An analysis of the relationship between the jury votes and televotes cast during Eurovision 2021

July 1, 2021

1 Introduction to the research space and project background

The Eurovision Song contest is the largest international song competition held annually in Europe. The participants are representing their country with an original song which is made especially for the contest. Each European country (and some outside Europe) that participate have the possibility of voting their favourite entry. The voting system at Eurovision has changed multiple times since the first time Eurovision took place, in 1958.

The competition itself is a reflection of the unity in diversity and so many cultural backgrounds joined together in one single stage. In 2021, Eurovision was organized in Rotterdam (Holland), giving a message of hope in the particular times of a world pandemic and the winner was Italy, with the song 'Zitti e buoni', performed by the rock band Maneskin.

What is particular about the competition is the fact that art is bringing together nations and each participant country gets to express its votes. An artistic act is a subjective creation by definition and in particular at this big scale, things are even more challenging. During time, the aspect of voting has been an intense topic and different solutions were implemented.

The current voting system is in place since 2016 and represents a mixt between the opinion of the large public and professional juries. Each country is being awarded two sets of points (for the public and the jury) from 1 to 8, 10 and 12 - they can vote for everyone else except their country. There is no 9 and no 11 in the votes.

The professional jury is formed by representatives of each participant country, usually having 4 members with different professional backgrounds related to music.

The public vote is accessible during the show, for a certain period of time. In 2021, the public had the possibility to vote using methods like by sms or via an online application. An interesting aspect regarding this is that you could vote as much as 18 times for your favourite performance, from the same phone number.

Rules regarding both the jury vote and the televote are set in place rigorously and this is just a small summary of how it works, for the purposes of this analysis.

In order to determine the final results, Europe is using two methods: a qualitative one (the professional jury) and the quantitative one (the televotes). This analysis is aiming to compare them with the lenses of Eurovision 2021. Even if it's challenging in itself to decide the value of an artistic performance, Europe is up for this challenge and analysing the results is a way to determine if there are similarities, significant differences or potential nuances to spot.

The objectives of this analysis are the following: * Analysing the winner's final vote from the perspective of the jury vote and the televote. I've chosen to look at the winner's results considering that nobody knows the 'secret formula' of winning Eurovision and it's interesting to check out how this has happened in 2021 with Italy. Obviously this will not be an analysis of their artistic performance, but one of the correlation of the votes that has made Italy the winner. Also, this analysis could be reproduced for any country who participated. I've limited myself to Italy and there is a risk that the analysis is not complete without looking at the situations of other countries. In any case, it's representative for the competition overall, so I think that this limitation is welcomed at this stage and for the purposes of this analysis. A comparison between the jury vote and the televote will be made also with the case of the UK, who received null points in this Eurovision edition. * Analysing the overall distribution of votes from the televote and the jury vote. This will be a way to spot if there are any differences we can notice in the way the professional jury and the public voted, while also having a clear picture of where the votes went. * Analysing the Top 10 according to the televote and the jury vote, while also placing it in perspective with the overall Top 10 of the competition. I've limited the analysis only to the Top 10, for the purposes of this project. A risk would be not to have a clear perspective on the voting system due to this limitation. However, the analysis is aiming to spot if there are some correlations at this level and could be further extended in future research projects, particularly more extensive ones. * Analysing the geographical distribution of countries who have received 12 points from the professional jury and from the public. I've limited myself to only analysing where the 12 points have gone, due to the scope of this project, but this geographical distribution can be extended to every participant country and this could be a great way to explore the surface identified in this current project. However, to note is that 12 points is the highest number of points a country can offer and the most valuable one for determining the overall outcome of the competition. Therefore, it's the best start for analysing the correlation between the jury vote and the public vote. The geographical distribution is particularly interesting to look at, due to significant analysis done in the past on voting trends in the Eurovision song contest, which showed that it has a political component: some countries vote for their neighbours, which is a bias to spot in the overall results. This analysis will look both at the jury votes and the televotes, to see if this bias is common in both.

The field of Eurovision voting system is interesting for its diversity, uniqueness, determining a potential winning formula and identifying voting trends. Significant research was performed in this space and also the competition is publishing results on the official website. An interesting aspect is the betting odds, which in 2021 has placed Italy on top.

A reasonable expectation that I have at the beginning of this analysis is that the professional jury is qualified to make a fair judgement. Therefore, I expect to see a lot of similarities in the way the jury vote went. In the case of the public vote, I think that there is potential for a higher level of subjectivity here, but on a larger scale this might be overcome by more precision in the final outcome, when all the millions of votes are being converted into points. In the way the public is voting there is a bias also considering the heavy promotion some artists received previous to the competition, online presence, exposure, the artist's own fan base which could be significant and located in lots of other countries from Europe apart from the one of their origin. There are also the citizens of a country living abroad, who might vote for their country of origin based on this criterium solely, so there is a risk from this point of view as well. The current analysis is not considering this aspect, but further on an exploration could be made regarding a parallel with imigration statistics.

