## Introduction

Dave Harper is the quintessential New Yorker; he loves his state. There is no corner in the state of New York where he hasn’t been already, he knows the best places for the perfect occasions. Mr. Harper is known everywhere in the state because of the reputation he earned through the years, he even got a nickname from all the knowledge he has about New York, he is called “York”. He is faster than some platforms to recommend the same places to go and enjoy.

## Business Problem

Mr. Harper finally managed to get the job of his dreams as a Data Scientist in one of the top Video Game companies of the century. The job has one bad side, he needs to relocate to one of the available offices distributed around the globe. The states with available openings are:

* California
* Nevada
* Washington
* Florida

Since Dave is not going to let this opportunity of a lifetime go to waste, he decided to move to one of the proposed states with one condition: “The State needs to be the most similar to New York”. The employer could not answer his question, so he decided to use his amazing knowledge of New York and Data Science skills to pick a territory that will become his new “New York”.

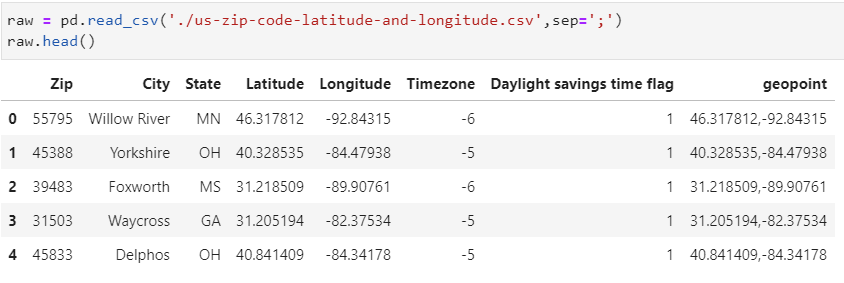
Dave asked the employer for a couple of weeks so he could do a more detailed analysis, but the Video Game company needed an answer on the next two days, otherwise, they would consider another person for the role. Mr. Harper did not have much time to transform all his knowledge into a numeric form to measure or compare all the cities within each region, so he decided to get the data from the platform he knew it was the most similar to his knowledge to NY, the name of the platform is ‘Foursquare’.

After a couple of hours thinking, Dave defined the KPI’s in which he would base his analysis to conclude that a state is ‘similar’ to New York:

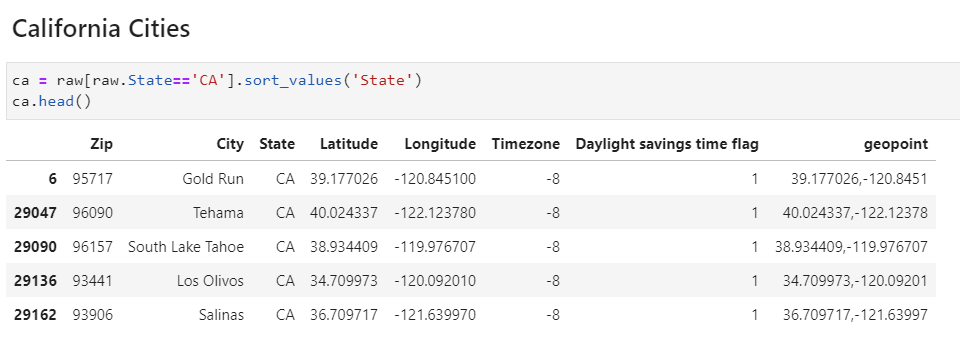
* Number of Similar Clusters of Zip Codes mapped with categories
* Number of Differences between clusters of other states and NY
* Number of total likes gathered within all the venues in comparison of NY likes

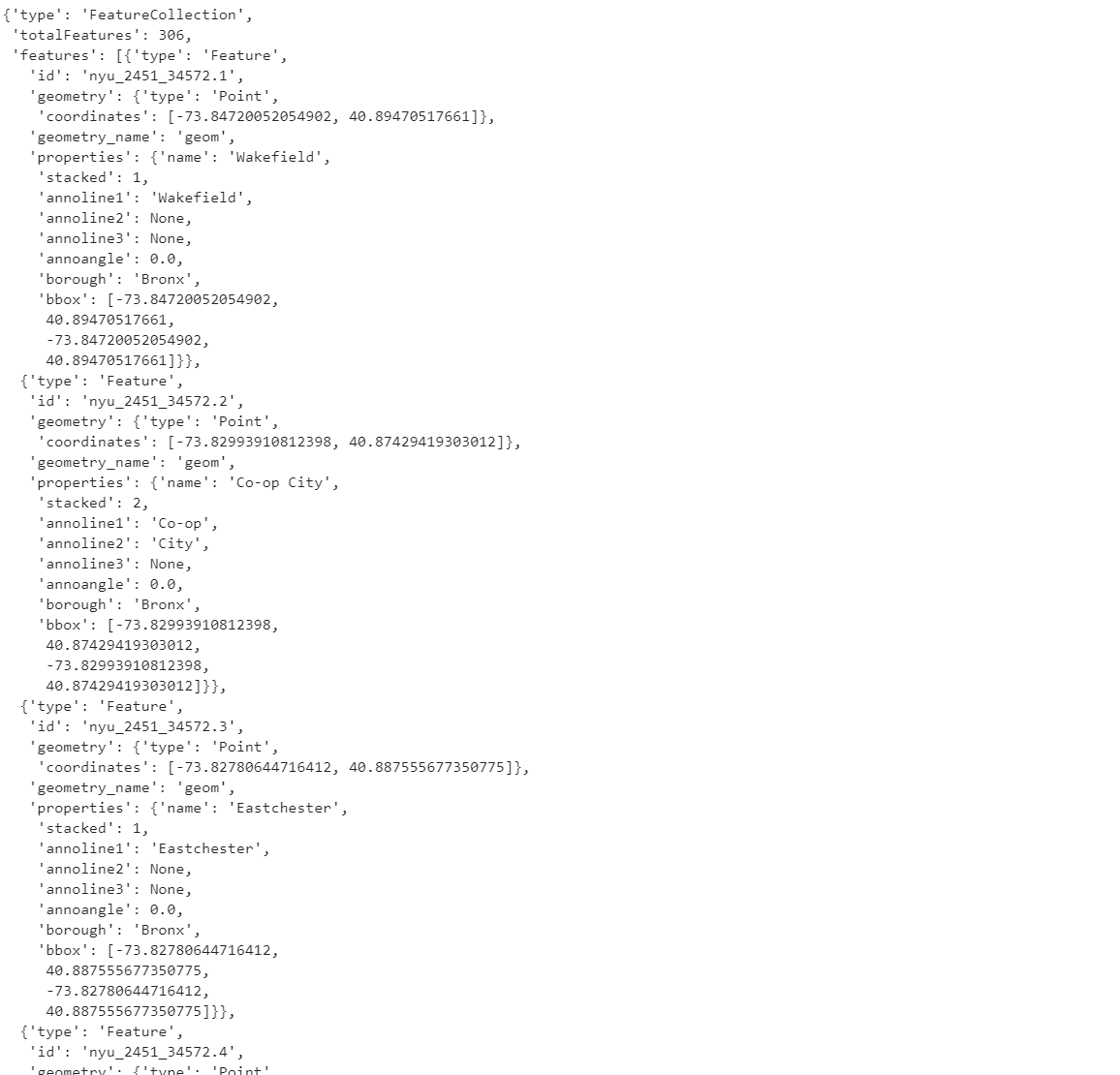
## **Data Sources**

The data that will be used for this project was extracted from public.opendatasoft.com and foursquare.com. The first dataset contains data about all of the geographical locations in the US, it has the following columns: Zip, City, State, Latitude, Longitude, Timezone, Daylight savings time flag and geopoint.



The main dataset will be divided into five datasets which will contain the data from states of: New York, California, Nevada, Washington, and Florida. For example:



After the division, the dataset will be merged with data from the available venues in the area for each zip code. The data extracted from Foursquare using the REST-API which comes in JSON format, which can be cleaned and put inside a pandas data frame for better handling. Example of JSON data extracted for a random zip code:

## Methodology

The process to compare the venues of each state using the Zip codes will need several KPI’s to determine which state is the most similar. In this analysis three KPI will be used to determine if a state is similar:

* Number of Similar Clusters of Zip Codes mapped with categories
* Number of Differences between clusters of other states and NY
* Number of total likes gathered within all the venues in comparison of NY likes

