Implementation and Visualizing K-Means Clustering

by 蘇拉雅

Introduction

For this project we are going to use the results obtained from a previous project (Data weekly update with cron). This data is already in a Database and correspond to the USD-NTD exchange rate of March and April retrieved from a bank website.[1]

The goal of this project is to find groups in the data (clustering) implementing the K-Means Algorithm, and plot after achieve the result.

What do you need to know?

Programming Language:

- PHP and Javascript.

What do you need to make the implementation possible?

Tools:

- K-Means Implementation: PHP class jacobemerick/kmeans
- Visualizing K-Means Clustering: Javascript Highcharts charting library
- A Data resource (Database)

Before continue with the implementation, please take a look to the following text, this will help you to have an idea about what clustering is without knowledge of machine learning.

Introduction to K-means Clustering

K-means clustering is a type of unsupervised learning, which is used when you have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K. The algorithm works iteratively to assign each data point to one of K groups based on the features that are provided. Data points are clustered based on feature similarity. The results of the K-means clustering algorithm are:

1. The centroids of the K clusters, which can be used to label new data

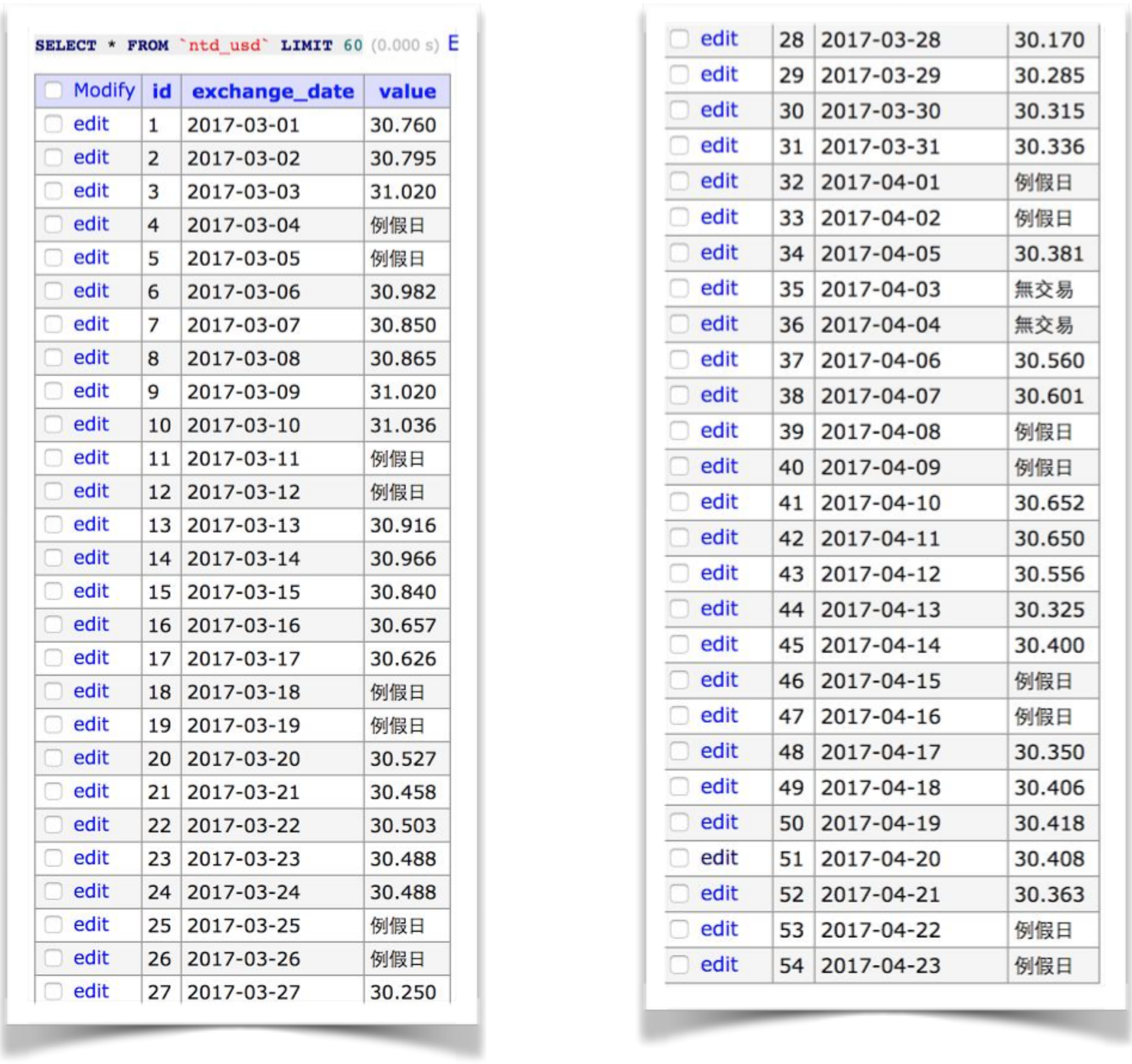
2. Labels for the training data (each data point is assigned to a single cluster)

Rather than defining groups before looking at the data, clustering allows you to find and analyze the groups that have formed organically. The "Choosing K" section below describes how the number of groups can be determined.

Each centroid of a cluster is a collection of feature values which define the resulting groups. Examining the centroid feature weights can be used to qualitatively interpret what kind of group each cluster represents. [2]

K-Means Implementation

1. The data source:



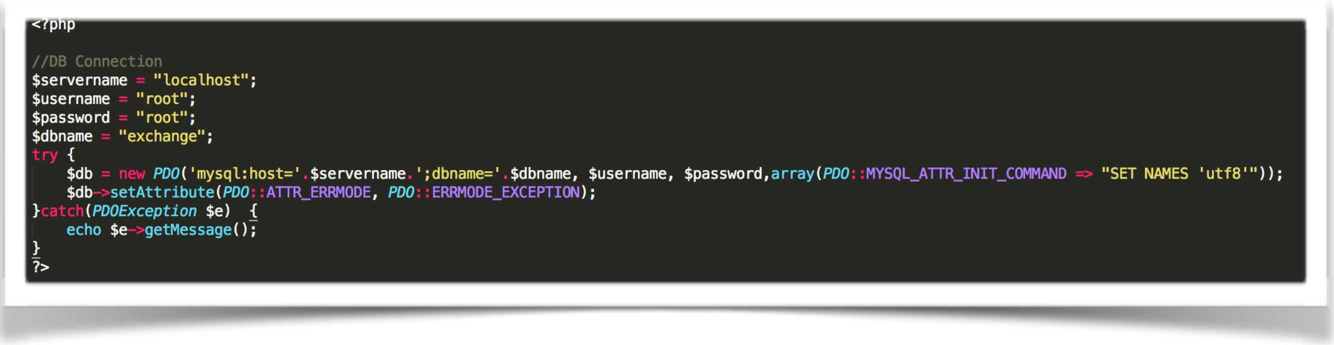
```
SELECT * FROM `ntd_usd` LIMIT 60 (0.000 s) E
```