The steps and stages of my analysis are the following: * Identifying the dataset and pointing

out some characteristics it has / cleaning the data if necessary * Comparing briefly the data for accuracy with some information found online, on Wikipedia, using the web scraping technique. * Making an analysis of the voting results for Italy, using some operations performed on the data set, histograms with the distribution of the votes, a graphical representation of the detailed votes in a plot * Using the null points received by UK in an original way for creating heat maps which will show the overall distribution of votes throughout the final, for all participants * Make bar charts for analysing the final Top 10 according to the jury, the public and overall * Determine the latitude and longitude of countries and showing maps that point to the ones who received 12 points from the jury and from the public.

The questions that I wish to address in this analysis are based on what is the correlation between the jury and the public votes: - received by the winner of Eurovision 2021 - received by countries in general, in the overall competition - in determining the top 10 of the competition - in awarding the maximum of 12 points

2 Considerations about the data analysed

The data that will be analysed originates in two datasets: one with Eurovision jury votes and another with the televotes, both in CSV format. I've merged them together into one single file, as an output, in order to ease the analysis and make a more complete single source of the data to be analysed for the project.

The output CSV file is displaying a significant amount of numerical data representing points that countries have offered to each other in the televote and the jury vote. It also displays the voters and names of each country, to make the data analysed more easy to follow.

The content is sufficient for providing an overall visibility of the votes distribution and offers enough details to answer the main questions of this analysis, which are regarding the votes received by the winner of Eurovision, the overall competition, top 10 and in awarding the maximum 12 points, both in the jury vote and the televote. If the dataset would have contained only the overall results, this would not have been sufficient to make a correlation between the two types of voting present in the competition.

The source of the two CSV files with the jury vote and the televote data is https://www.kaggle.com/prajittr/eurovision-2021-song-competition-voting-patterns , a site which gathers the world's largest data science community. These two CSV files which I've downloaded have been merged into a single output.csv file, which represents the main file used for this analysis.

I also wanted to check the accuracy of the information from this source and for this I've used the web scraping technique from the Wikipedia page entitled 'Eurovision song contest 2021' https://en.wikipedia.org/wiki/Eurovision_Song_Contest_2021 . I've used the information from the fifth table from the page about the final results, from the 'Participating countries' section. I was particularly interested in the column with the total points obtained by the participants. I've used this information for a random check of the accuracy of the data present in the main CSV file I am using for this analysis. In order to do this, I've calculated the sum of the televotes and the jury vote some countries have received with the data present in the total points column from Wikipedia. As the results were identical, I've concluded that the data is accurate after this random check. However, there is a risk some errors might be present, as I did not compare all the data to see if it sums up. Also, the data from Wikipedia is not showing the details for the votes, being

limited only to the overall results. Therefore, a risk of errors still exists, but is not identifiable on the surface that I have directly checked using the web scraping technique.

To note is also that the information about the Eurovision Song contest 2021 is mainly present on the official competition site https://eurovision.tv/ , which shows the results. I did not scrape that one, due to ethical reasons and I have chosen to limit myself to public domain and Wikipedia for finding information (more details in the dedicated section). Because the competition just took place, it's difficult to find accurate data sets to perform an analysis on them, in a proper format.

I've also considered using information about the semifinals in my analysis, particularly from the above mentioned Wikipedia page. However, I've decided to look only at the final results, thinking that the significant audience is present mainly in the final and both the jury and the public are offering a full complete vote, not only in part. For example, it is more relevant to look at how Ukraine did in the final, competing against all the other finalists, than to also check what Ukraine did in the semifinal, where only a part of the finalists were present. In any case, I do believe that an analysis on that would have helped the conclusions, going more in detail, but for the present analysis I've limited myself to the final only.

I have also considered extending the analysis on a broader scale, by looking at all results from the final and semi finals from more Eurovision competitions, from 1975 to 2019. The dataset I've found is on kaggle: https://www.kaggle.com/datagraver/eurovision-song-contest-scores-19752019 , in .xlsx format. This dataset would have been useful for extending the analysis, but it is not updated with the Eurovision 2021 dataset. For further exploration, it would be interesting to look at this data set to see if the conclusions I've reached for the 2021 competition are similar.

3 Ethical aspects regarding the data

The two CSV files forming the output.csv file analysed is licenced under public domain, as we can see here: https://www.kaggle.com/prajittr/eurovision-2021-song-competition-voting-patterns. From the Creative commons website (https://creativecommons.org/publicdomain/zero/1.0/) we can see that this kind of licence is allowing the general public to copy, bring changes to the analysis and even use it for commercial purposes. The current project is using the CSV files with the jury vote and the televote for personal purposes, with the aim to present it as a midterm work for the University of London course Programming with data.

The present project is generating intellectual proprietary rights, particularly for the author of it, who is intrinsically connected to the work performed, as it represents an original creation. For now, it is used for the course and as a midterm submission. It has the potential to generate in the future more intellectual rights, particularly if I would decide to publish it on github, for example, under a licence of my choice. I would contribute this way to the Open Source Software community. I could also publish it on kaggle, connecting it to the Eurovision 2021 song competition patterns, as it has its origins there. I could release it under the public domain, as the original sets of data. However, there is a risk that publishing a midterm submission would violate University of London's code of conduct, an aspect which exceeds the intellectual proprietary topic, but is nonetheless relevant for the discussion and a potential publishing of the project.