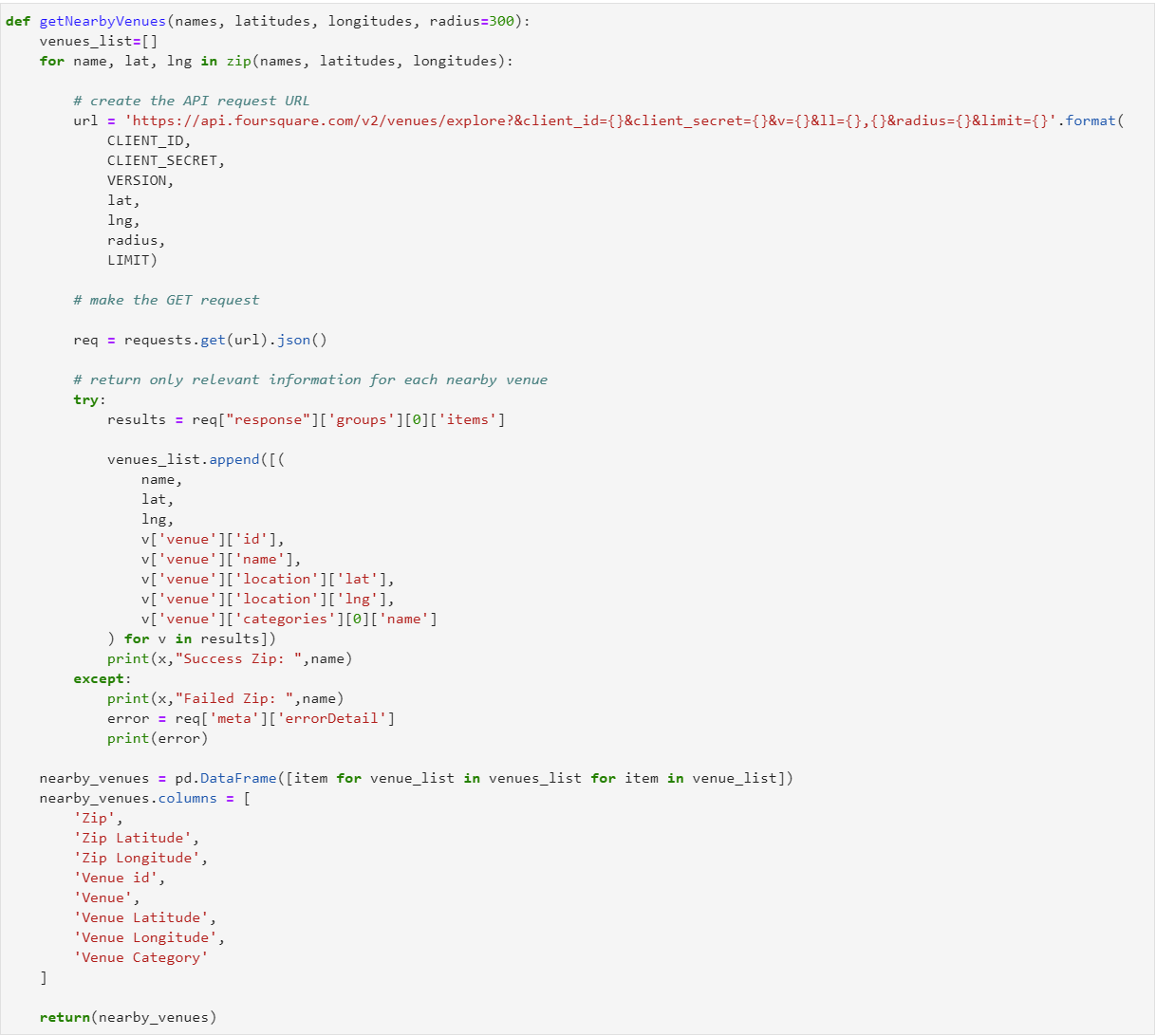
To extract all the venues that are inside the Json file there has to be done in a process in which takes all the relevant information for each venue such as Zip, Zip Latitude, Zip Longitude, Venue Id, Venue Name, Venue Latitude, Venue Longitude, and Venue Category. The process to query to API is the following:

First, create a function named “getNearbyVenues” to create the URL which will get the data from Foursquare using the endpoint “explore” to look for available venues within the latitude and longitude of the zip code in a radius of 300 meters wide.

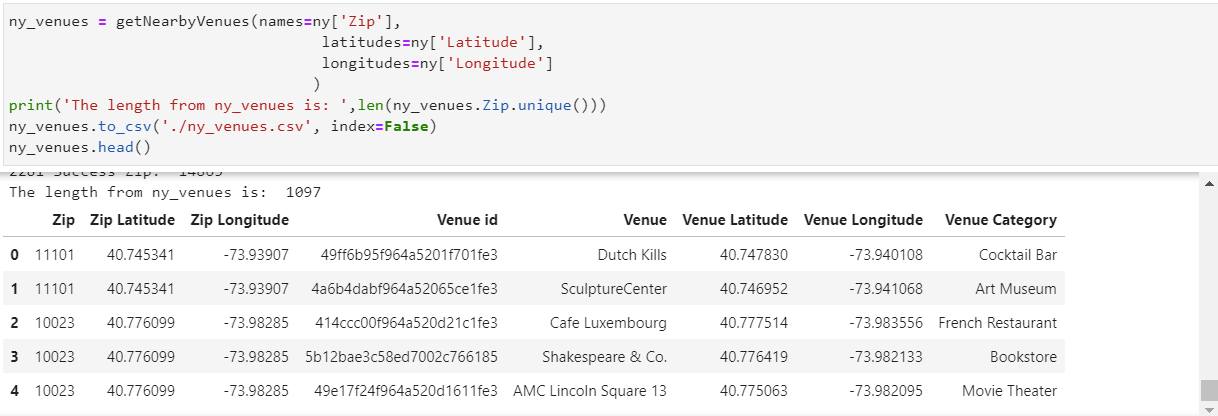


Second, use exception handling to prevent execution errors to be added to our data. This precaution was taken into consideration because there were scenarios in which the user did not know which zip codes where not getting a single venue. There were several zip codes without venues because of diverse reasons such as: residential areas, office areas, wild zones, etcetera.

In case the result contained at least one venue, the function is able to extract it and put it into a list, with their related zip, latitude and longitude information that came from our main data set. At the end the list was transformed into a pandas data frame and returned to the user.

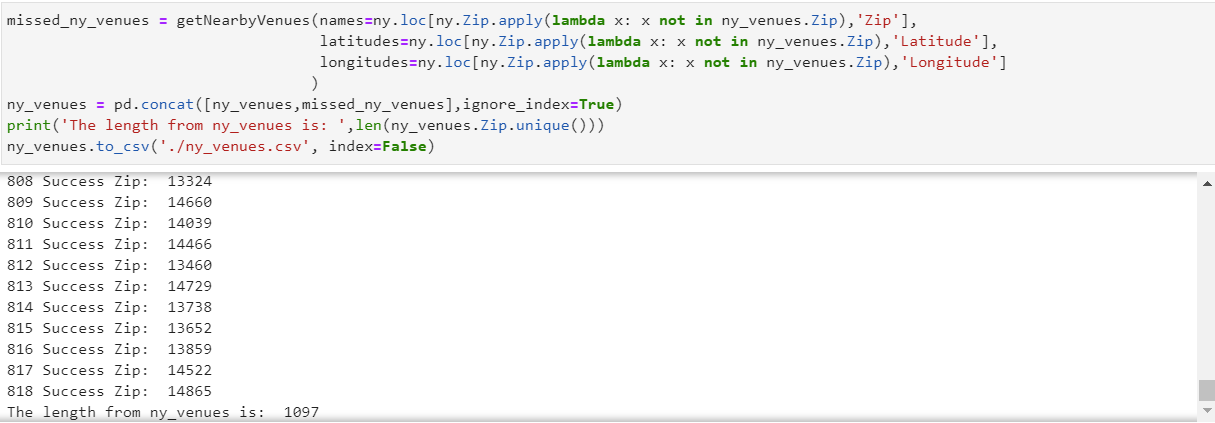


Example of how the function was used to extract all the venues from the state of New York, and the results which were returned by Foursquare:

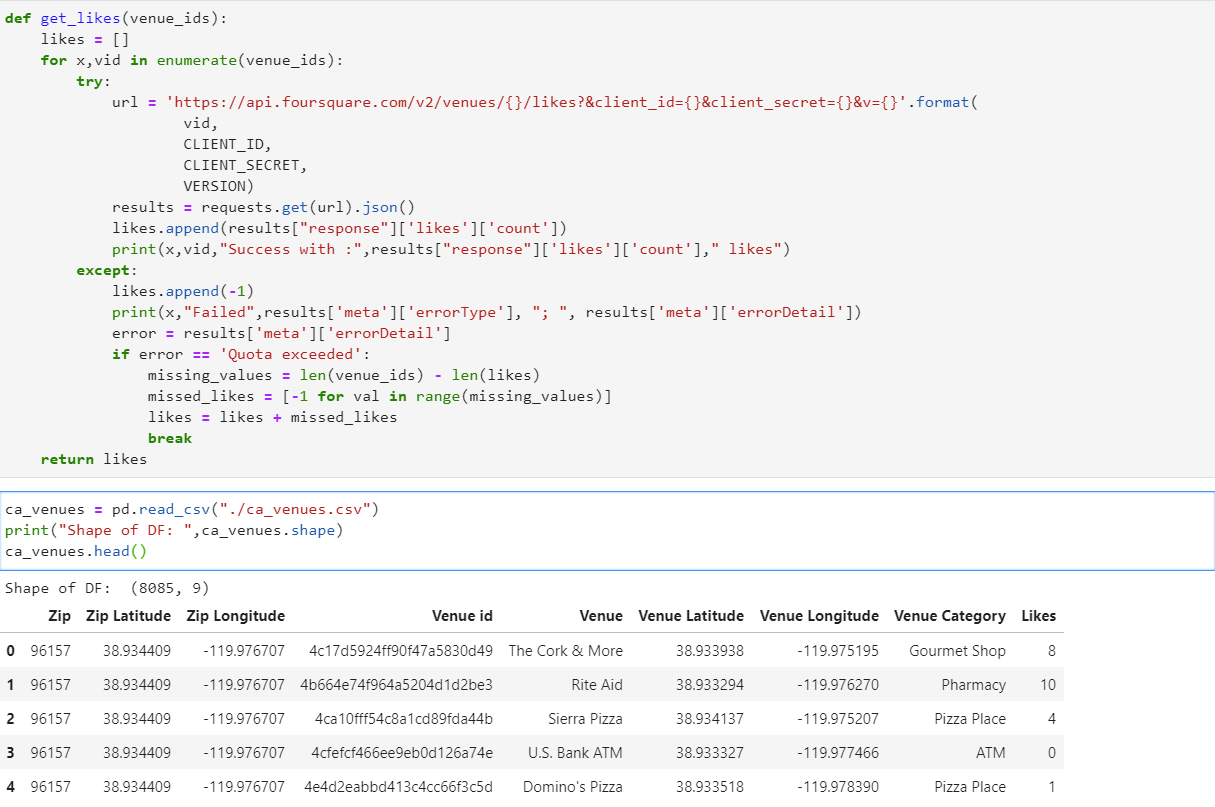


The function was used with the rest of the states because the study needs available data for all venues in California, Washington, Nevada, and Florida. Several setbacks were experienced within this step of the process due to Foursquare having a quota limit of five thousand queries per hour.