□ Modify	id	exchange_date	value
□ edit	1	2017-03-01	30.760
□ edit	2	2017-03-02	30.795
□ edit	3	2017-03-03	31.020
□ edit	4	2017-03-04	例假日
□ edit	5	2017-03-05	例假日
□ edit	6	2017-03-06	30.982
□ edit	7	2017-03-07	30.850
□ edit	8	2017-03-08	30.865
□ edit	9	2017-03-09	31.020
□ edit	10	2017-03-10	31.036
□ edit	11	2017-03-11	例假日
□ edit	12	2017-03-12	例假日
□ edit	13	2017-03-13	30.916
□ edit	14	2017-03-14	30.966
□ edit	15	2017-03-15	30.840
□ edit	16	2017-03-16	30.657
□ edit	17	2017-03-17	30.626
□ edit	18	2017-03-18	例假日
□ edit	19	2017-03-19	例假日
□ edit	20	2017-03-20	30.527
□ edit	21	2017-03-21	30.458
□ edit	22	2017-03-22	30.503
□ edit	23	2017-03-23	30.488
□ edit	24	2017-03-24	30.488
□ edit	25	2017-03-25	例假日
□ edit	26	2017-03-26	例假日
□ edit	27	2017-03-27	30.250
□ edit	28	2017-03-28	30.170
□ edit	29	2017-03-29	30.285
□ edit	30	2017-03-30	30.315
□ edit	31	2017-03-31	30.336
□ edit	32	2017-04-01	例假日
□ edit	33	2017-04-02	例假日
□ edit	34	2017-04-05	30.381
□ edit	35	2017-04-03	無交易
□ edit	36	2017-04-04	無交易
□ edit	37	2017-04-06	30.560
□ edit	38	2017-04-07	30.601
□ edit	39	2017-04-08	例假日
□ edit	40	2017-04-09	例假日
□ edit	41	2017-04-10	30.652
□ edit	42	2017-04-11	30.650
□ edit	43	2017-04-12	30.556
□ edit	44	2017-04-13	30.325
□ edit	45	2017-04-14	30.400
□ edit	46	2017-04-15	例假日
□ edit	47	2017-04-16	例假日
□ edit	48	2017-04-17	30.350
□ edit	49	2017-04-18	30.406
□ edit	50	2017-04-19	30.418
□ edit	51	2017-04-20	30.408
□ edit	52	2017-04-21	30.363
□ edit	53	2017-04-22	例假日
□ edit	54	2017-04-23	例假日

Fig. 1: Table exchange_rate

2. Retrieve the data and create data set

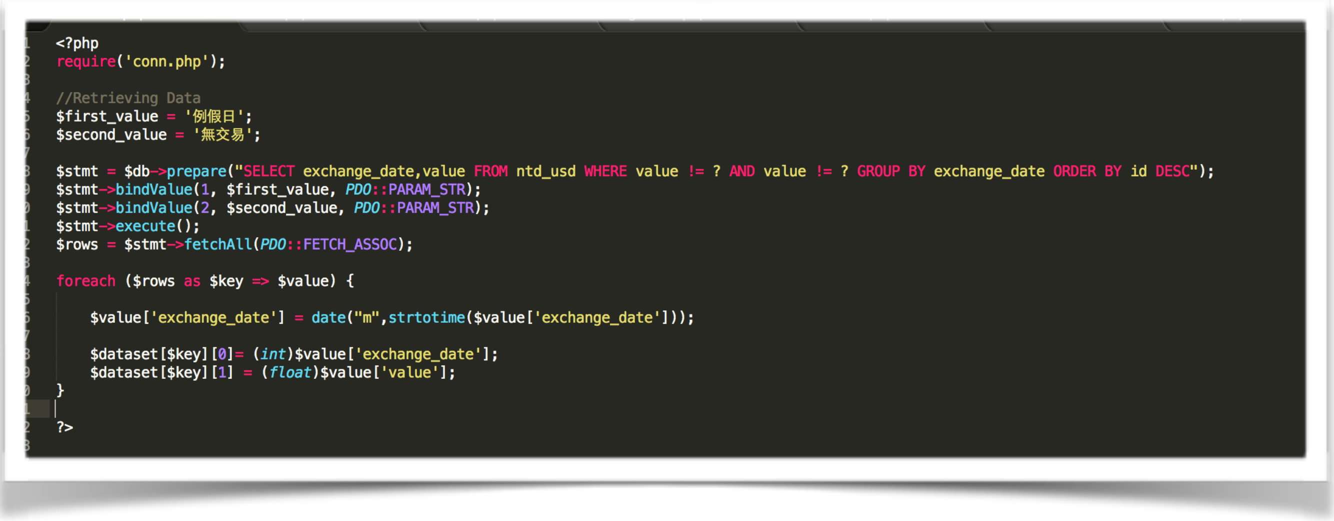
- Database connection



```php
<?php

//DB Connection
$servername = "localhost";
$username = "root";
$password = "root";
$dbname = "exchange";
try {
    $db = new PDO('mysql:host='.$servername.';dbname='.$dbname, $username, $password,array(PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES 'utf8'"));
    $db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
}catch(PDOException $e)  {
    echo $e->getMessage();
}
?>
```