The data with the jury vote and the public vote in the Eurovision 2021 contest has the potential to be biased in the topic analysed if we extend it for other years. The conclusions generated by these results are relevant for the 2021 competition. As mentioned previously, art in itself is a subjective matter and judging an artistic performance at this scale is challenging. There is potential for lots

of biases to intervene. Some examples of biases: geo-political voting, various affiliations (e.g. fan bases of the artists, citizens of a certain state living abroad and voting for the state of origin). The votes analysed in this project carry in themselves some biases, but this does not influence the accuracy of the analysis, due to the fact that the number of votes is considered the premise of the analysis on the correlation done.

As the origin of the CSV data is public domain, I also used web scraping a Wikipedia page, for doing some checks about accuracy of the data. The page I've scraped is https://en.wikipedia.org/wiki/Eurovision_Song_Contest_2021#Final_2 , in particular using the table with the final results. Web scraping has the potential of violating terms of services from websites, so it should be done in accordance with the intellectual property limitations. The page scraped has the following mention: "Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy.'. Regarding the license it makes reference to (https://en.wikipedia.org/wiki/Wikipedia:Text_of_Creative_Commons_Attribution-ShareAlike_3.0_Unported_License), we can notice that it fits to the purposes of the present project, considering that I am referencing the source of the information.

Regarding the web scraping technique applied to the Wikipedia page, I have also verified the page https://en.wikipedia.org/robots.txt which is showing the robots exclusion protocol specific for the website. We can notice that the protocol is allowing this kind of scraping applied to the Eurovision page. In any case, there are valid concerns regarding the ethical aspects of scraping websites. This is why I limited myself to applying this technique to a site like Wikipedia, which is specifically known for a careful consideration of intellectual property and prioritizing public access. I chose not to go on the eurovision.tv website, as this would have been more problematic from the point of view of ethical aspects / policy.

An aspect to consider is also that this analysis is an independent one and it might be different from the official analysis or trends released by the competition organisers. I did not check in specific the rules of the competition regarding what is allowed to analyse or not, but I am aware that the voting system is thoroughly monitored and there is a high level of attention offered to the way the points are given / to the topic of what is the best way to vote at Eurovision. The conclusions of this analysis might not be aligned to the official views / releases and this is a risk to point out. In any case, I am trying to maintain a high level of objectivity and good faith in this independent project and not intentionally harm anyone with the conclusions reached.

The data does not contain any personal identifiable information, so from this point of view there is no concern. Though we can identify the artists present in the competition, they are acting in their public role.

4 Preparing and exploring the dataset

I am importing here the relevant libraries for the analysis. I am also merging the two CSV files with the Eurovision jury votes and the televote into one single file, output.csv. This last one will represent the main data which will be further analysed. As the output.csv file came out with column names which had x and y in their naming, I've decided to clean this and rename them. I've also renamed the first column into 'Voters', because it's more space efficient.

[1]:
```
# Importing numpy, pandas and matplotlib.pyplot
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

# Importing folium, Nominatim for the geographical analysis
import folium
from geopy.geocoders import Nominatim
geolocator = Nominatim(user_agent="my_application")

# Importing seaborn for the heatmaps
import seaborn as sns
sns.set_style('whitegrid')

# Read CSV files
juryvotes = pd.read_csv("Eurovision_juryvotes_2021.csv")
televotes = pd.read_csv("Eurovision_televotes_2021.csv")

# Rename first column
a = juryvotes.rename({'Country (Voters (vertical), Finalists (horizontal))':␣
 ↪'Voters'}, axis='columns')
b = televotes.rename({'Country (Voters (vertical), Finalists (horizontal))':␣
 ↪'Voters'}, axis='columns')

# Sort values from b in alphabetical order
b.sort_values('Voters')

# Merge a and b into a single CSV file
merged = a.merge(b, on='Voters')

merged.to_csv("output.csv", index=False)

# Read the new CSV file
df = pd.read_csv('output.csv')

# Replace values '_x' and '_y' with ' Juryvotes' and ' Televotes'
df.columns = df.columns.str.replace('_x', ' Juryvotes')
df.columns = df.columns.str.replace('_y', ' Televotes')
```

Here I am checking the columns names from the file. We can notice that they have clear namings and there is a distinction between the jury votes and the televotes.

[2]:
```
# Read headers
df.columns
```

[2]:
```
Index(['Voters', 'Albania Juryvotes', 'Azerbaijan Juryvotes',
       'Belgium Juryvotes', 'Bulgaria Juryvotes', 'Cyprus Juryvotes',
```

```
       'Finland Juryvotes', 'France Juryvotes', 'Germany Juryvotes',
       'Greece Juryvotes', 'Iceland Juryvotes', 'Israel Juryvotes',
       'Italy Juryvotes', 'Lithuania Juryvotes', 'Malta Juryvotes',
       'Moldova Juryvotes', 'Netherlands Juryvotes', 'Norway Juryvotes',
       'Portugal Juryvotes', 'Russia Juryvotes', 'San Marino Juryvotes',
       'Serbia Juryvotes', 'Spain Juryvotes', 'Sweden Juryvotes',
       'Switzerland Juryvotes', 'Ukraine Juryvotes',
       'United Kingdom Juryvotes', 'Albania Televotes', 'Azerbaijan Televotes',
       'Belgium Televotes', 'Bulgaria Televotes', 'Cyprus Televotes',
       'Finland Televotes', 'France Televotes', 'Germany Televotes',
       'Greece Televotes', 'Iceland Televotes', 'Israel Televotes',
       'Italy Televotes', 'Lithuania Televotes', 'Malta Televotes',
       'Moldova Televotes', 'Netherlands Televotes', 'Norway Televotes',
       'Portugal Televotes', 'Russia Televotes', 'San Marino Televotes',
       'Serbia Televotes', 'Spain Televotes', 'Sweden Televotes',
       'Switzerland Televotes', 'Ukraine Televotes',
       'United Kingdom Televotes'],
      dtype='object')
```