The first results of the function were saved into a csv to save the progress and save quota for more further queries to the API. The next execution of the code, the zip codes that were already inside the csv file were excluded using a lambda function. Afterwards, the missed venues from the previous run where combined with the one from the first run and saved within the same csv file. The user had to run the function several times due to quota.

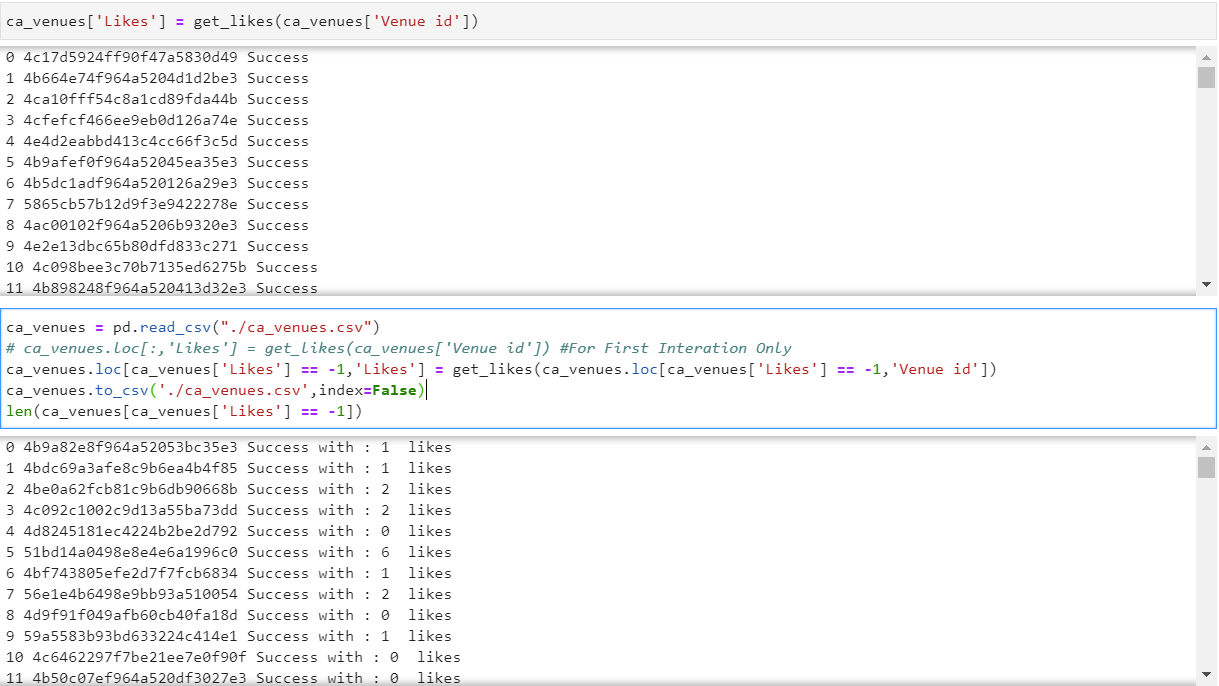


To query the likes from Foursquare it is needed to create a new function named “get\_likes” which it is very similar at the last one to get the venues. This time the function will use the endpoint “likes” that is inside the venue category to get all the likes that a venue has gotten through the years.

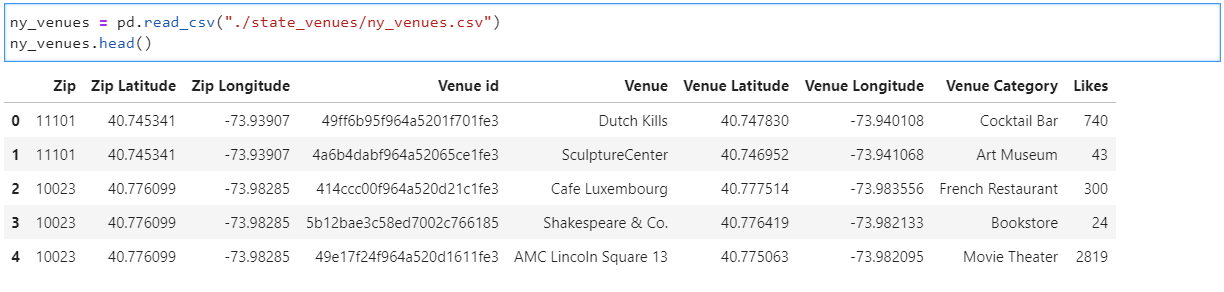


The function had several complications due to several states contained more than ten thousand venues. This was a big problem because the server only allowed five thousand requests of likes per hour, so the function had to be able to recognize whenever an error of “Quota exceeded” occurred.

When an error like that happened, the function stopped the loop and assigned “-1” to the remaining venues so later on the user could execute the next cell of code. The next block would save all the values and then create a column for the Likes (Only for the first iteration), then use the function get\_likes to get new values for all the venues that had exactly “-1” likes.

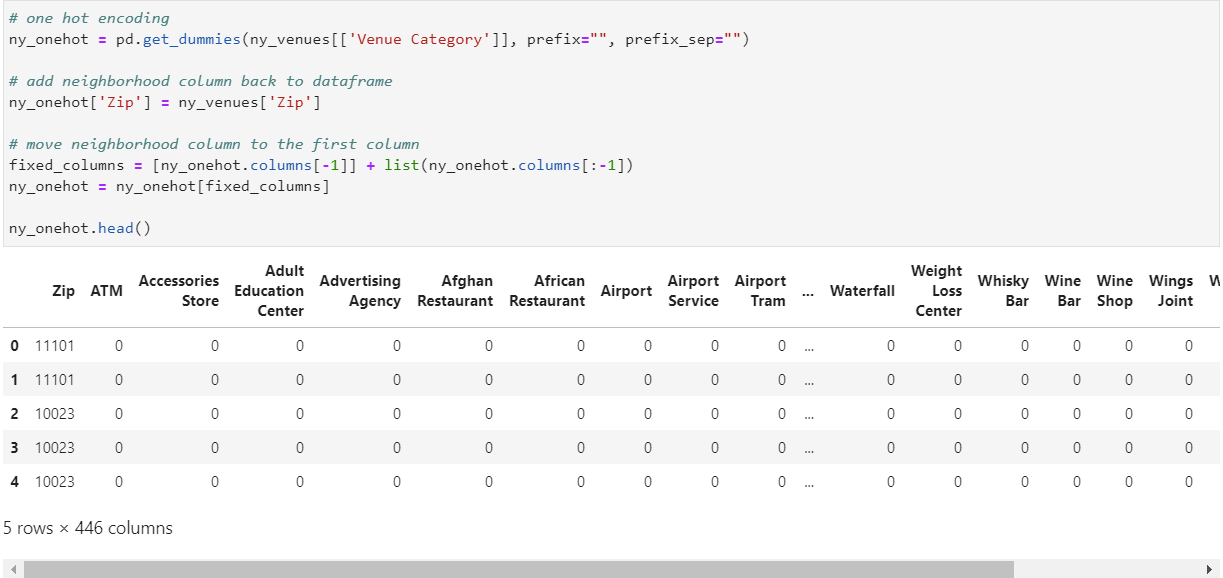


The results of the past two functions will generate a file just as the one shown below:

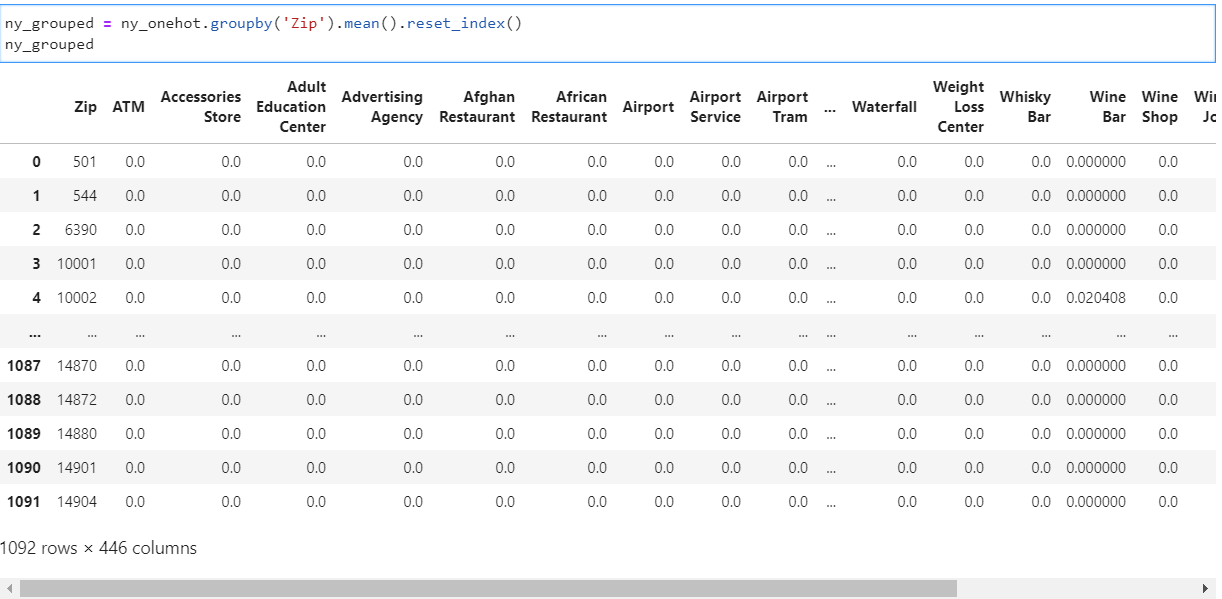


The comparison of all the categories of all the venues available New York is not easy task for a human being, with no mention that those comparisons need to be done as well for the zip codes in the other locations. The main idea is to compare all the available categories for each zip code, create clusters of them and then be able to use these same clusters standards to classify the zip codes by category of California, Washington, Nevada, and Florida using New York as our base. Finally, with that data calculated get which state contains the least differences between clusters so we can decide the state that is the most similar as New York.

