1. Data Processing
   1. Data introduction

Twitter data --Adebayo

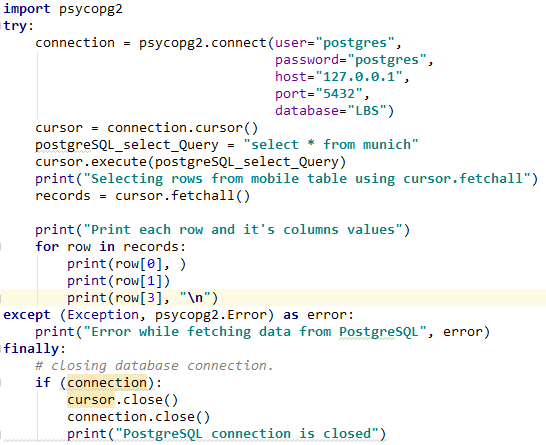
Statistical districts in three cities -- Olga

* 1. **Database – Lin**
  2. **Valence value calculation** **– Lin**

Sentiment analysis is the computational process of calculating whether the texts are positive, neutral or negative. Twitter is an online social media platform, users can post and share their state or feeling in the short text which named as “tweet” with others. Tweets are very popular and good candidate in the sentiment analysis. In this project, we also use tweets to do the sentiment analysis in the three different cities Vienna, Munich and Brussels. Calculating the valence value which present the character of tweets is one of the most important process. In this part, the method of calculation will be introduced. They are processed in Python. There are three main procedures included: connecting the Database, Cleaning the texts and Analyzing the sentiment.

1. **Connecting the Database**

The raw twitter data is imported and stored in PostgreDB. Python provide a library “psycopg2” which can be used to access the data in PostgreDB by using the SQL query. The code example shows as Figure xx.

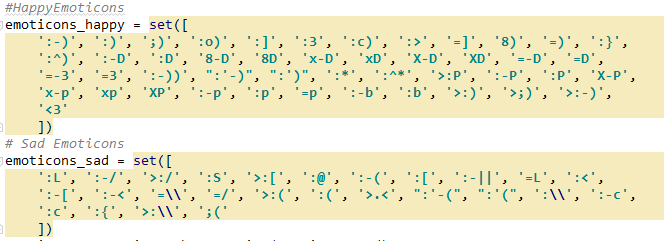


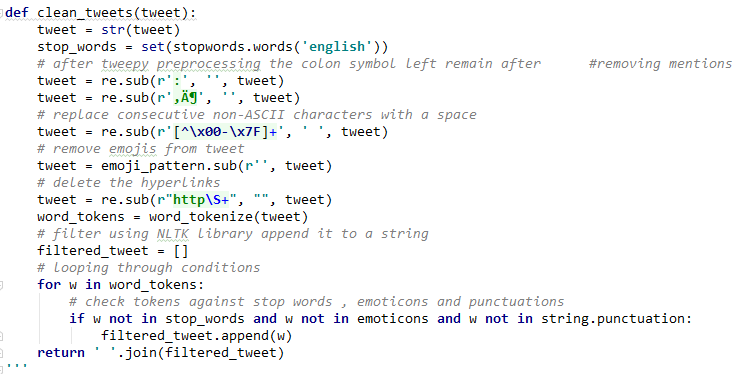
1. **Cleaning the texts**

Tweets data extracted from the twitter cannot be used directly. They can always contain emojis, characters, hyperlinks and meaningless stop words etc. like Table xx shows.

|  |
| --- |
| Voor ons vliegen ze dan altijd 😁 #venice #venezia here we come #aviation #avgeek #citytrip… https://t.co/j2k514ZvWV |

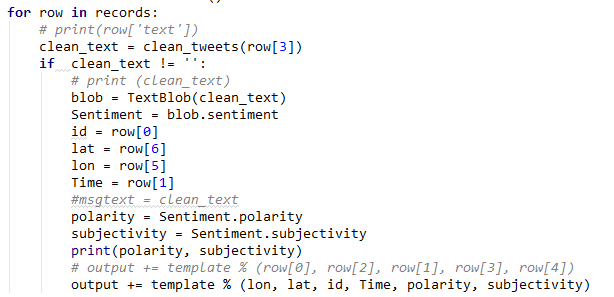
Targeted to these data in the texts, some are deleted and some are kept. As some emojis do have very distinctive emotion type, we create a emoji patterns which can used to present happy and sad emotion clearly, as Figure xx shows. For emojis existing in the emoji pattern will be kept for the further calculating. Specifically, the deleted data are: 1) English stop words 2) Consecutive non-ASCII characters 3) hyperlinks 4) emojis not in the emoji pattern. Code for the text cleaning shows as Figure xx. For selecting the stop words from the English texts, we used python library “nltk” .