Here we can see the first column 'Voters', formed by each country that voted in the competition. One remark to be made is that the voters are more numerous than the finalists. This is because other participants were eliminated in the semi-finals and only 26 countries have been present in the final with a song.

[3]:
```
# Read first column
print(df['Voters'])
```

```
0            Albania
1          Australia
2            Austria
3         Azerbaijan
4            Belgium
5           Bulgaria
6            Croatia
7             Cyprus
8            Czechia
9            Denmark
10           Estonia
11           Finland
12            France
13           Georgia
14           Germany
15            Greece
16           Iceland
17           Ireland
18            Israel
19             Italy
20            Latvia
```

```
21          Lithuania
22              Malta
23            Moldova
24        Netherlands
25    North Macedonia
26             Norway
27             Poland
28           Portugal
29            Romania
30             Russia
31         San Marino
32             Serbia
33           Slovenia
34              Spain
35             Sweden
36        Switzerland
37            Ukraine
38     United Kingdom
Name: Voters, dtype: object
```

I've also done a check of the information about the file, for more clarity.

[4]:
```
# More info about the dataset
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 39 entries, 0 to 38
Data columns (total 53 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Voters                    39 non-null     object
 1   Albania Juryvotes         39 non-null     int64
 2   Azerbaijan Juryvotes      39 non-null     int64
 3   Belgium Juryvotes         39 non-null     int64
 4   Bulgaria Juryvotes        39 non-null     int64
 5   Cyprus Juryvotes          39 non-null     int64
 6   Finland Juryvotes         39 non-null     int64
 7   France Juryvotes          39 non-null     int64
 8   Germany Juryvotes         39 non-null     int64
 9   Greece Juryvotes          39 non-null     int64
 10  Iceland Juryvotes         39 non-null     int64
 11  Israel Juryvotes          39 non-null     int64
 12  Italy Juryvotes           39 non-null     int64
 13  Lithuania Juryvotes       39 non-null     int64
 14  Malta Juryvotes           39 non-null     int64
 15  Moldova Juryvotes         39 non-null     int64
 16  Netherlands Juryvotes     39 non-null     int64
 17  Norway Juryvotes          39 non-null     int64
 18  Portugal Juryvotes        39 non-null     int64
```

```
 19  Russia Juryvotes          39 non-null     int64
 20  San Marino Juryvotes      39 non-null     int64
 21  Serbia Juryvotes          39 non-null     int64
 22  Spain Juryvotes           39 non-null     int64
 23  Sweden Juryvotes          39 non-null     int64
 24  Switzerland Juryvotes     39 non-null     int64
 25  Ukraine Juryvotes         39 non-null     int64
 26  United Kingdom Juryvotes  39 non-null     int64
 27  Albania Televotes         39 non-null     int64
 28  Azerbaijan Televotes      39 non-null     int64
 29  Belgium Televotes         39 non-null     int64
 30  Bulgaria Televotes        39 non-null     int64
 31  Cyprus Televotes          39 non-null     int64
 32  Finland Televotes         39 non-null     int64
 33  France Televotes          39 non-null     int64
 34  Germany Televotes         39 non-null     int64
 35  Greece Televotes          39 non-null     int64
 36  Iceland Televotes         39 non-null     int64
 37  Israel Televotes          39 non-null     int64
 38  Italy Televotes           39 non-null     int64
 39  Lithuania Televotes       39 non-null     int64
 40  Malta Televotes           39 non-null     int64
 41  Moldova Televotes         39 non-null     int64
 42  Netherlands Televotes     39 non-null     int64
 43  Norway Televotes          39 non-null     int64
 44  Portugal Televotes        39 non-null     int64
 45  Russia Televotes          39 non-null     int64
 46  San Marino Televotes      39 non-null     int64
 47  Serbia Televotes          39 non-null     int64
 48  Spain Televotes           39 non-null     int64
 49  Sweden Televotes          39 non-null     int64
 50  Switzerland Televotes     39 non-null     int64
 51  Ukraine Televotes         39 non-null     int64
 52  United Kingdom Televotes  39 non-null     int64
dtypes: int64(52), object(1)
memory usage: 16.3+ KB
```

Here I checked the heading of the dataset, with the first three rows shown. We can notice the numerous columns which represent in the first half the jury votes received by each country from the heading and in the last half the televotes. The first column is showing the countries who voted.

