |
| 5 | 0.259259 | 0.740741 | 0.485714 | 0.514286 | 0.200000 | 0.800000 |
| 6 | 0.750000 | 0.250000 | 0.769231 | 0.230769 | 0.769231 | 0.230769 |
| 7 | 0.558140 | 0.441860 | 0.441860 | 0.558140 | 0.548387 | 0.451613 |
| 8 | 0.540541 | 0.459459 | 0.727273 | 0.272727 | 0.468750 | 0.531250 |
| 9 | 0.146341 | 0.853659 | 0.150000 | 0.850000 | 0.081081 | 0.918919 |
| 10 | 1.000000 | nan | 1.000000 | nan | 1.000000 | nan |
| 11 | 0.547619 | 0.452381 | 0.205128 | 0.794872 | 0.459459 | 0.540541 |
| 12 | 0.540541 | 0.459459 | nan | nan | 0.500000 | 0.500000 |
| 13 | 0.891304 | 0.108696 | 0.891304 | 0.108696 | 0.885714 | 0.114286 |
| 14 | 0.893617 | 0.106383 | nan | nan | 0.818182 | 0.181818 |
| 15 | 0.193548 | 0.806452 | 0.088235 | 0.911765 | 0.081081 | 0.918919 |
| 16 | 0.717949 | 0.282051 | 0.714286 | 0.285714 | 0.685714 | 0.314286 |
| 17 | 0.794118 | 0.205882 | 0.709677 | 0.290323 | 0.766667 | 0.233333 |
| 18 | 0.277778 | 0.722222 | 0.315789 | 0.684211 | 0.297297 | 0.702703 |
| 19 | 0.729730 | 0.270270 | 0.566667 | 0.433333 | 0.571429 | 0.428571 |
| 20 | 0.844444 | 0.155556 | 0.772727 | 0.227273 | 0.767442 | 0.232558 |
| 21 | 0.844444 | 0.155556 | 0.891304 | 0.108696 | 0.900000 | 0.100000 |
| 22 | 0.131579 | 0.868421 | 0.023810 | 0.976190 | 0.046512 | 0.953488 |
| 23 | 0.100000 | 0.900000 | 0.048780 | 0.951220 | 0.048780 | 0.951220 |
| 24 | 0.790698 | 0.209302 | 0.700000 | 0.300000 | 0.666667 | 0.333333 |
| 25 | 0.631579 | 0.368421 | 0.447368 | 0.552632 | 0.451613 | 0.548387 |
| 26 | 0.578947 | 0.421053 | 0.485714 | 0.514286 | 0.500000 | 0.500000 |
| 27 | 0.500000 | 0.500000 | 0.410256 | 0.589744 | 0.487805 | 0.512195 |
| 28 | nan | 1.000000 | 0.119048 | 0.880952 | 0.111111 | 0.888889 |
| 29 | 0.218750 | 0.781250 | 0.178571 | 0.821429 | 0.314286 | 0.685714 |

Figure 10: Texas t-test results for unequal variances

| | p-value_>0.05_2017-2018 | p-value_<0.05_2017-2018 | p-value_>0.05_2018-2019 | p-value_<0.05_2018-2019 | p-value_>0.05_2019-2020 | p-value_<0.05_2019-2020 |
|---|---|---|---|---|---|---|
| 0 | 0.500000 | 0.500000 | 0.800000 | 0.200000 | 0.500000 | 0.500000 |
| 1 | 0.684211 | 0.315789 | 0.555556 | 0.444444 | 0.695652 | 0.304348 |
| 2 | 0.400000 | 0.600000 | 0.666667 | 0.333333 | 0.846154 | 0.153846 |
| 3 | 0.350000 | 0.650000 | 0.500000 | 0.500000 | 0.437500 | 0.562500 |
| 4 | 0.520000 | 0.480000 | 0.277778 | 0.722222 | 0.375000 | 0.625000 |
| 5 | 0.722222 | 0.277778 | 0.416667 | 0.583333 | 0.400000 | 0.600000 |
| 6 | 0.500000 | 0.500000 | 0.625000 | 0.375000 | 0.500000 | 0.500000 |
| 7 | 0.875000 | 0.125000 | 0.714286 | 0.285714 | 0.470588 | 0.529412 |
| 8 | 0.500000 | 0.500000 | 0.833333 | 0.166667 | 0.428571 | 0.571429 |
| 9 | 0.500000 | 0.500000 | 0.666667 | 0.333333 | 0.500000 | 0.500000 |
| 10 | 0.375000 | 0.625000 | 0.250000 | 0.750000 | 0.153846 | 0.846154 |
| 11 | 0.500000 | 0.500000 | 0.333333 | 0.666667 | 0.764706 | 0.235294 |
| 12 | 0.555556 | 0.444444 | nan | nan | 0.500000 | 0.500000 |
| 13 | 0.818182 | 0.181818 | 0.750000 | 0.250000 | 0.666667 | 0.333333 |
| 14 | 0.636364 | 0.363636 | nan | nan | 0.461538 | 0.538462 |
| 15 | 0.384615 | 0.615385 | 0.533333 | 0.466667 | 0.545455 | 0.454545 |
| 16 | 0.666667 | 0.333333 | 0.692308 | 0.307692 | 0.600000 | 0.400000 |
| 17 | 0.600000 | 0.400000 | 0.461538 | 0.538462 | 0.529412 | 0.470588 |
| 18 | 0.555556 | 0.444444 | 0.545455 | 0.454545 | 0.333333 | 0.666667 |
| 19 | 0.500000 | 0.500000 | 0.562500 | 0.437500 | 0.476190 | 0.523810 |
| 20 | 0.571429 | 0.428571 | 0.555556 | 0.444444 | 0.636364 | 0.363636 |
| 21 | 0.636364 | 0.363636 | 0.636364 | 0.363636 | 0.625000 | 0.375000 |
| 22 | 0.333333 | 0.666667 | 1.000000 | nan | 0.818182 | 0.181818 |
| 23 | 0.562500 | 0.437500 | 0.857143 | 0.142857 | 0.812500 | 0.187500 |
| 24 | 0.666667 | 0.333333 | 0.833333 | 0.166667 | 0.454545 | 0.545455 |
| 25 | 1.000000 | nan | 0.909091 | 0.090909 | 0.500000 | 0.500000 |
| 26 | 0.222222 | 0.777778 | 0.461538 | 0.538462 | 0.444444 | 0.555556 |
| 27 | 0.727273 | 0.272727 | 0.600000 | 0.400000 | 0.692308 | 0.307692 |
| 28 | 0.300000 | 0.700000 | 0.500000 | 0.500000 | 0.538462 | 0.461538 |
| 29 | 1.000000 | nan | 0.600000 | 0.400000 | 0.500000 | 0.500000 |