Fig. 2: conn.php file

- Create Dataset

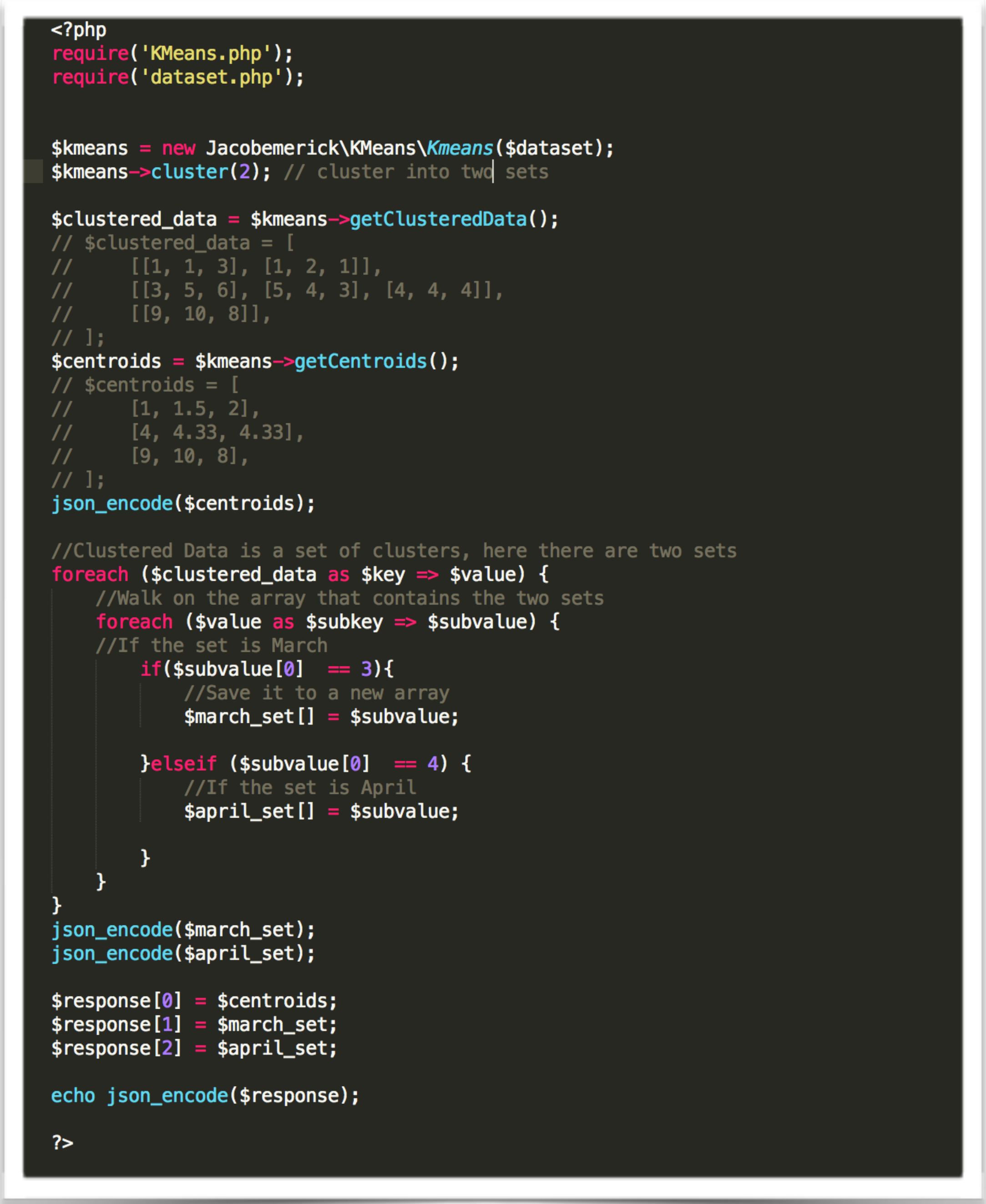


```php
<?php
require('conn.php');

//Retrieving Data
$first_value = '例假日';
$second_value = '無交易';

$stmt = $db->prepare("SELECT exchange_date,value FROM ntd_usd WHERE value != ? AND value != ? GROUP BY exchange_date ORDER BY id DESC");
$stmt->bindValue(1, $first_value, PDO::PARAM_STR);
$stmt->bindValue(2, $second_value, PDO::PARAM_STR);
$stmt->execute();
$rows = $stmt->fetchAll(PDO::FETCH_ASSOC);

foreach ($rows as $key => $value) {

    $value['exchange_date'] = date("m",strtotime($value['exchange_date']));

    $dataset[$key][0]= (int)$value['exchange_date'];
    $dataset[$key][1] = (float)$value['value'];
}

?>
```

Fig. 3: dataset.php file

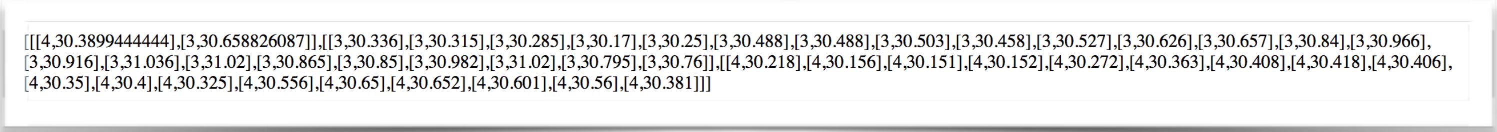
3. The Result

Because later will be required a JSON object so we store the results as a JSON object (centroids and clusters)



```php
<?php
require('KMeans.php');
require('dataset.php');


$kmeans = new Jacobemerick\KMeans\Kmeans($dataset);
$kmeans->cluster(2); // cluster into two sets

$clustered_data = $kmeans->getClusteredData();
// $clustered_data = [
//     [[1, 1, 3], [1, 2, 1]],
//     [[3, 5, 6], [5, 4, 3], [4, 4, 4]],
//     [[9, 10, 8]],
// ];
$centroids = $kmeans->getCentroids();
// $centroids = [
//     [1, 1.5, 2],
//     [4, 4.33, 4.33],
//     [9, 10, 8],
// ];
json_encode($centroids);

//Clustered Data is a set of clusters, here there are two sets
foreach ($clustered_data as $key => $value) {
    //Walk on the array that contains the two sets
    foreach ($value as $subkey => $subvalue) {
    //If the set is March
        if($subvalue[0]  == 3){
            //Save it to a new array
            $march_set[] = $subvalue;

        }elseif ($subvalue[0]  == 4) {
            //If the set is April
            $april_set[] = $subvalue;

        }
    }
}
json_encode($march_set);
json_encode($april_set);

$response[0] = $centroids;
$response[1] = $march_set;
$response[2] = $april_set;

echo json_encode($response);

?>
```