[5]:
```
#Print first three entries from the dataset
df.head(3)
```

[5]:

	Voters	Albania Juryvotes	Azerbaijan Juryvotes	Belgium Juryvotes	\
0	Albania	0	0	0	
1	Australia	0	6	0	
2	Austria	0	0	0	

```
   Bulgaria Juryvotes  Cyprus Juryvotes  Finland Juryvotes  France Juryvotes  \
0                   0                 7                  3                10
1                   2                 4                  0                 7
2                   5                 0                  1                 8

   Germany Juryvotes  Greece Juryvotes  …  Norway Televotes  \
0                  0                 6  …                 0
1                  0                 0  …                 1
2                  2                 0  …                 1

   Portugal Televotes  Russia Televotes  San Marino Televotes  \
0                   0                 0                     0
1                   0                 0                     0
2                   0                 0                     0

   Serbia Televotes  Spain Televotes  Sweden Televotes  Switzerland Televotes  \
0                 0                0                 1                     12
1                 2                0                 0                      5
2                12                0                 0                      5

   Ukraine Televotes  United Kingdom Televotes
0                  4                         0
1                 10                         0
2                  7                         0

[3 rows x 53 columns]
```

I decided to make a column which will show the total points that Italy received from each country, overall. This was for having a first overview on the scores obtained by the winner.

[20]:
```
# Create a new column with Italy Total points (juryvotes + televotes)
df['Italy Total'] = df['Italy Juryvotes'] + df['Italy Televotes']


print(df[['Voters', 'Italy Juryvotes', 'Italy Televotes', 'Italy Total']])
```

```
            Voters  Italy Juryvotes  Italy Televotes  Italy Total
0          Albania                4               10           14
1        Australia                6                7           13
2          Austria                6                8           14
3       Azerbaijan                0               10           10
4          Belgium                2                8           10
5         Bulgaria               10               12           22
6          Croatia               12               10           22
7           Cyprus                8               10           18
8          Czechia                6                6           12
9          Denmark                0                5            5
```

10	Estonia	3	8	11
11	Finland	6	8	14
12	France	0	10	10
13	Georgia	12	8	20
14	Germany	6	7	13
15	Greece	4	10	14
16	Iceland	7	5	12
17	Ireland	0	6	6
18	Israel	0	7	7
19	Italy	0	0	0
20	Latvia	8	7	15
21	Lithuania	10	10	20
22	Malta	0	12	12
23	Moldova	0	8	8
24	Netherlands	0	2	2
25	North Macedonia	10	8	18
26	Norway	5	7	12
27	Poland	5	10	15
28	Portugal	3	7	10
29	Romania	3	10	13
30	Russia	10	10	20
31	San Marino	10	12	22
32	Serbia	8	12	20
33	Slovenia	12	10	22
34	Spain	0	10	10
35	Sweden	10	3	13
36	Switzerland	8	10	18
37	Ukraine	12	12	24
38	United Kingdom	0	3	3

Further on, if I am making a sum of the values from the new 'Italy Total' column, I am obtaining the overall score that the winner received.

As this can be seen also if I am summing up the values from the two columns with jury votes and televotes, I decided to delete the new column for the following purposes of the analysis. My initial goal was just to have a visual overview of the scores obtained by Italy overall, in one single dataframe.

After performing the deletion, I made a new check of the dataset and it looks ready for the analysis planned further.

[8]:
```
# Calculate the sum of all points received by Italy (juryvotes + televotes)
total = df['Italy Total'].sum()

print (total)
```

```
524
```

[10]:
```
# Alternative way to calculate the sum of all points received by Italy␣
↪(juryvotes + televotes)
total = df['Italy Juryvotes'].sum() + df['Italy Televotes'].sum()

print (total)
```

```
524
```

[22]:
```
#
df = df.drop(columns = ['Italy Total'])

df.head(3)
```

[22]:

	Voters	Albania Juryvotes	Azerbaijan Juryvotes	Belgium Juryvotes	Bulgaria Juryvotes	Cyprus Juryvotes	Finland Juryvotes	France Juryvotes	Germany Juryvotes	Greece Juryvotes	…	Norway Televotes	Portugal Televotes	Russia Televotes	San Marino Televotes	Serbia Televotes	Spain Televotes	Sweden Televotes	Switzerland Televotes	Ukraine Televotes	United Kingdom Televotes
0	Albania	0	0	0	0	7	3	10	0	6	…	0	0	0	0	0	0	1	12	4	0
1	Australia	0	6	0	2	4	0	7	0	0	…	1	0	0	0	2	0	0	5	10	0
2	Austria	0	0	0	5	0	1	8	2	0	…	1	0	0	0	12	0	0	5	7	0

[3 rows x 53 columns]

5 Checking the main dataset for accuracy using web scraping

I've used a script provided in the course in order to be able to scrape a table with total points from Wikipedia.