1. **Analyzing the sentiment**

Valence value calculation depends on python library “TextBlob” It provides a simple API for doing the sentiment analysis task. The result from the sentiment analyzation can be divided into two parts: polarity and subjectivity. Both of them composed the valence degree of the texts. The code shows as Figure xx.



* 1. Data Formatting --Adeayo

Selecting time stamp

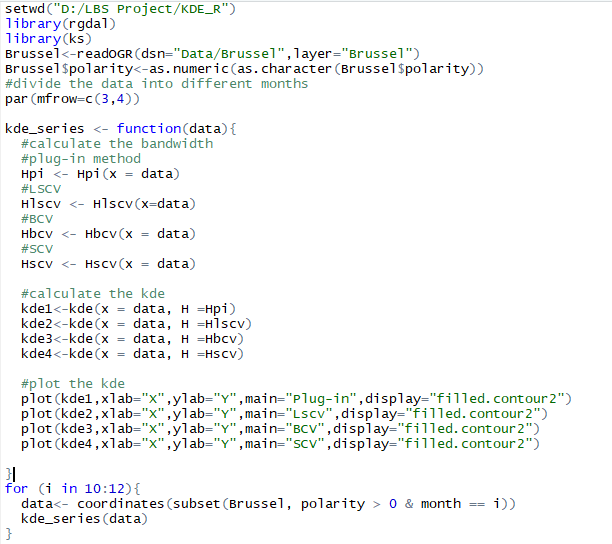
1. Heat Map
   1. **KDE Analyzation – Lin**

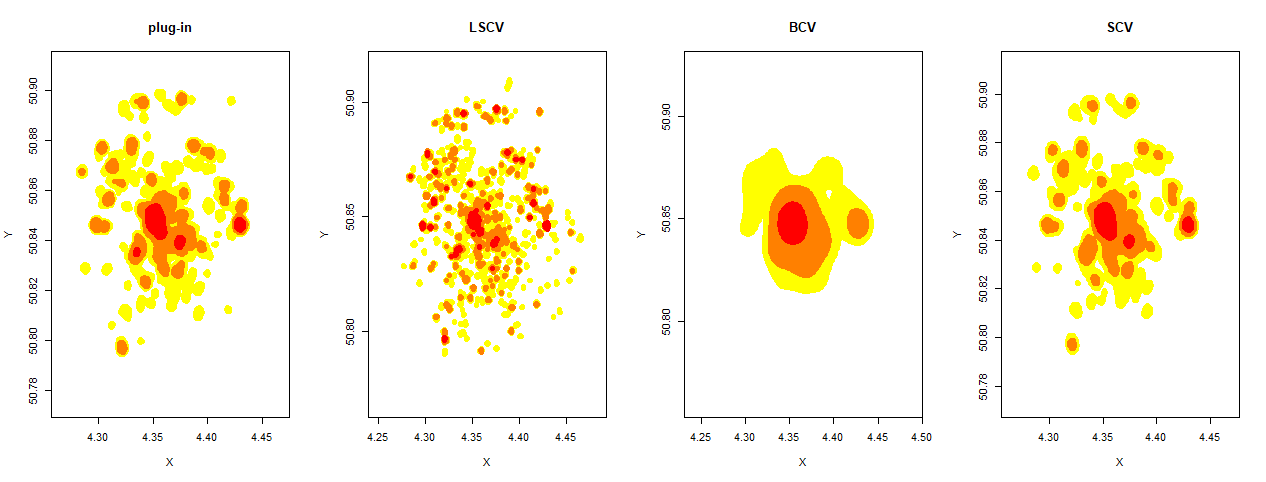
The first analyzation approach is using the KDE method to calculate and map the density of positive and negative tweets separately. In this project, we used the ks R library to calculate the KDE.

**The Principle of KDE**

**Band Width Selection**

The code for the bandwidth calculation shows as Figure xx. The January Brussels positive tweets data were used as the test data for doing the comparison among the results from the different bandwidth selectin algorithm. The result shows as **Figure xxx**. In the map the red part present the density is above 25%, orange and red combined part presents the density is above 50% and the yellow combined with orange and red presents the density is above 75%. Check the appendix to get the complete results for three cities in the 12 months.