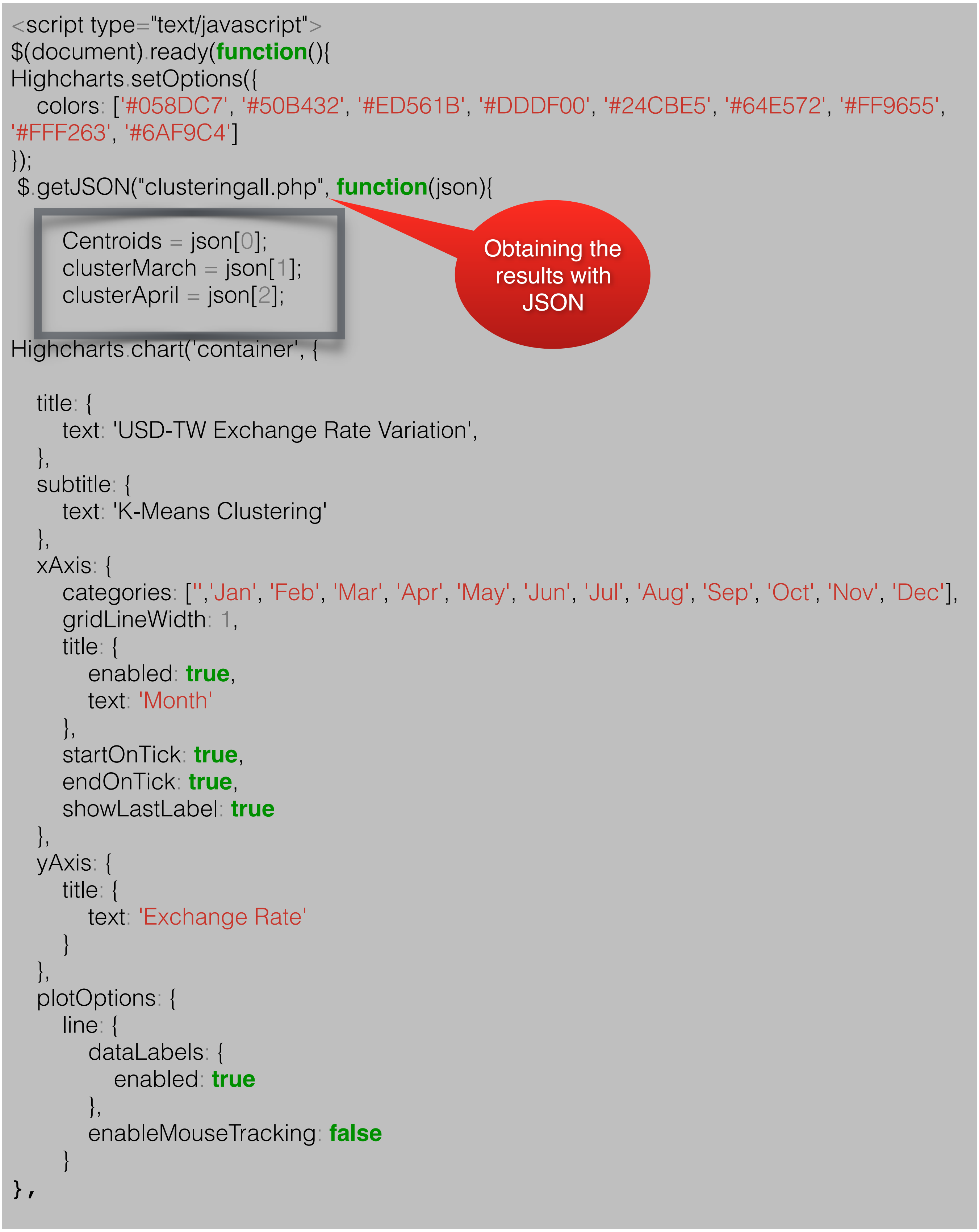
Fig. 4: clusteringall.php [3]



```
[[[4,30.3899444444],[3,30.658826087]],[[3,30.336],[3,30.315],[3,30.285],[3,30.17],[3,30.25],[3,30.488],[3,30.488],[3,30.503],[3,30.458],[3,30.527],[3,30.626],[3,30.657],[3,30.84],[3,30.966],
[3,30.916],[3,31.036],[3,31.02],[3,30.865],[3,30.85],[3,30.982],[3,31.02],[3,30.795],[3,30.76]],[[4,30.218],[4,30.156],[4,30.151],[4,30.152],[4,30.272],[4,30.363],[4,30.408],[4,30.418],[4,30.406],
[4,30.35],[4,30.4],[4,30.325],[4,30.556],[4,30.65],[4,30.652],[4,30.601],[4,30.56],[4,30.381]]]
```