[118]:
```
# Imports for the web scraping
from bs4 import *
import urllib3
```

[119]:
```
# Script to run for scraping tables
# Source: Programing with Data course, material provided by the tutors
def tableToDataFrame(table):
    # Process header row, building up a data frame with the specified columns
    col_names = ['Index']                # the first col is always the index
    row = table.find('tr')               # get the first row (the header row)
    columns = row.find_all(['td','th'])   # find all the columns in this row
    for column in columns:               # go through all the columns
        text = column.find(text=True)    # get the text from the column - this␣
↪is the column name
        text = text.strip()
        col_names.append(text)           # add the column name to the list of␣
↪column names
    df = pd.DataFrame(columns=col_names) # create a Pandas dataframe with the␣
↪same column structure

    row_index = 0
    first_row = True

    # Find all the rows in the table.  Rows are 'tr' elements within the table
    for row in table.find_all('tr'):
        # Skip the first row, which is the header which we have already␣
↪processed
        if first_row:
            first_row = False
            continue

        # Create a new row in the DataFrame
        df.loc[row_index] =  ''

        # Find all the columns in this row.  Columns are in 'td' elements␣
↪within the row
        columns = row.find_all(['td','th'])

        # Iterate through the columns
        column_index = 1
        for column in columns:
            # Pull out all the text from the column
```

```
            text = "".join(column.findAll(text=True,recursive='True')).strip()
            #text = column.find(text=True)
            #text = text.strip()

            # Set the value in the DataTable cell
            df.iat[row_index,column_index] = text

            column_index += 1
        row_index += 1
    return df
```

[120]:
```
# Scraping the 5th table from the Eurovision Song Contest 2021 Wikipedia page
url = 'https://en.wikipedia.org/wiki/Eurovision_Song_Contest_2021'
http = urllib3.PoolManager()
response = http.request('GET', url)
page = BeautifulSoup(response.data, 'lxml')
table = page.find_all('table', {'class': 'wikitable'})
df1 = tableToDataFrame(table[5])
```

[121]:
```
# Show the scraped data from the table
df1
```

[121]:

	Index Draw	Country	Artist	\
0	01	Cyprus	Elena Tsagrinou	
1	02	Albania	Anxhela Peristeri	
2	03	Israel	Eden Alene	
3	04	Belgium	Hooverphonic	
4	05	Russia	Manizha	
5	06	Malta	Destiny	
6	07	Portugal	The Black Mamba	
7	08	Serbia	Hurricane	
8	09	United Kingdom	James Newman	
9	10	Greece	Stefania	
10	11	Switzerland	Gjon's Tears	
11	12	Iceland[o]	Daði og Gagnamagnið	
12	13	Spain	Blas Cantó	
13	14	Moldova	Natalia Gordienko	
14	15	Germany	Jendrik	
15	16	Finland	Blind Channel	
16	17	Bulgaria	Victoria	
17	18	Lithuania	The Roop	
18	19	Ukraine	Go_A	
19	20	France	Barbara Pravi	
20	21	Azerbaijan	Efendi	
21	22	Norway	Tix	
22	23	Netherlands	Jeangu Macrooy	
23	24	Italy	Måneskin	

```
24         25          Sweden                Tusse
25         26       San Marino             Senhit[l]
```

	Song	Language(s)	Place	Points
0	"El Diablo"	English[h]	16	94
1	"Karma"	Albanian	21	57
2	"Set Me Free"	English[i]	17	93
3	"The Wrong Place"	English	19	74
4	"Russian Woman"	Russian, English	9	204
5	"Je me casse"	English[k]	7	255
6	"Love Is on My Side"	English	12	153
7	"Loco Loco"	Serbian[p]	15	102
8	"Embers"	English	26	0
9	"Last Dance"	English	10[q]	170
10	"Tout l'univers"	French	3	432
11	"10 Years"	English	4	378
12	"Voy a quedarme"	Spanish	24	6
13	"Sugar"	English	13	115
14	"I Don't Feel Hate"	English[r]	25	3
15	"Dark Side"	English	6	301
16	"Growing Up Is Getting Old"	English	11	170
17	"Discoteque"	English	8	220
18	"Shum" ( )	Ukrainian	5	364
19	"Voilà"	French	2	499
20	"Mata Hari"	English[j]	20	65
21	"Fallen Angel"	English	18	75
22	"Birth of a New Age"	English, Sranan Tongo	23	11
23	"Zitti e buoni"	Italian	1	524
24	"Voices"	English	14	109
25	"Adrenalina"	English[m]	22	50

[27]:
```
# Read columns
df1.columns
```

[27]:
```
Index(['Index', 'Draw', 'Country', 'Artist', 'Song', 'Language(s)', 'Place',
       'Points'],
      dtype='object')
```

We notice that the scraped data is having some columns which are not relevant for the analysis. The goal is to have the column with the points in order to check the main dataset for accuracy. I deleted the columns which were not relevant and I've left also the place column, just for informative reasons.

[28]:
```
# Drop columns & print the result
df1 = df1.drop(columns=['Index', 'Draw', 'Artist', 'Song', 'Language(s)'])
df1
```

[28]:

	Country	Place	Points
0	Cyprus	16	94
1	Albania	21	57
2	Israel	17	93
3	Belgium	19	74
4	Russia	9	204
5	Malta	7	255
6	Portugal	12	153
7	Serbia	15	102
8	United Kingdom	26	0
9	Greece	10[q]	170
10	Switzerland	3	432
11	Iceland[o]	4	378
12	Spain	24	6
13	Moldova	13	115
14	Germany	25	3
15	Finland	6	301
16	Bulgaria	11	170
17	Lithuania	8	220
18	Ukraine	5	364
19	France	2	499
20	Azerbaijan	20	65
21	Norway	18	75
22	Netherlands	23	11
23	Italy	1	524
24	Sweden	14	109
25	San Marino	22	50