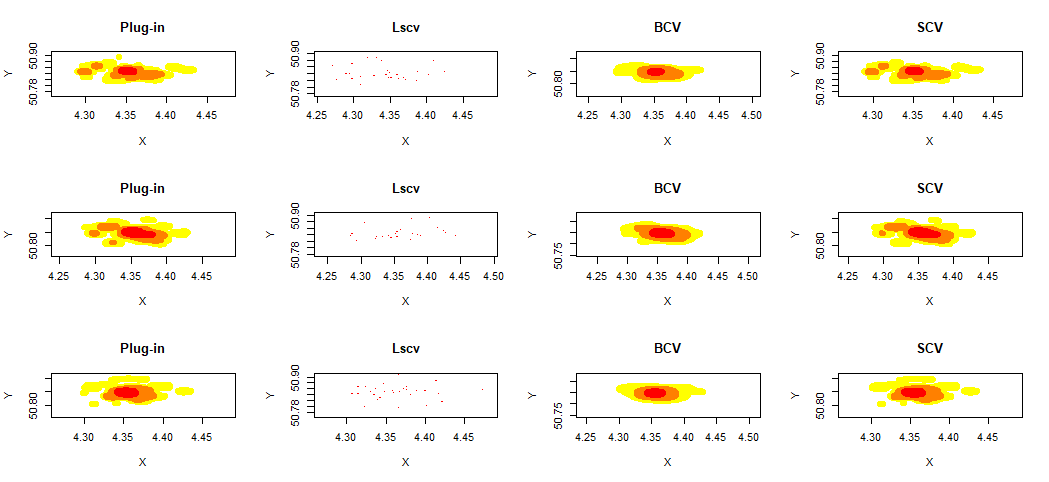
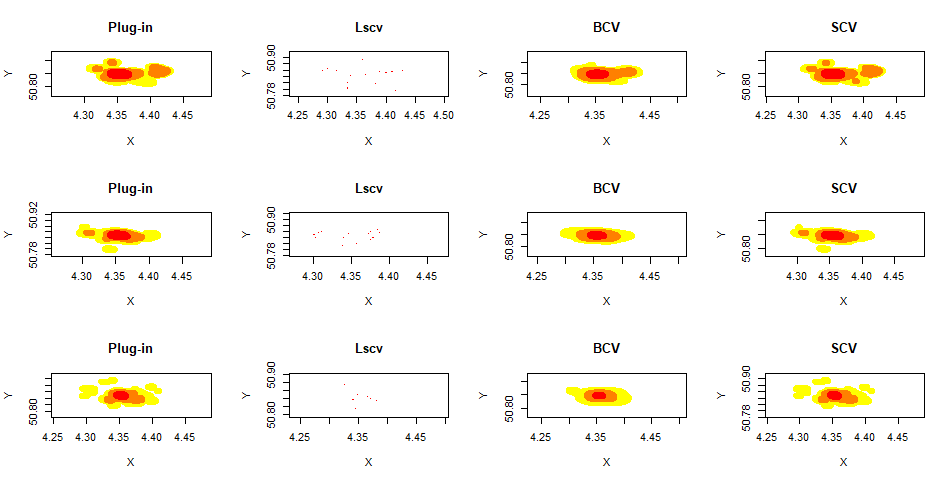
****