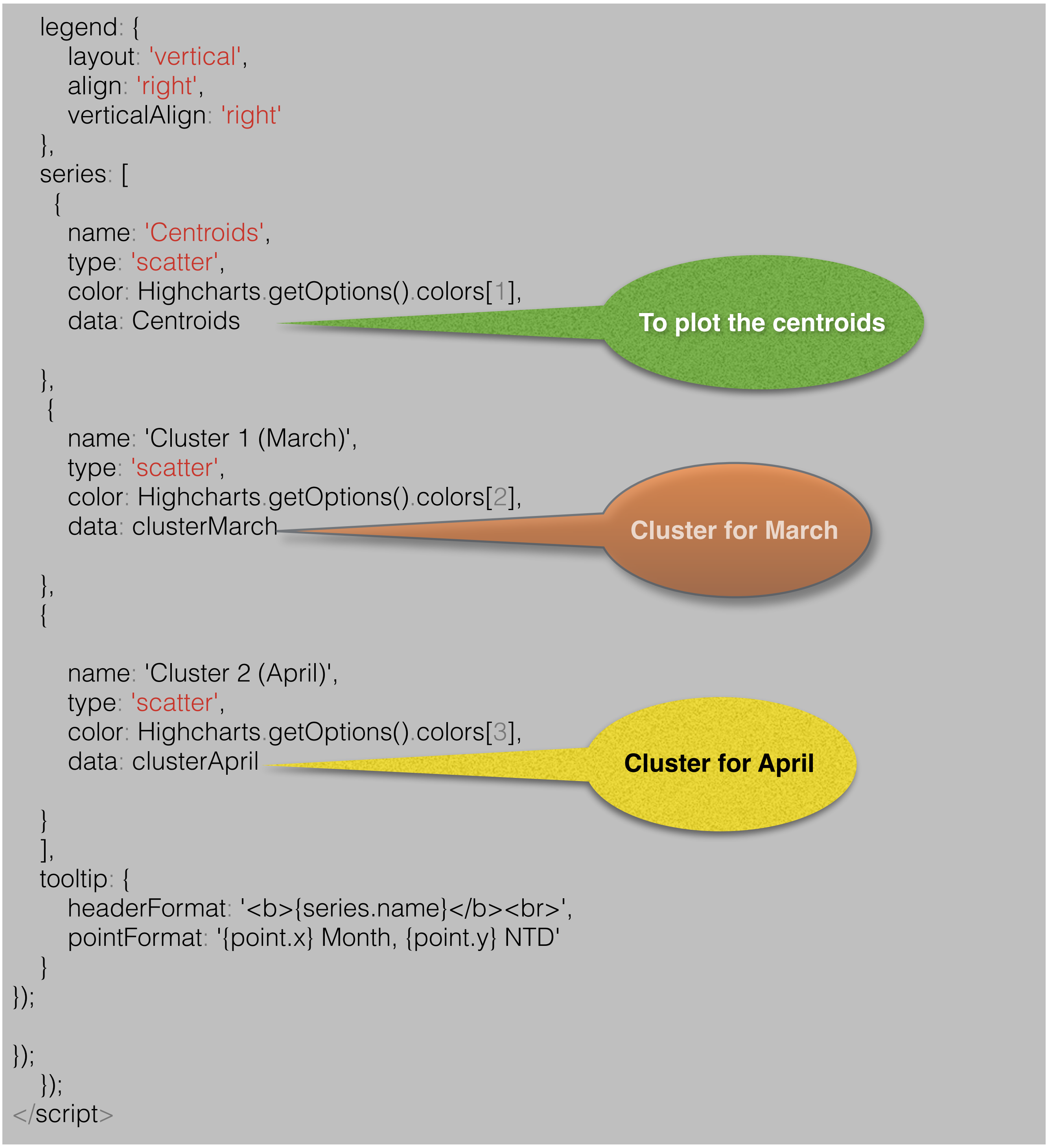
Fig. 5: The JSON Object for the clusters and centroids

1. Highchart Implementation Script: The results (centroids and clusters) are stored in different variables to add it to the highchart configuration.



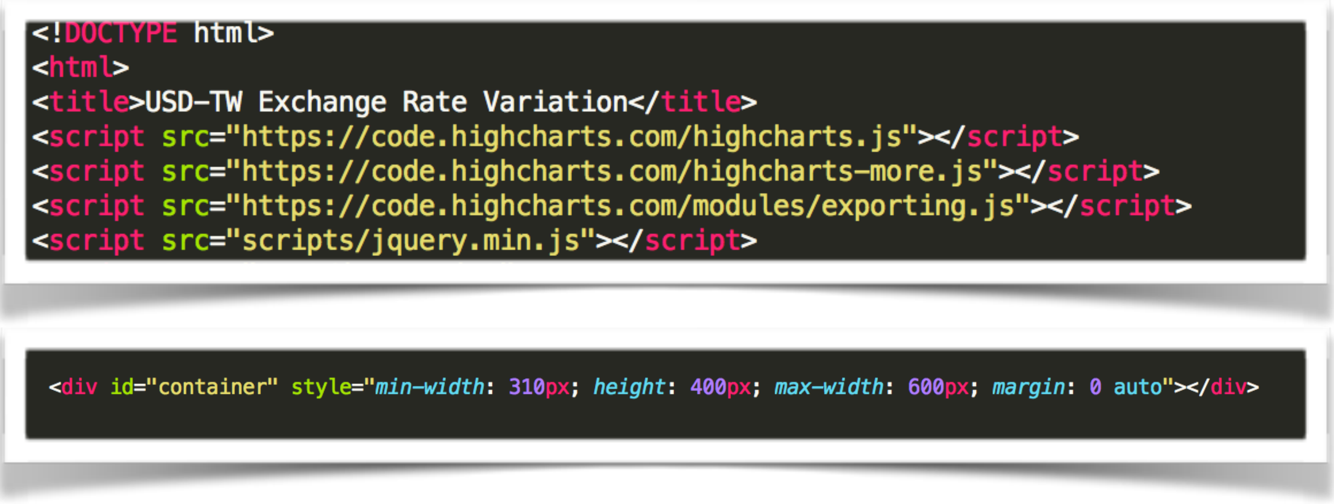
```
<script type="text/javascript">
$(document).ready(function(){
Highcharts.setOptions({
   colors: ['#058DC7', '#50B432', '#ED561B', '#DDDF00', '#24CBE5', '#64E572', '#FF9655',
'#FFF263', '#6AF9C4']
});
 $.getJSON("clusteringall.php", function(json){

      Centroids = json[0];
      clusterMarch = json[1];
      clusterApril = json[2];

Highcharts.chart('container', {

   title: {
      text: 'USD-TW Exchange Rate Variation',
   },
   subtitle: {
      text: 'K-Means Clustering'
   },
   xAxis: {
      categories: ['','Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'],
      gridLineWidth: 1,
      title: {
         enabled: true,
         text: 'Month'
      },
      startOnTick: true,
      endOnTick: true,
      showLastLabel: true
   },
   yAxis: {
      title: {
         text: 'Exchange Rate'
      }
   },
   plotOptions: {
      line: {
         dataLabels: {
            enabled: true
         },
         enableMouseTracking: false
      }
},
```