[31]:
```
# Sort countries alpabetically
df1 = df1.sort_values('Country')
print (df1)
```

	Country	Place	Points
1	Albania	21	57
20	Azerbaijan	20	65
3	Belgium	19	74
16	Bulgaria	11	170
0	Cyprus	16	94
15	Finland	6	301
19	France	2	499
14	Germany	25	3
9	Greece	10[q]	170
11	Iceland[o]	4	378
2	Israel	17	93
23	Italy	1	524
17	Lithuania	8	220
5	Malta	7	255
13	Moldova	13	115

```
22     Netherlands     23     11
21          Norway     18     75
6         Portugal     12    153
4           Russia      9    204
25      San Marino     22     50
7           Serbia     15    102
12           Spain     24      6
24          Sweden     14    109
10     Switzerland      3    432
18         Ukraine      5    364
8   United Kingdom     26      0
```

After sorting the countries in an alphabetical order to make the data set more similar to the main one from the output.csv file, I am calculating the sum of the televotes and the jury votes from the first 5 countries from the original data set. We can notice that the sum is the same as the number of points displayed in the Wikipedia data.

We can conclude at this point that the data is accurate, from this point of view and under the limitations explained already in the introductory sections.

[32]:
```
# Calculate the sum of juryvotes and televotes for the country from the␣
↪original dataframe
df['Albania Juryvotes'].sum() + df['Albania Televotes'].sum()
```

[32]: 57

[33]:
```
df['Azerbaijan Juryvotes'].sum() + df['Azerbaijan Televotes'].sum()
```

[33]: 65

[34]:
```
df['Belgium Juryvotes'].sum() + df['Belgium Televotes'].sum()
```

[34]: 74

[35]:
```
df['Bulgaria Juryvotes'].sum() + df['Bulgaria Televotes'].sum()
```

[35]: 170

[36]:
```
df['Cyprus Juryvotes'].sum() + df['Cyprus Televotes'].sum()
```

[36]: 94

6 Analysis of the votes received by the winner of Eurovision 2021

This section will answer the first question of the analysis, regarding the correlation between the votes obtained by the winner of the Eurovision 2021, Italy, in the jury voting and the public one.

I'm starting with calculating the mean of the votes obtained from the jury, the public and overall. We can notice that there is a significant difference in the averages obtained, compared to the maximum of 12 points that one participant can give. One might expect that the winner of the

competition is receving points that add to an average close to the maximum, but this is not the case, as we ca see here.

We can conclude by now that: * the jury votes are smaller than the televotes average * both the jury votes and the televotes are mixed - they are not reaching closely the maximum 12 points that a country can give to a participant; neither of them get over the 10 points barrier * the mean of all points received by Italy is showing that other countries have received as well a mix of points, 13.x being the maximum average of the competition, considering that Italy is the winner

[38]:
```
# Calculate the mean of Italy juryvotes
df['Italy Juryvotes'].mean()
```

[38]:
```
5.282051282051282
```

[39]:
```
# Calculate the mean of Italy televotes
df['Italy Televotes'].mean()
```

[39]:
```
8.153846153846153
```

[40]:
```
# Calculate the mean of all points received by Italy (juryvotes + televotes)
df['Italy Juryvotes'].mean() + df['Italy Televotes'].mean()
```

[40]:
```
13.435897435897434
```

As 12 points is the maximum a country can offer to a participant, I've calculated further to see if there is a country that has given 12 points to Italy both in the jury vote and the televote.

The result is showing that only Ukraine saw Italy as being the winner of the competition, both in the jury vote and the televote. This is an interesting result, considering that each country is giving their own view of who should be the winner of the competition. Only the jury and the public from Ukraine has reached this level of equilibrium between the votes and the overall result.

[41]:
```
# Show which countries gave Italy 12 points both in the juryvoting and the
↪televoting
df.loc[(df['Italy Juryvotes'] == 12) & (df['Italy Televotes'] == 12) ]
```

[41]:
```
     Voters  Albania Juryvotes  Azerbaijan Juryvotes  Belgium Juryvotes  \
37  Ukraine                  0                     0                  6

    Bulgaria Juryvotes  Cyprus Juryvotes  Finland Juryvotes  France Juryvotes  \
37                   0                 0                  0                10

    Germany Juryvotes  Greece Juryvotes  …  Norway Televotes  \
37                  0                 0  …                 0

    Portugal Televotes  Russia Televotes  San Marino Televotes  \
37                   0                 4                     0

    Serbia Televotes  Spain Televotes  Sweden Televotes  \
```