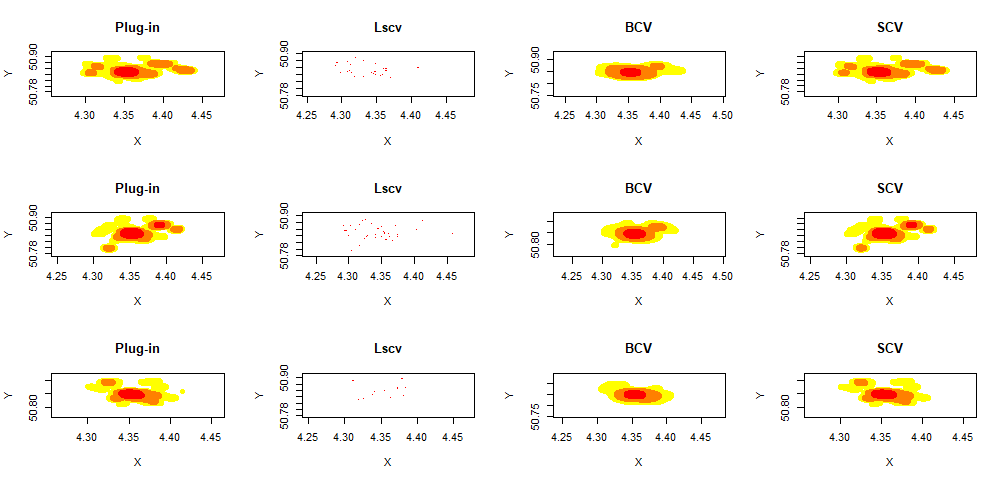
****

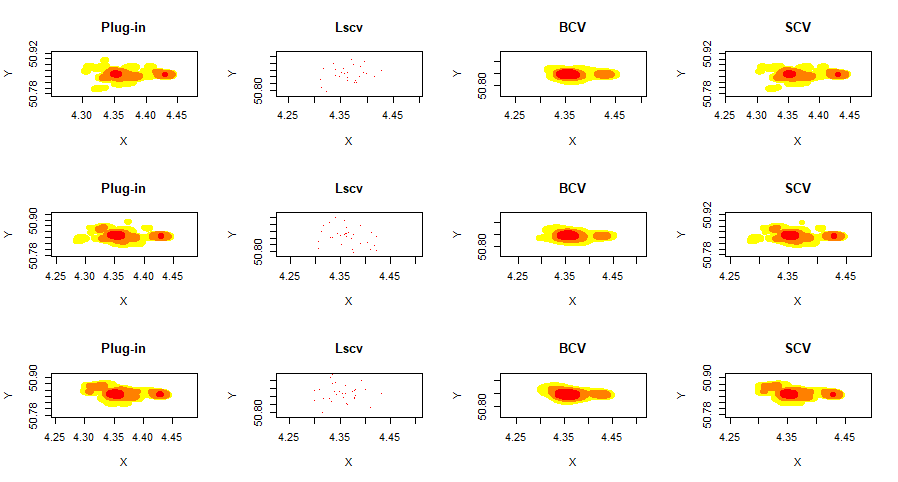
According to the figure showed above, the created density maps from different bandwidth selection algorithm are quite different.

* 1. Map Algebra -- Adebayo

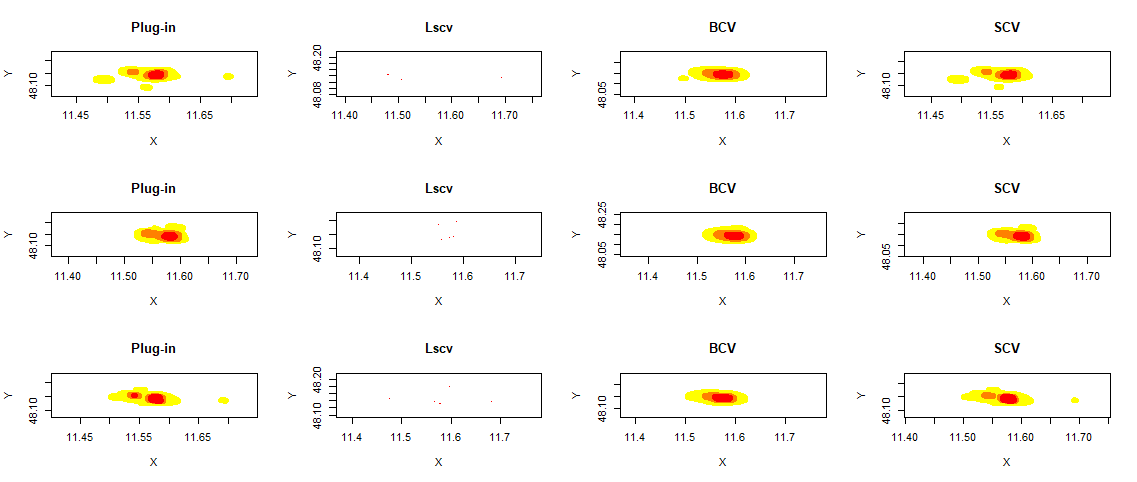
1. Time HeatMap -- Adebayo
   1. Method
   2. Interpretation
2. Create hot spot districts -- Olga
3. Conclusion
4. Appendix
   1. **Bandwidth selection results for three cities**
5. **Brussels**