Obtaining the results with JSON



```
    legend: {
        layout: 'vertical',
        align: 'right',
        verticalAlign: 'right'
    },
    series: [
      {
        name: 'Centroids',
        type: 'scatter',
        color: Highcharts.getOptions().colors[1],
        data: Centroids

    },
     {
        name: 'Cluster 1 (March)',
        type: 'scatter',
        color: Highcharts.getOptions().colors[2],
        data: clusterMarch

    },
    {

        name: 'Cluster 2 (April)',
        type: 'scatter',
        color: Highcharts.getOptions().colors[3],
        data: clusterApril

    }
    ],
    tooltip: {
        headerFormat: '<b>{series.name}</b><br>',
        pointFormat: '{point.x} Month, {point.y} NTD'
    }
});

});
    });
</script>
```

Fig. 6: Highchart Script [4]



```
<!DOCTYPE html>
<html>
<title>USD-TW Exchange Rate Variation</title>
<script src="https://code.highcharts.com/highcharts.js"></script>
<script src="https://code.highcharts.com/highcharts-more.js"></script>
<script src="https://code.highcharts.com/modules/exporting.js"></script>
<script src="scripts/jquery.min.js"></script>
```

```
<div id="container" style="min-width: 310px; height: 400px; max-width: 600px; margin: 0 auto"></div>
```