Figure 11: Pennsylvania t-test results for equal variances

| | p-value_>0.05_2017-2018 | p-value_<0.05_2017-2018 | p-value_>0.05_2018-2019 | p-value_<0.05_2018-2019 | p-value_>0.05_2019-2020 | p-value_<0.05_2019-2020 |
|---|---|---|---|---|---|---|
| 0 | 0.487805 | 0.512195 | 0.409091 | 0.590909 | 0.459459 | 0.540541 |
| 1 | 0.800000 | 0.200000 | 0.709677 | 0.290323 | 0.880000 | 0.120000 |
| 2 | 0.840909 | 0.159091 | 0.775000 | 0.225000 | 0.657143 | 0.342857 |
| 3 | 0.724138 | 0.275862 | 0.566667 | 0.433333 | 0.531250 | 0.468750 |
| 4 | 0.708333 | 0.291667 | 0.451613 | 0.548387 | 0.451613 | 0.548387 |
| 5 | 0.400000 | 0.600000 | 0.432432 | 0.567568 | 0.552632 | 0.447368 |
| 6 | 0.302326 | 0.697674 | 0.219512 | 0.780488 | 0.238095 | 0.761905 |
| 7 | 0.536585 | 0.463415 | 0.452381 | 0.547619 | 0.548387 | 0.451613 |
| 8 | 0.343750 | 0.656250 | 0.189189 | 0.810811 | 0.242424 | 0.757576 |
| 9 | 0.459459 | 0.540541 | 0.594595 | 0.405405 | 0.878788 | 0.121212 |
| 10 | 0.170732 | 0.829268 | 0.066667 | 0.933333 | 0.142857 | 0.857143 |
| 11 | 0.609756 | 0.390244 | 0.675000 | 0.325000 | 0.612903 | 0.387097 |
| 12 | 0.578947 | 0.421053 | nan | nan | 0.809524 | 0.190476 |
| 13 | 0.263158 | 0.736842 | 0.268293 | 0.731707 | 0.128205 | 0.871795 |
| 14 | 0.351351 | 0.648649 | nan | nan | 0.485714 | 0.514286 |
| 15 | 0.944444 | 0.055556 | 0.941176 | 0.058824 | 0.894737 | 0.105263 |
| 16 | 0.250000 | 0.750000 | 0.138889 | 0.861111 | 0.181818 | 0.818182 |
| 17 | 0.242424 | 0.757576 | 0.277778 | 0.722222 | 0.258065 | 0.741935 |
| 18 | 0.483871 | 0.516129 | 0.421053 | 0.578947 | 0.465116 | 0.534884 |
| 19 | 0.722222 | 0.277778 | 0.666667 | 0.333333 | 0.615385 | 0.384615 |
| 20 | 0.380952 | 0.619048 | 0.375000 | 0.625000 | 0.378378 | 0.621622 |
| 21 | 0.342105 | 0.657895 | 0.289474 | 0.710526 | 0.400000 | 0.600000 |
| 22 | 0.375000 | 0.625000 | 0.750000 | 0.250000 | 0.837838 | 0.162162 |
| 23 | 0.636364 | 0.363636 | 0.685714 | 0.314286 | 0.709677 | 0.290323 |
| 24 | 0.425000 | 0.575000 | 0.348837 | 0.651163 | 0.461538 | 0.538462 |
| 25 | 0.729730 | 0.270270 | 0.710526 | 0.289474 | 0.825000 | 0.175000 |
| 26 | 0.375000 | 0.625000 | 0.277778 | 0.722222 | 0.333333 | 0.666667 |
| 27 | 0.378378 | 0.621622 | 0.363636 | 0.636364 | 0.342857 | 0.657143 |
| 28 | 0.564103 | 0.435897 | 0.585366 | 0.414634 | 0.588235 | 0.411765 |
| 29 | 0.790698 | 0.209302 | 0.655172 | 0.344828 | 0.780488 | 0.219512 |

Figure 12: Pennsylvania t-test results for unequal variances