Two machine learning models need to be created to be able to find which state is the new “New York” for Dave. First, it is needed to create cluster using the data of New York as our base, to accomplish this a k-means model will be used to get clusters for all the available venues in New York. This model will only take into consideration categorical variables, so they need to be replaced by dummy variables to be processed by the model.

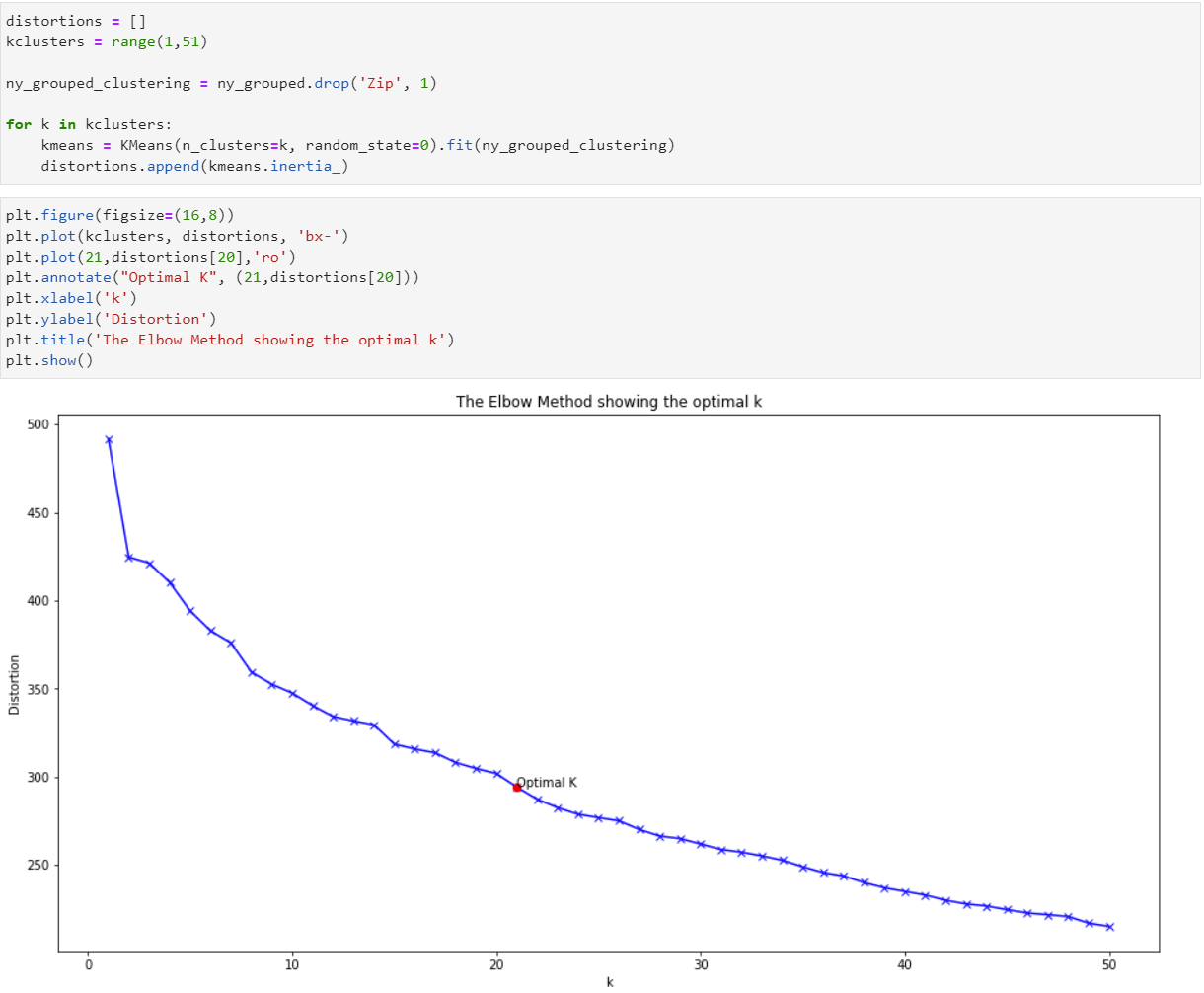


Second, after each possible value in the categorical column is transformed into a dummy variable, then it is needed to get the average appearance for the category for each zip code. To get that data the dataset was grouped by the zip code and extracted the average appearance for each category in the zone.

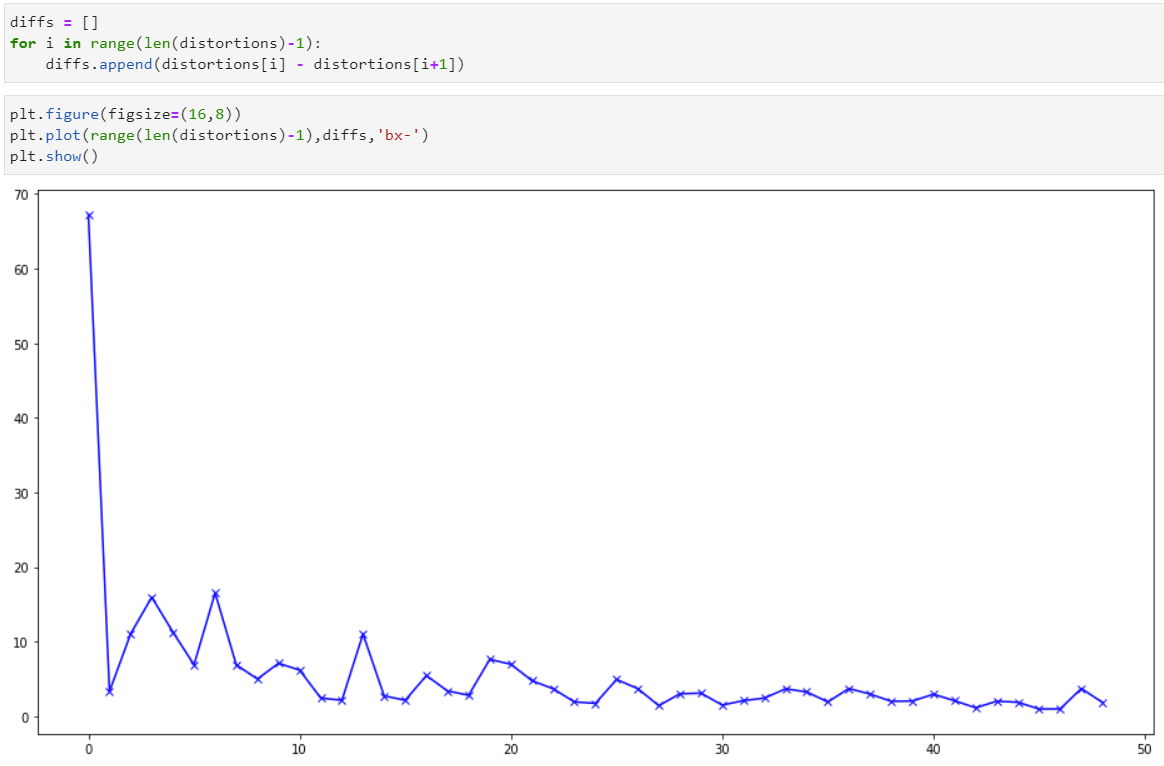


Third, it is needed to calculate the best K for the model. The best K can be calculated by several algorithms, in this specific project the Elbow Method was used. The elbow method uses the values of the inertia, or the sum of squared distances from each point to the centroid, to determine the best K.

Since there were four hundred and forty-five columns of categories, it must be check if it is needed to use many categories to distinguish a cluster from another one. A loop to extract all the inertia for a model with a k within the range of one to fifty was created, the results were plotted below:

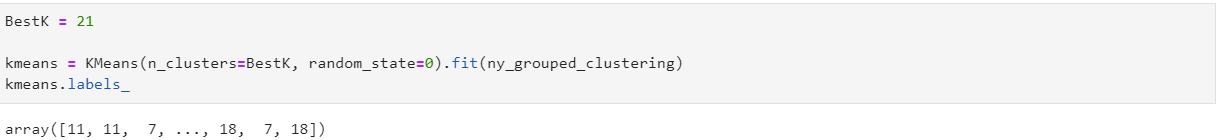


Also, a graph of residue or difference between each point of inertia was plotted to see the last greatest jump in terms of inertia because from the graph shown above it is hard to visualize the best K by eye.