Fig. 7: Highchart HTML code

2. The result:

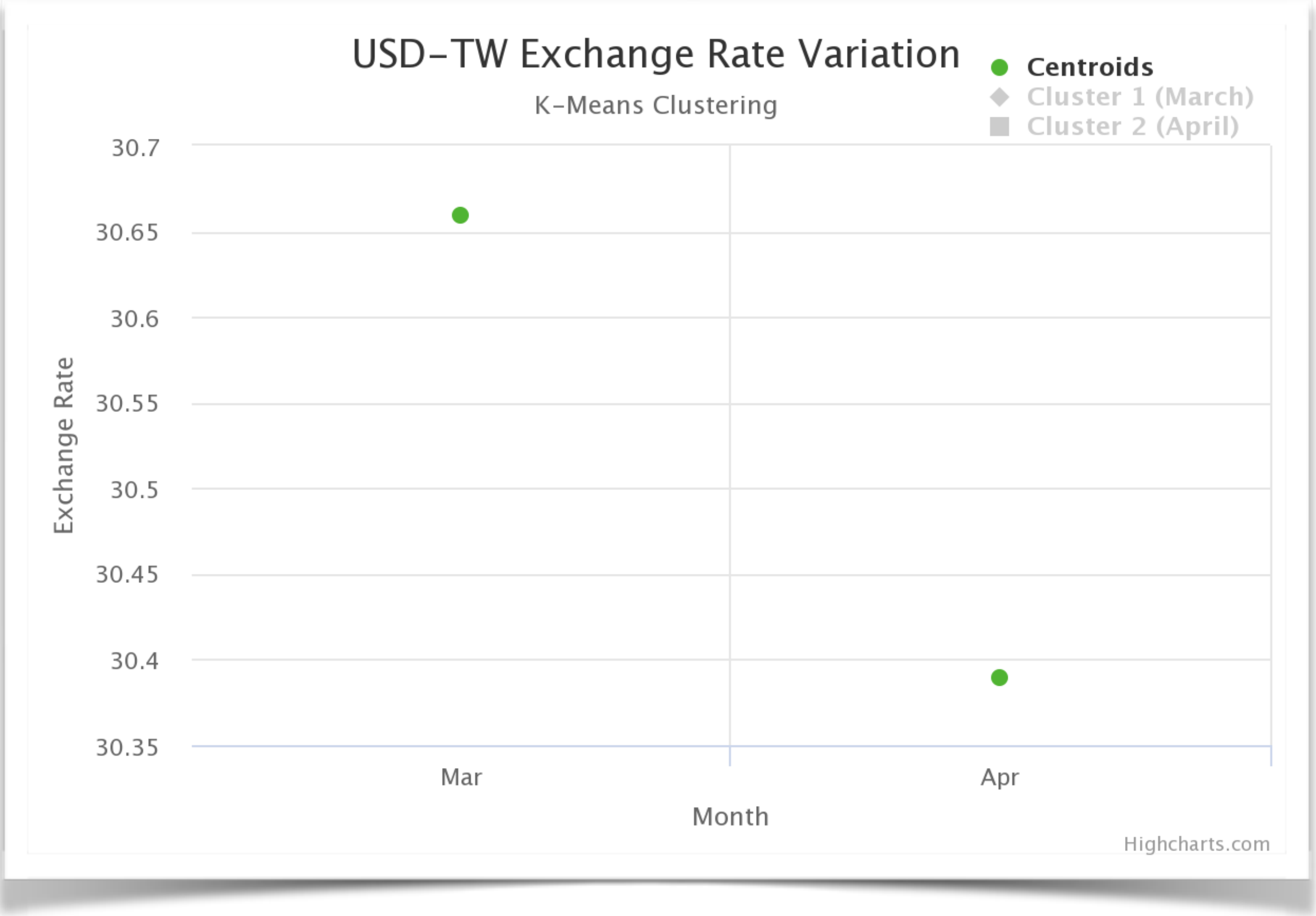


Fig. 8: The centroids

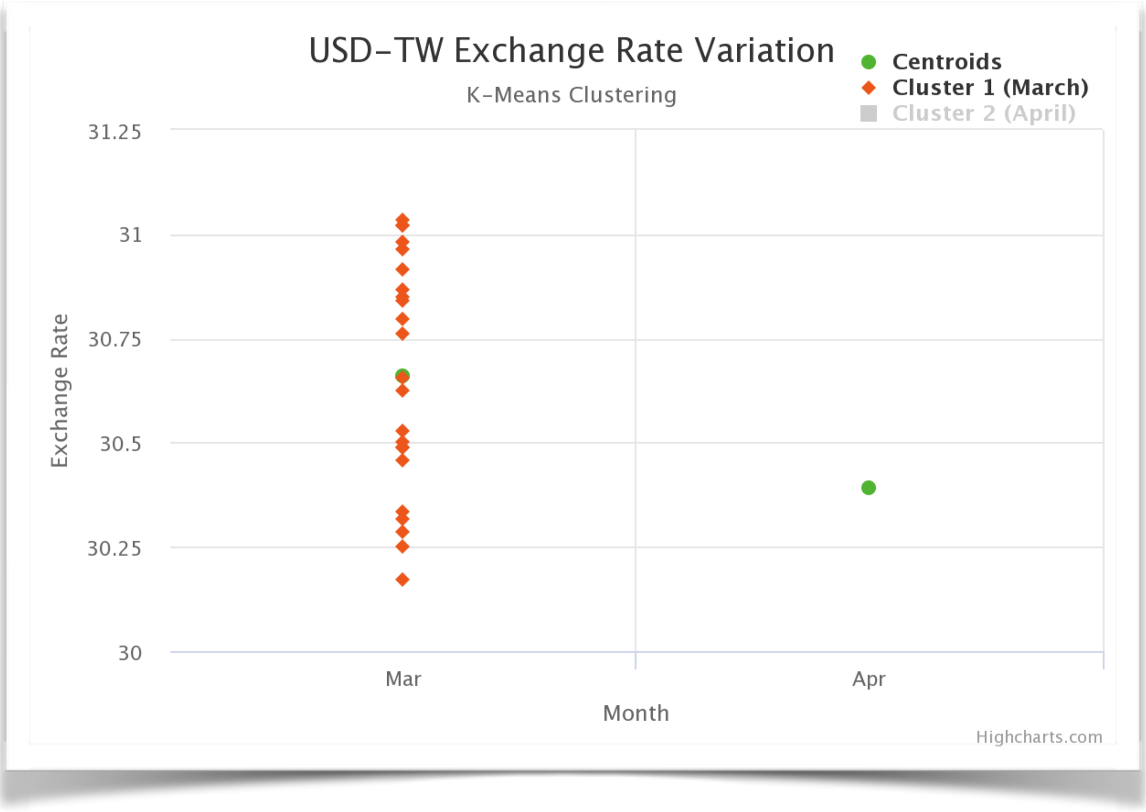


Fig9: Cluster 1 (March)