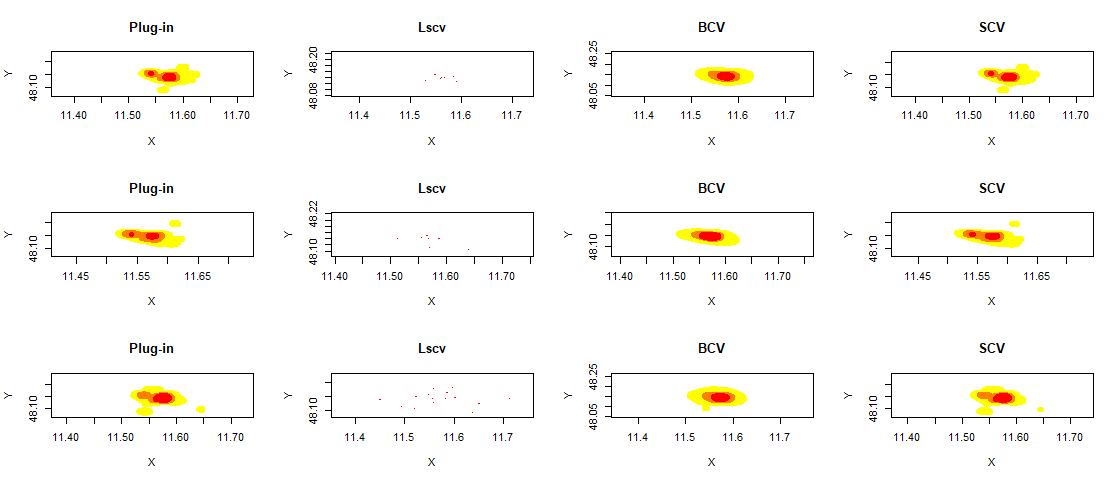


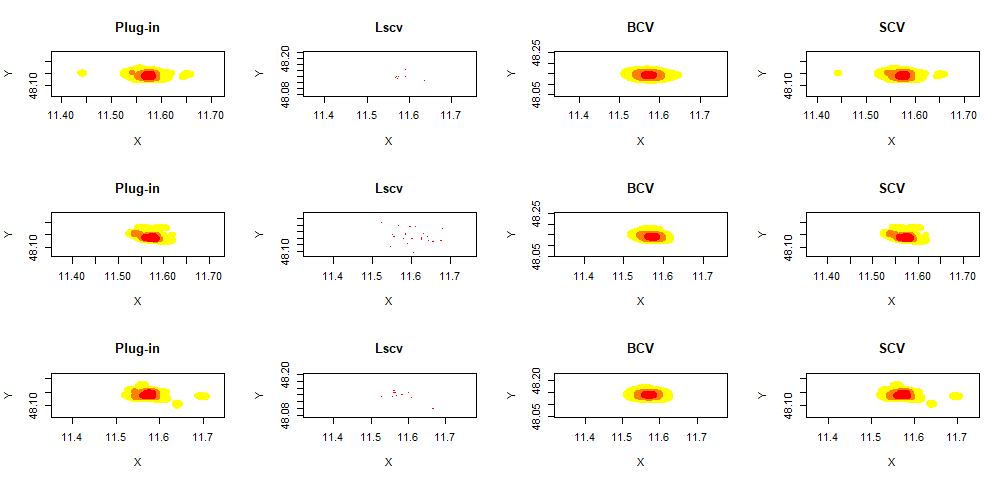


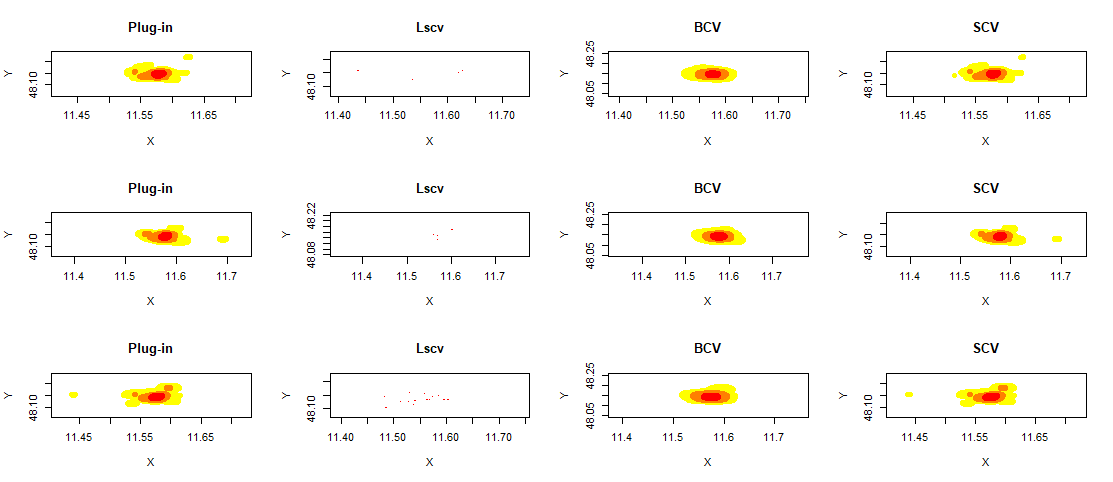


1. **Munich**









1. **Vienna**