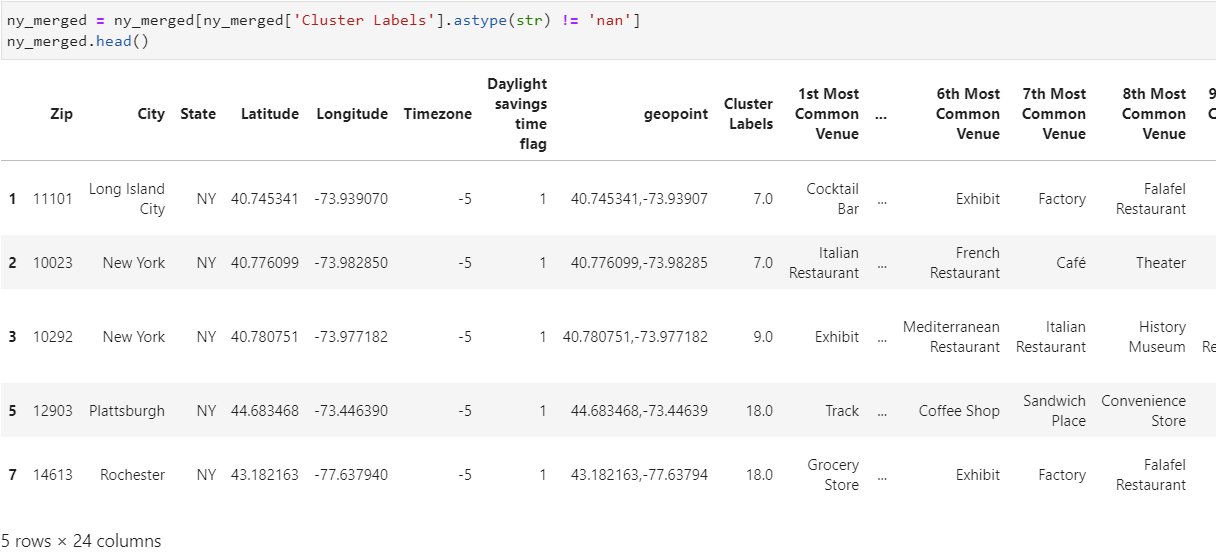


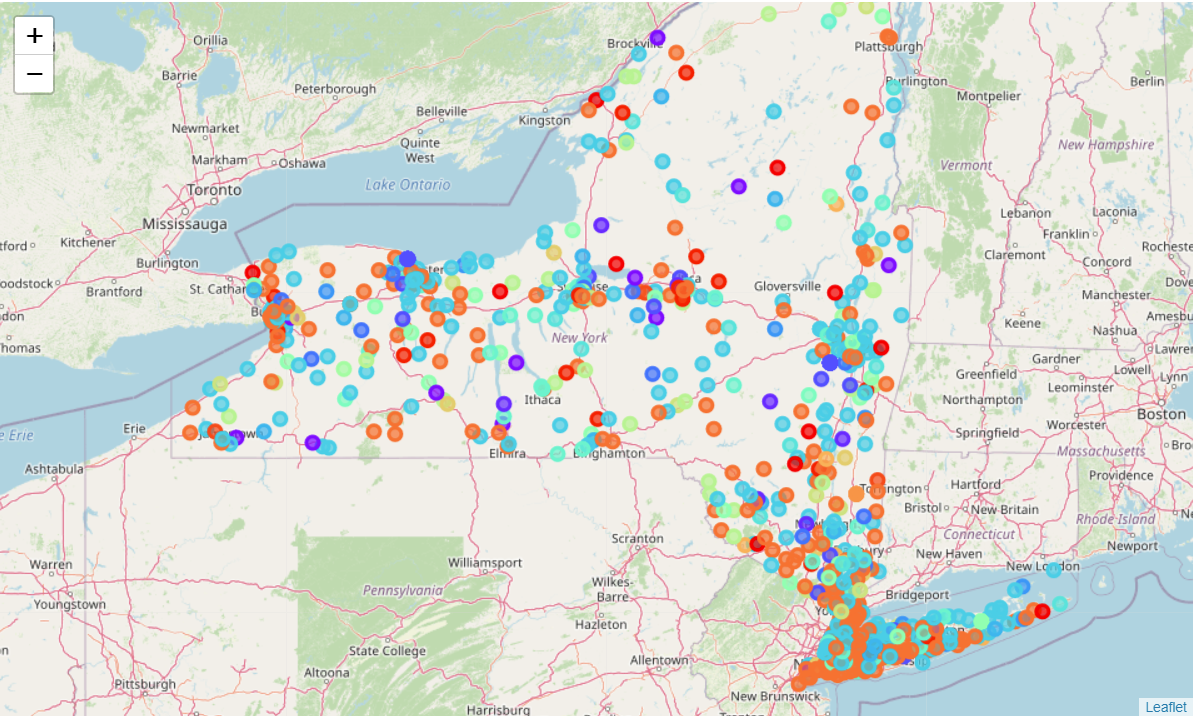
From the graph it is hard to see the exact point of the maximum decrease of change between the inertia. There were a couple of good candidates at first instance, like: four, seven, and thirteen. But all of them were discarded due to the same graph not trending to reach an equilibrium. The best K was chosen is twenty-one because after that point both graphs start to reach to a softer line.

The model with K equals to twenty-one is chosen. With the model created it is possible to get the labels to map each zip code of New York to a cluster. Since the scope of this project is only to decide which state is the most similar to NY, the clusters will have a name that provide context of what the cluster contains.

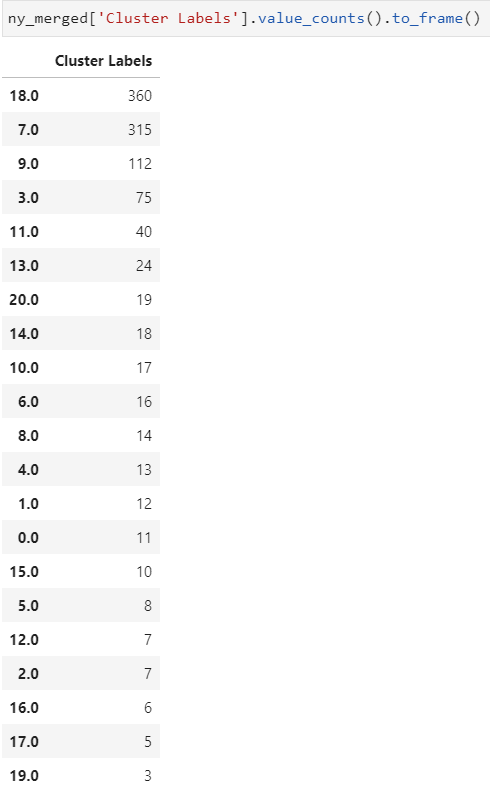


With the model created and labels generated, the next step is to include the cluster information into the main dataset for the venues of New York and have a plot of the diverse zip codes with their respected cluster.



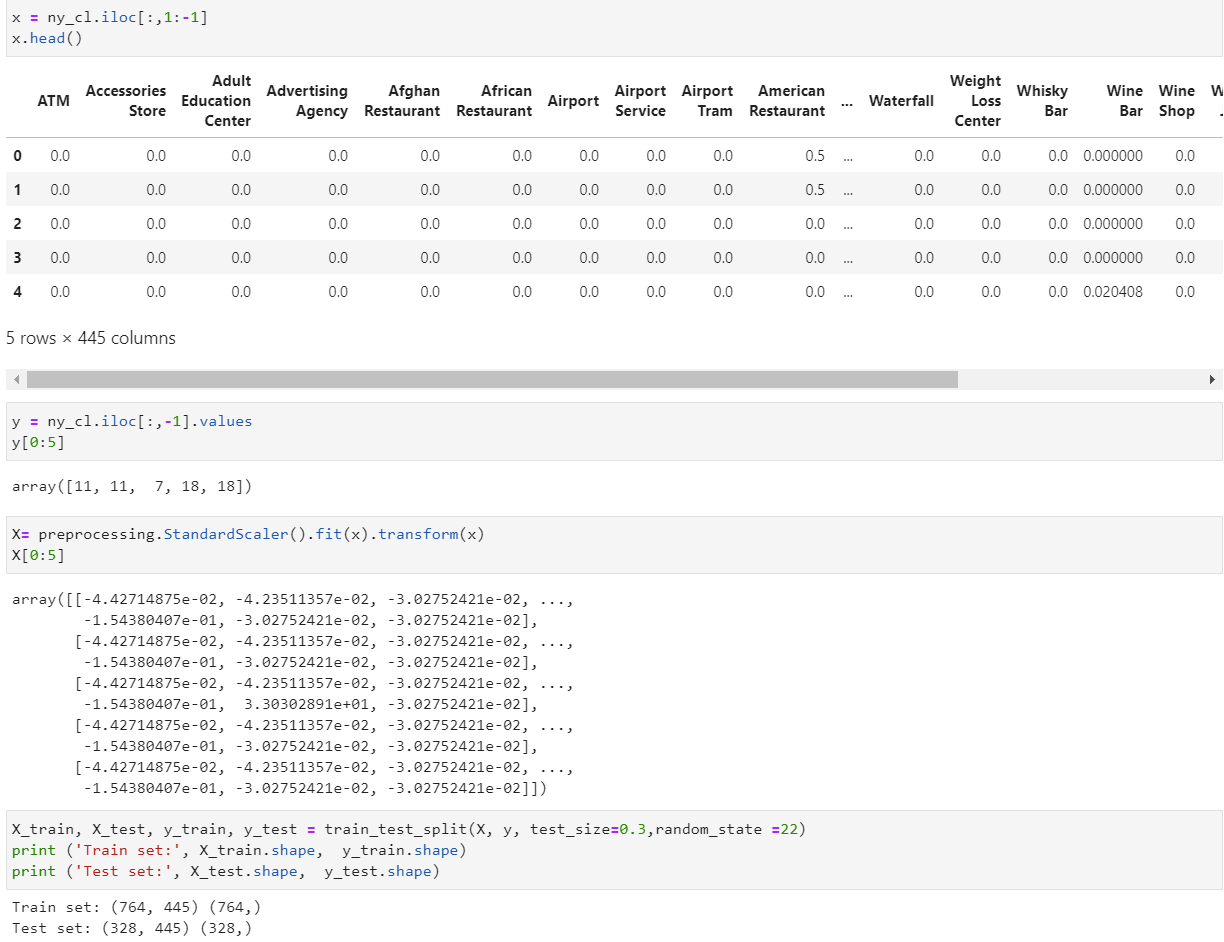


After looking at the map, four mayor colors have the most representation alongside the map, which are: Orange (Cluster No. 18), Sky Blue (Cluster No. 7), Purple (Cluster No. 9) and Light Green (Cluster No. 3). A better summary of the number of zip codes clusters in the area of New York is shown below:

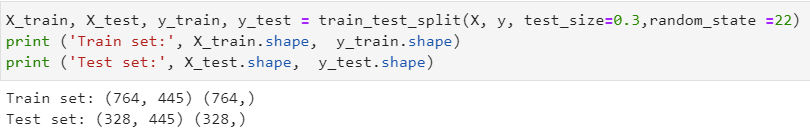
The state of New York has been divided into twenty-one clusters; this clusters will be used to categorize the zip codes for the other states. A k nearest neighbor model will be created to classify the rest of the zip codes for the remaining states, the results from the past model will serve this new model as target.