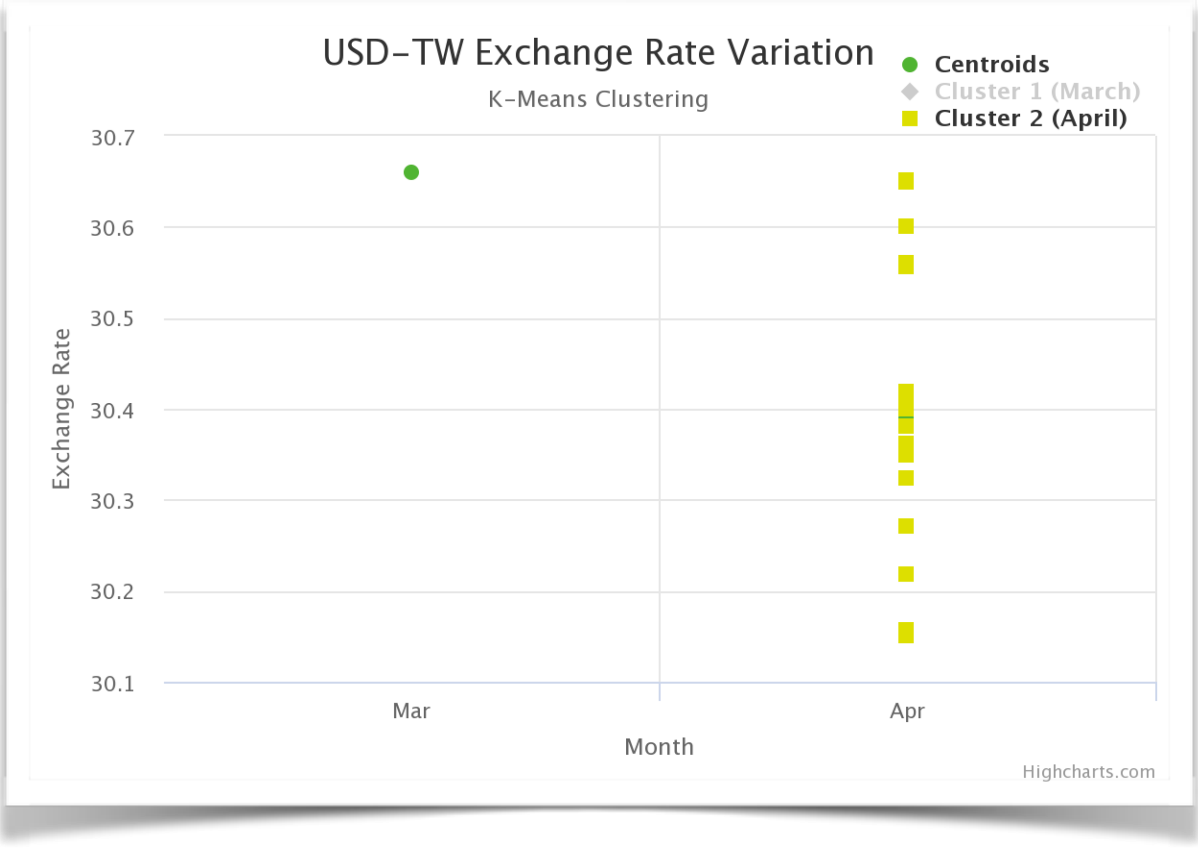
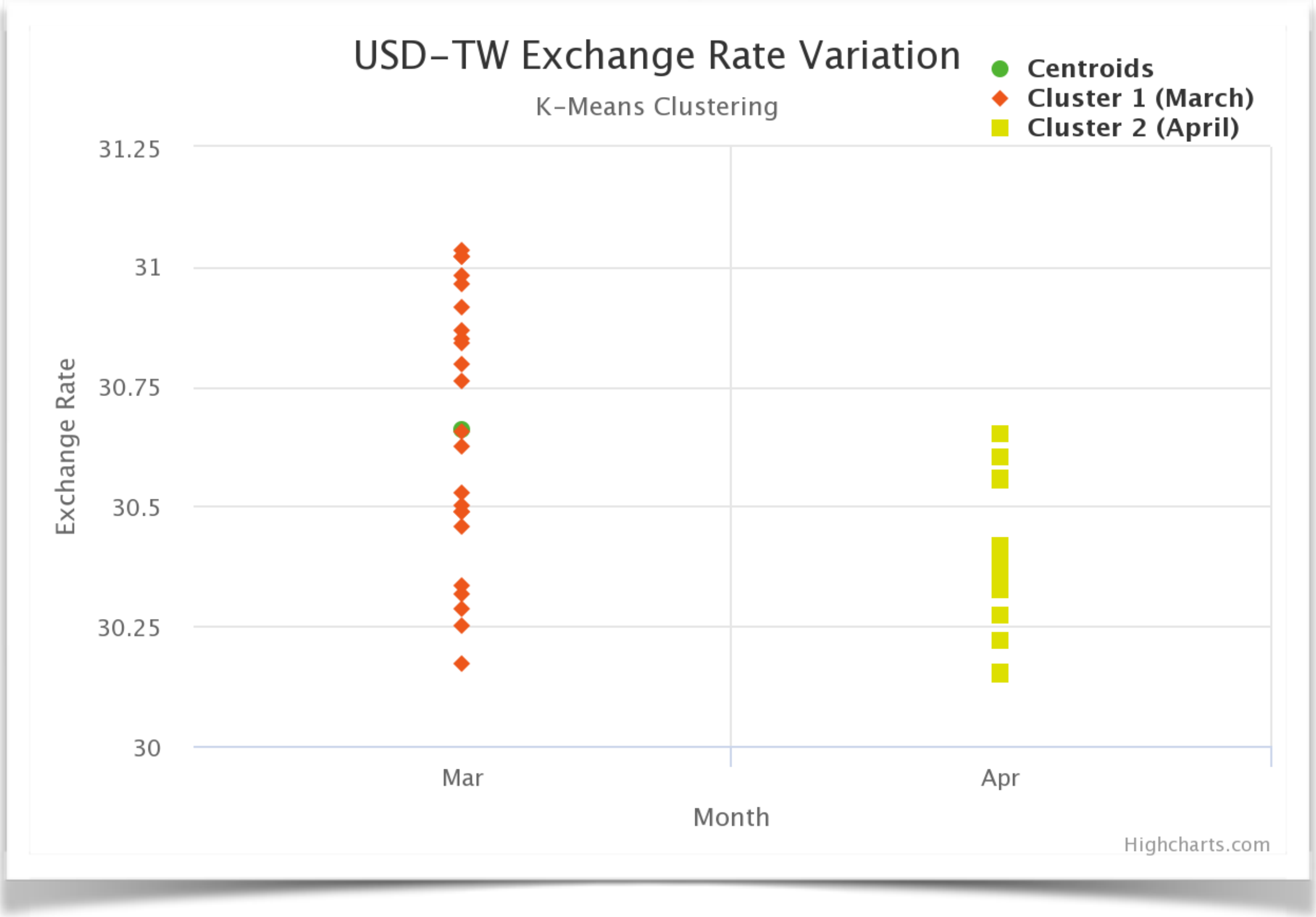
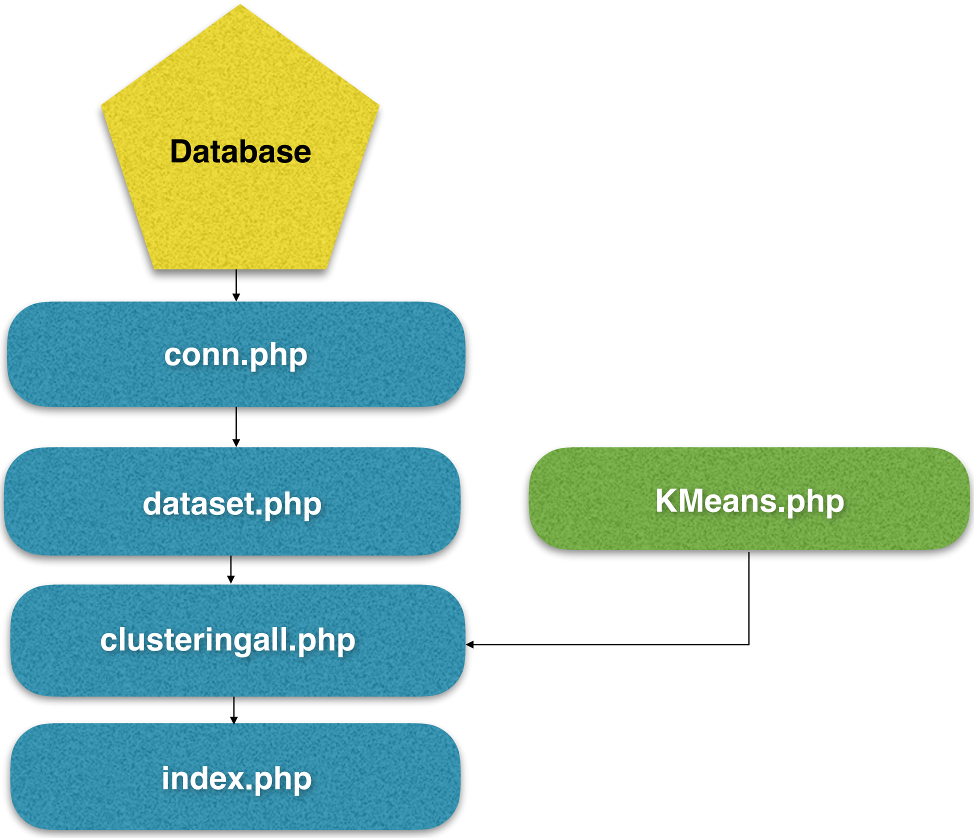


Fig. 10: Cluster 2 (April)

Fig. 11: K-Means Clustering Complete Result



For further references of implementation please review the the attached source code:



References

[1]"歡迎來到中央銀行全球資訊網 — 新臺幣/美元 銀行間收盤匯率", Cbc.gov.tw, 2017. [Online]. Available: http://www.cbc.gov.tw/lp.asp?CtNode=645&CtUnit=308&BaseDSD=32&mp=1&nowPag e=1&pagesize=50. [Accessed: 24- May- 2017].

[2]A. Trevino, "Introduction to K-means Clustering", Datascience.com, 2017. [Online]. Available: https://www.datascience.com/blog/introduction-to-k-means-clustering-algorithm-learn-data-science-tutorials. [Accessed: 24- May- 2017].

[3]"jacobemerick/kmeans", GitHub, 2017. [Online]. Available: https://github.com/jacobemerick/kmeans. [Accessed: 24- May- 2017].

[4]"Highcharts API Reference", Api.highcharts.com, 2017. [Online]. Available: http://api.highcharts.com/highcharts/. [Accessed: 24- May- 2017].