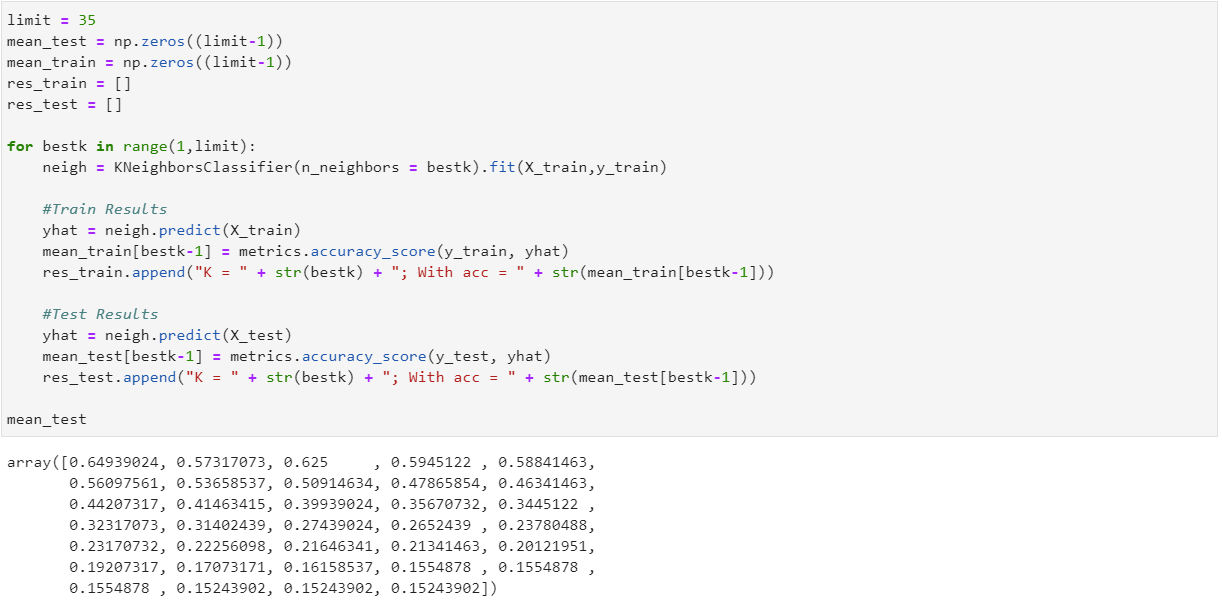
Before proceeding with the model, the data needs to be prepared so the model can run smoother and ignore other non-important pattern that the data contains. To proceed the data will be divided into x and y, being x all the categories of with the zip codes and y the cluster labels that came from the past model. The values from x will be transformed using a normal distribution, the result of this transformation will be called X.

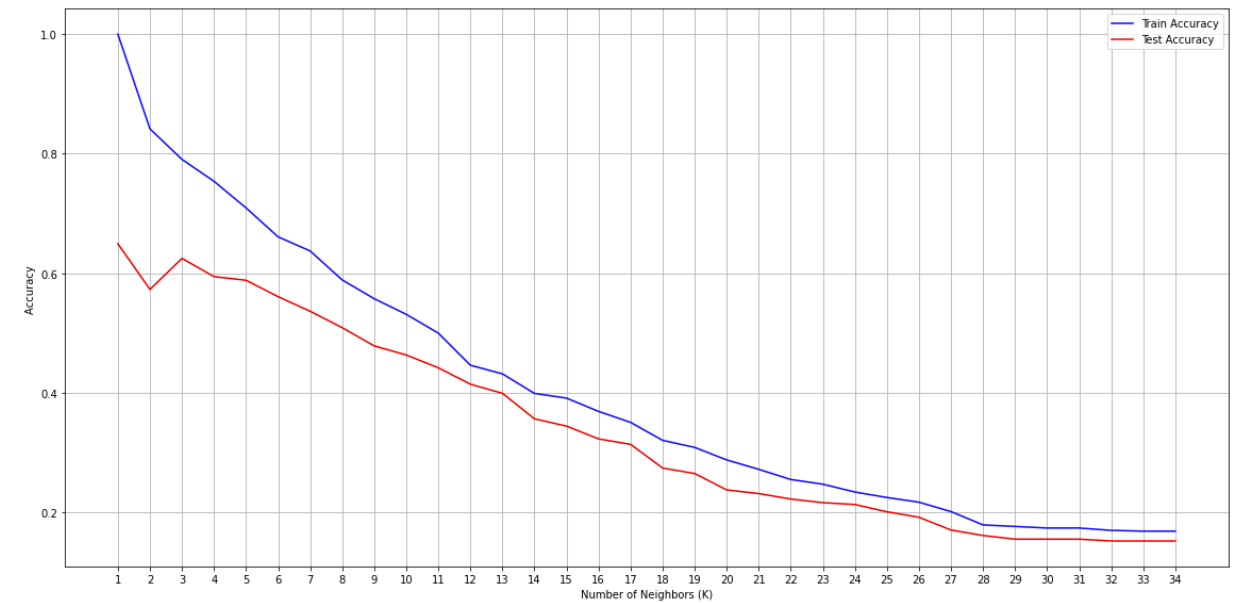
The variable X and y will be divided into X\_train, X\_test, y\_train, and y\_test using the function train\_test\_split. Thirty percent of the data will be used for testing the model and the rest for training it, it will also use a seed of twenty-two to select randomly the data used for test and train.



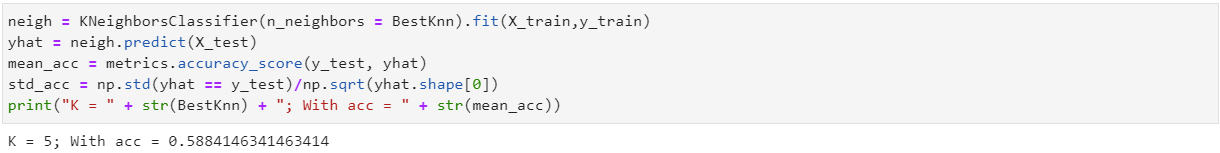
The best K needs to be determined so the KNN model can show the most accurate cluster category. To find the best K, the K that has the most accuracy for the test files will be chosen. Using a loop that will travel through the results and saving them into a list will make the process of plotting the curves of performances for each K using test and train files.





****

From the graph it is easy to tell that using more than thirty-five in the K will make the model a bad predictor; A much lower K must be used. K equals to five will be used because of the stability shown from passing from four to five, the detailed K provides a test accuracy of 0.59 approximately.



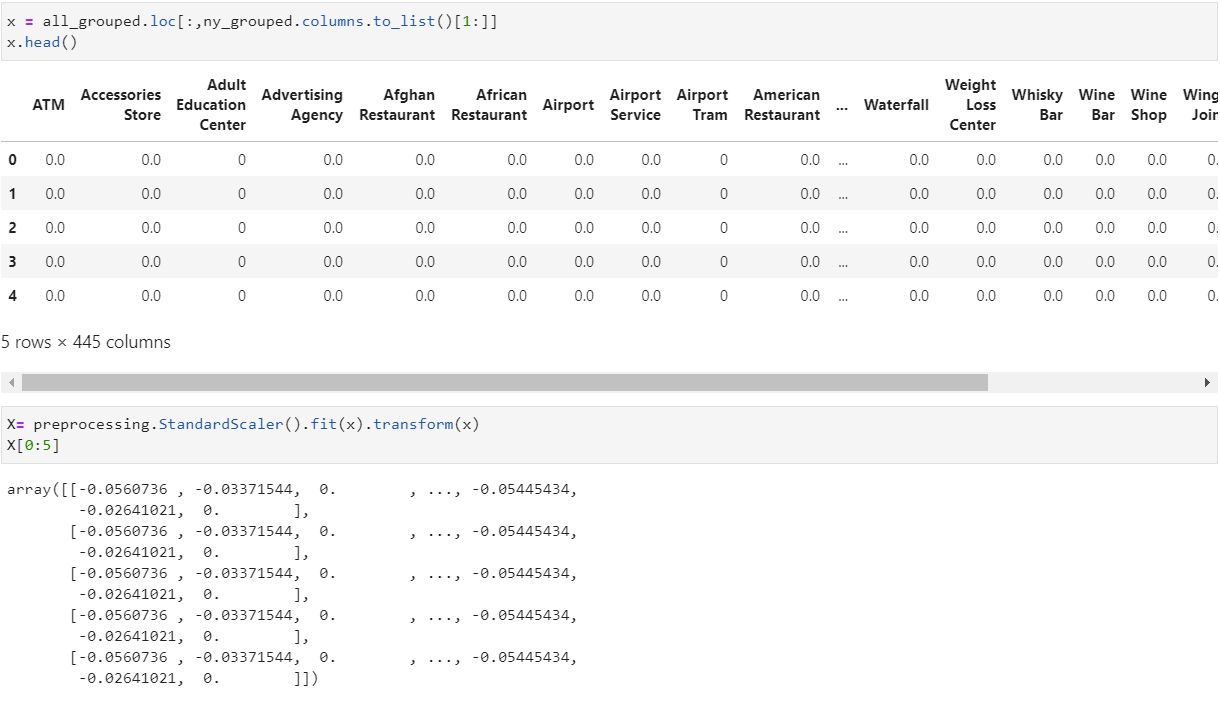
From this point, the process is ready to predict the values of the clusters of the categories of the available venues within the zip codes of CA, NV, FL, and WA. All their datasets need to be in the same format as NY’s was when the KNN model was trained, all the transformations that the state data sets need to go through are:

1. Using the function “getNearbyVenues” to get all the venues for each of the zip codes.
2. Use the one hot encoding technique to transform the categorical variable into several dummy variables so the model can predict.
3. Normalize the variable X for each state so it is represented in a normal distribution.

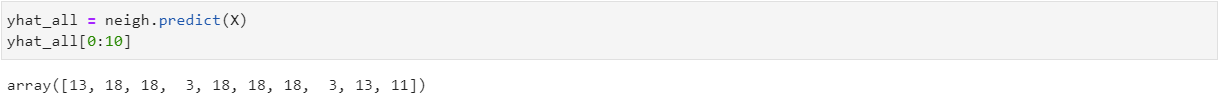
One problem emerged in this step that made the analysis biased in terms of the categories. The problem was that when the categories were extracted and transformed into dummy variables, those columns did not match the ones from NY.

To be able to proceed, the variables that were in NY but not in the rest of States had to be added with the value of 0, and the ones that appeared in the rest of the States but not in NY had to be deleted. All of this happed just because the model needs to have the same columns and format from the ones that the model received at the time of the training.

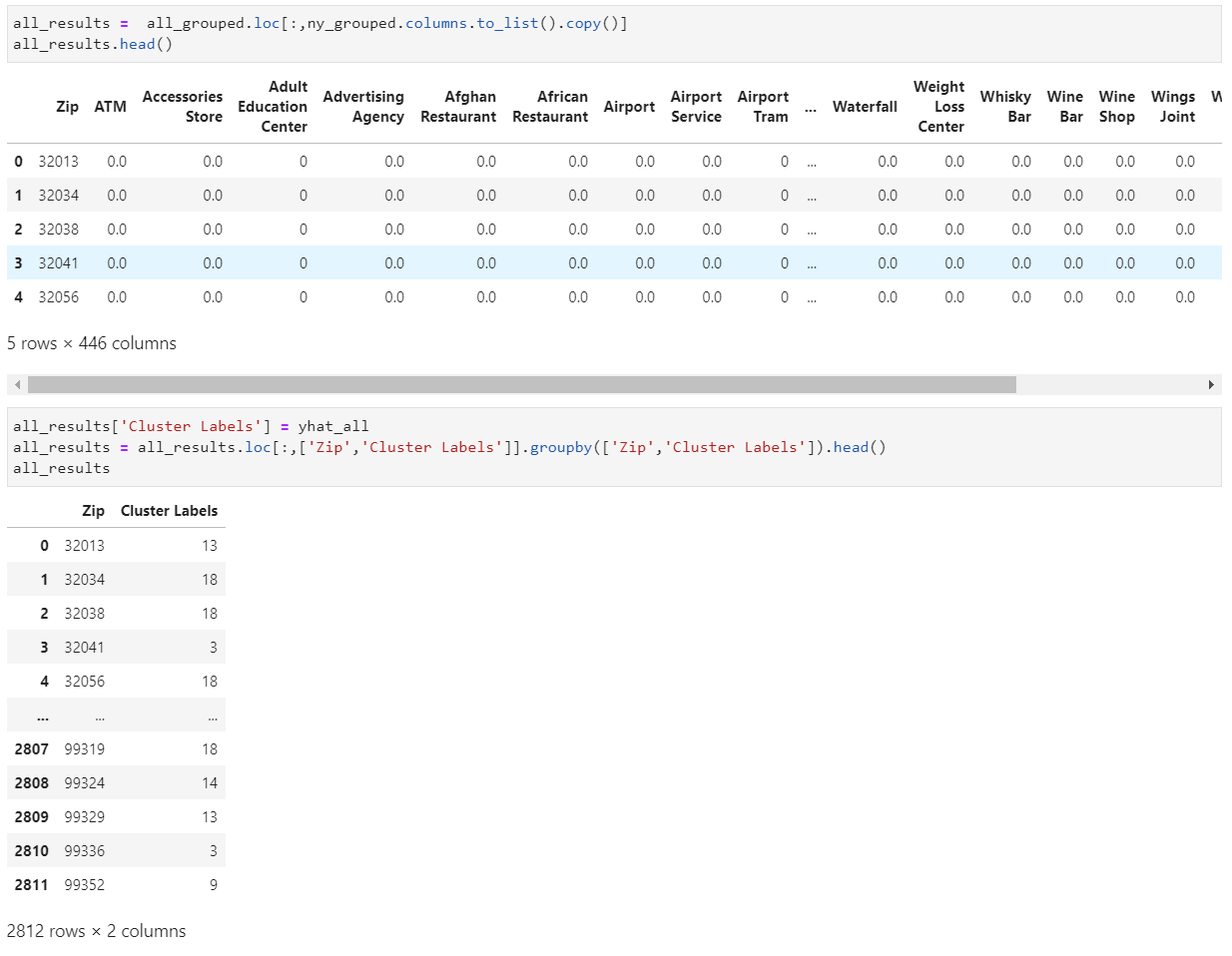
To reduce code, all the remaining states were combined into one single data set and had to go through the steps detailed above. The resulting data set had the following structure:



With the data ready and the best K determined, the only remaining step is to estimate the clusters labels for each zip code for the rest of the states.



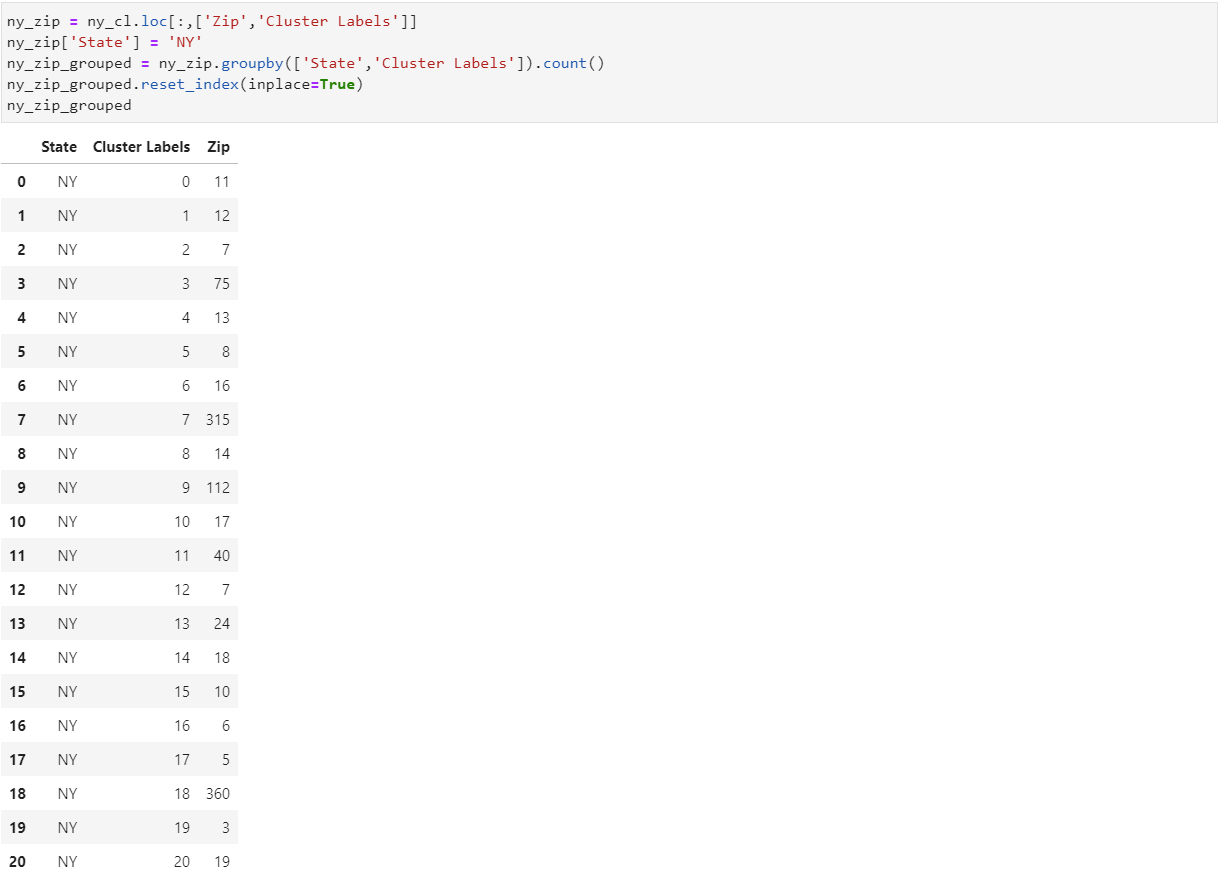
Now having the labels inside a list, it is needed to pair it with the main data set so the zip codes can be differentiated by the cluster labels. The labels will be joined with the table from where x was gotten, before normalizing the values and deleting the zip code for the training of the model.



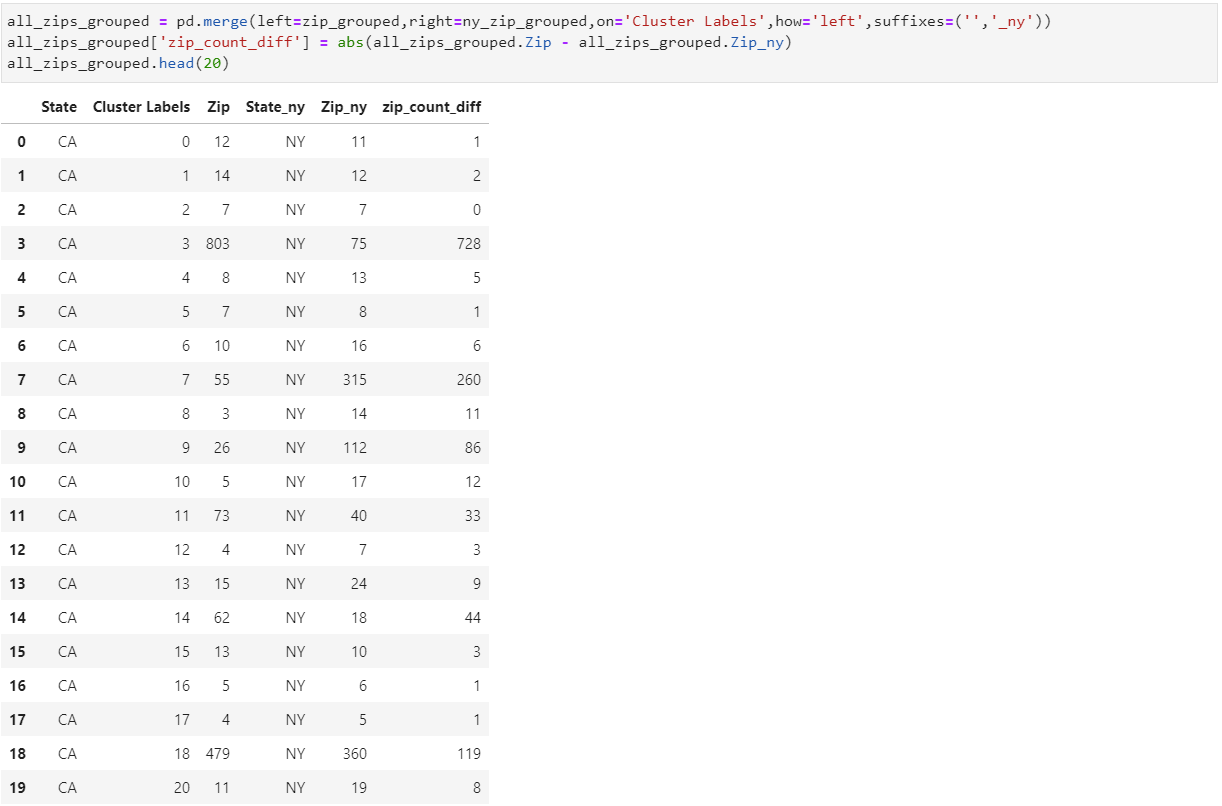
The main data set will be combined with the data set with the zip codes and cluster labels to get the column of the state.



A new data set will be generated using the cluster labels of the state of NY to count the amount of zip codes within each cluster per State, in this case only NY.



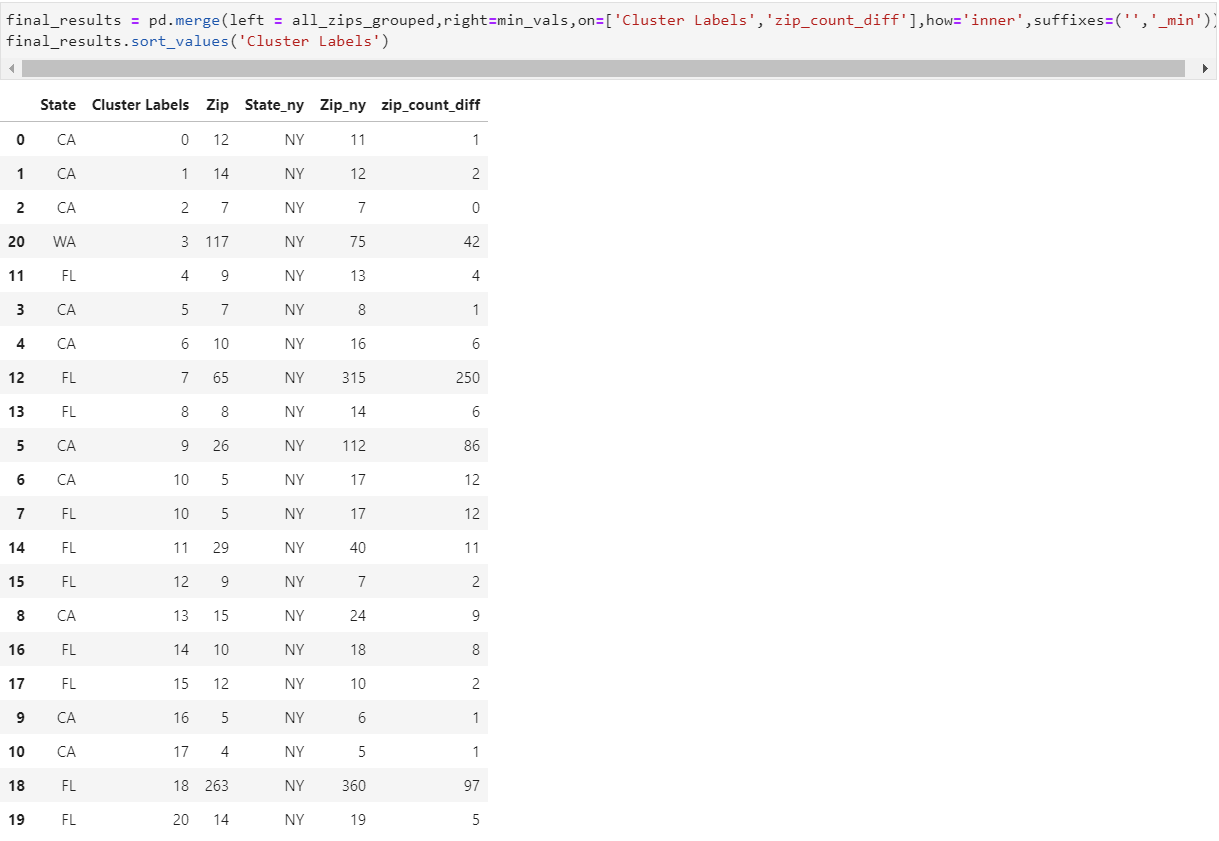
The last two data sets will be joined together using the key Cluster Labels so we can compare each cluster from NY with any other state in the list. A new column was added into this last data set which details the absolute differences between the amount of zip codes in NY and the other states.



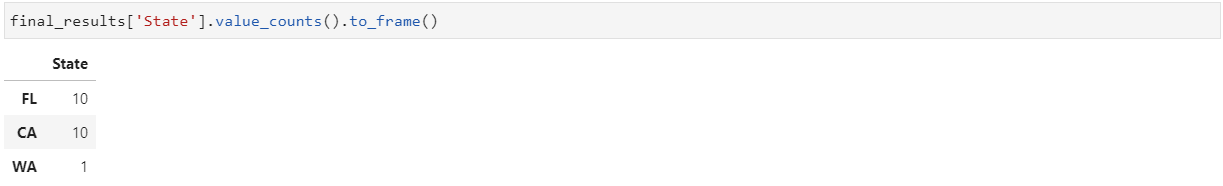
The data set of the comparisons will be grouped by the cluster labels to calculate the minimum difference for each cluster label of NY and other State.



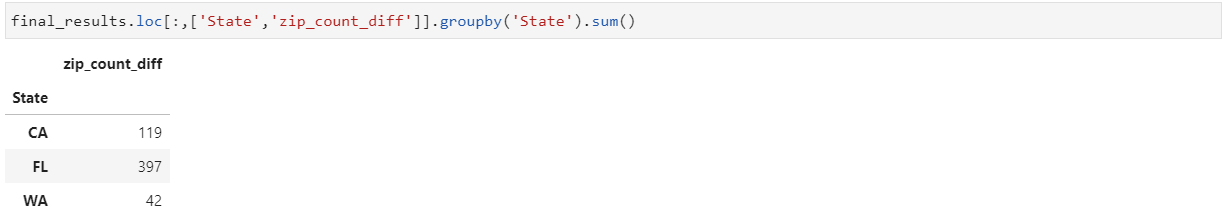
These minimum values will serve to determine which state has the least differences between NY in term of categories within all the venues. The minimum data set is going to be joined with the data set that just added the data from NY to the other states next to each other. The type of join will be inner because we intend to get the rows that match the cluster labels and zip\_count\_diff from the minimum dataset.



This final data set can be grouped by the state that managed to get the minimum differences alongside the categories of the available venues. The number of appearances for each state are detailed below:



Finally, the number of differences for each state that managed to have the minimum amount of differences:



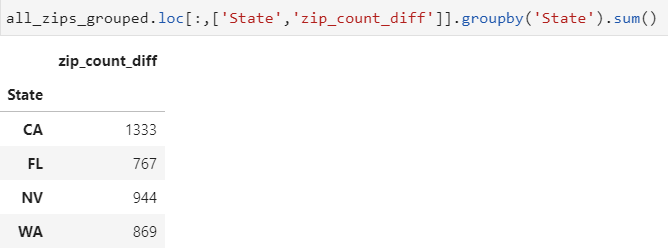
## Results

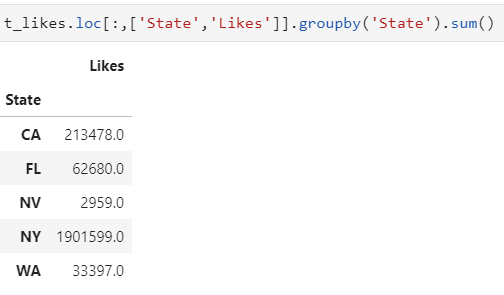
From the last two tables shown above it is important to recall that Nevada did not to have a minimum difference with a cluster of NY. With this it can said that NV is the least similar of the four possible states.

As shown in the table with the similarities within each state, it can be determined that there are two strong contenders in terms of similarity of clusters. Both California and Florida have ten clusters within their twenty-one clusters that are least different in comparison of the others. With this table it has been determined that the third most similar state to New York is Washington because it only managed to get one cluster that is very similar to one NY’s cluster.

Using the data from the last table it is easy to conclude that within the ten similar clusters from Florida contain a big number of differences in comparison of the ones of California. This analysis makes Florida the second most similar state to NY.

With the data shown below, it is easy to see that the total amount of differences across the clusters are attributed to California. Does this mean that it should be eliminated from the first place? No, because from the ten clusters, that are the most similar to ten clusters from NY, has the least differences. The quality goes over quantity. What should have Dave decide in this scenario?



When the number of likes per state is analyzed, a remarkable difference is shown:

## Discussion

This analysis can be categorized like a gross analysis because there were several variables that were not taken into consideration. One of them is the distribution of the likes and venues, in terms of how many clusters (geographically speaking) New York has and how well are the likes distributed between them. Also, the similarity of that index with the same index but for the other states.

The mayor problem of this tasks was to get all the data together because of the limitations of quota that Foursquare had, several times the data sets were deleted due to the lack of strategy when using an API. It is advised that the user saves all of the queries that he makes every time when working with a API, because of the kernel which will get reset at the time of a new session.

## Conclusion

Dave would like the opportunity to explore new stuff around his new state, but one of his conditions that he specified is that the state needed to be the most like New York. Since California had the least differences in the clusters that were the most similar and has the most amount of likes in comparison to the other states, the new “New York is California.