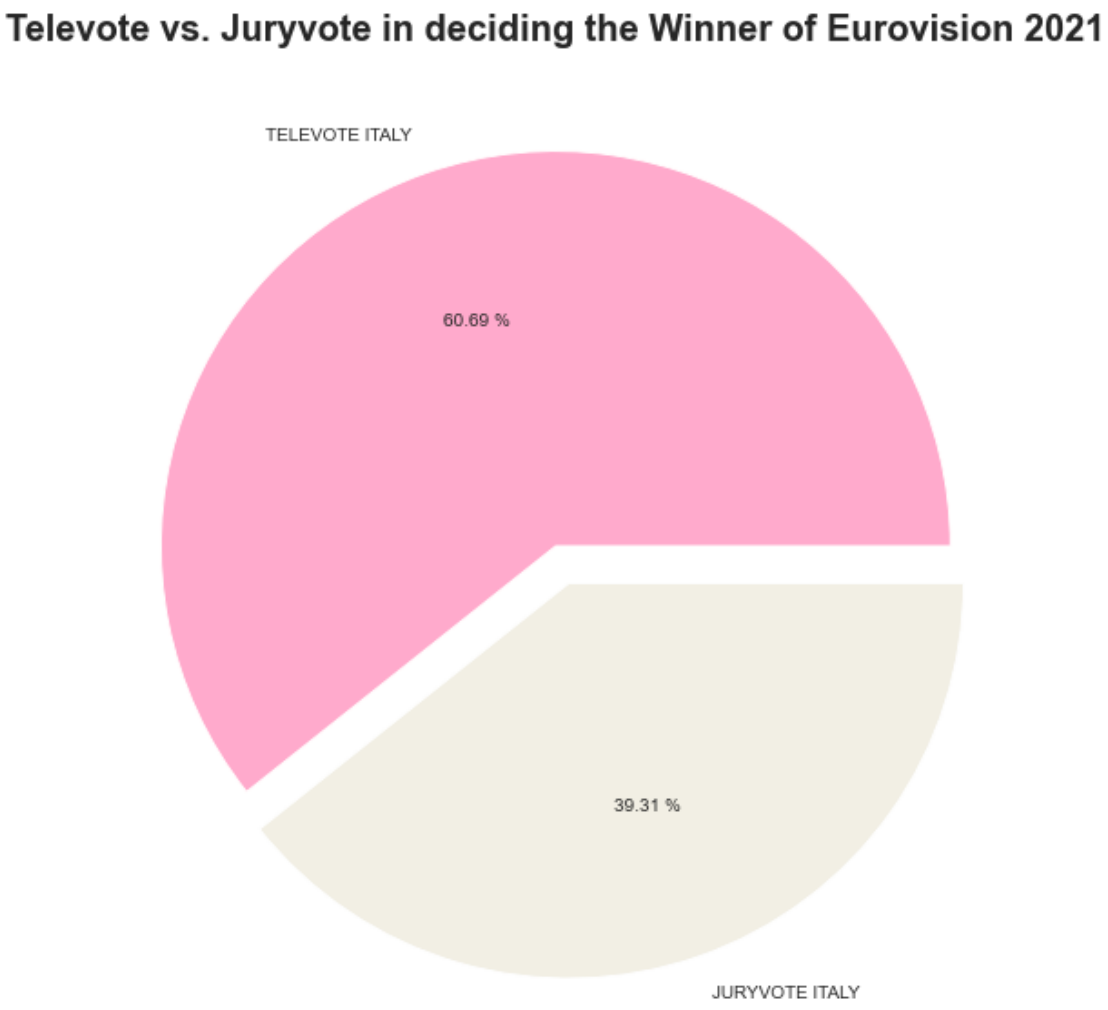
```
37                  0                  0                  2

    Switzerland Televotes  Ukraine Televotes  United Kingdom Televotes
37                      7                  0                         0

[1 rows x 53 columns]
```

I've created further a pie chart which calculates the percentage brought in the overall result by the jury vote and the televote, followed by a detailed sum of each of them.

The result is showing again the difference between the televote and the jury vote, the proportions being with around 20% different.

We can conclude that the public has contributed significantly in choosing the winner of Eurovision 2021, while also being significantly different than the professional jury vote. This raises a question regarding who did the jury saw as a winner of the competition, an answer which will be analysed in a further section.

[42]:
```
fig=plt.figure(figsize=(14, 10))

televote = df['Italy Televotes'].sum()
juryvote = df['Italy Juryvotes'].sum()

labels = ['TELEVOTE ITALY', 'JURYVOTE ITALY']
colors = ['#ffaacc', '#f2efe4']
explode = (0, .1)

plt.pie([televote, juryvote],
        labels = labels,
        colors = colors,
        autopct='%.2f %%',
        explode = explode)

plt.title('Televote vs. Juryvote in deciding the Winner of Eurovision 2021',
         fontdict={'fontweight':'bold',
                   'fontsize': 20})

plt.show()
```



[45]:
```
# Sum of Italy Juryvotes
df['Italy Juryvotes'].sum()
```

[45]: 206

[46]:
```
# Sum of Italy Televotes
df['Italy Televotes'].sum()
```

[46]: 318

I also wanted to find out the overall distribution of points given by the jury and the public. For this I've made a count and the results were surprising: 10 professional jury votes have given 0 points to Italy (the 0 that Italy gave for itself is exluded) and only 4 have given the maximum 12 points. In fact, 0 is the most type of points Italy has received from the jury, followed by 6 of 10 points.

[47]:
```
# Calculate how many of each of type of points (0 - 12, excluding 9, 11) Italy␣
↪received in the juryvote
df['count'] = 1

df.groupby(['Italy Juryvotes']).count()['count']
```

[47]:
```
Italy Juryvotes
0     11
2      1
3      3
4      2
5      2
6      5
7      1
8      4
10     6
12     4
Name: count, dtype: int64
```

In the public vote, we notice that 10 is the most frequent type of points received by Italy and only one 0 was given.

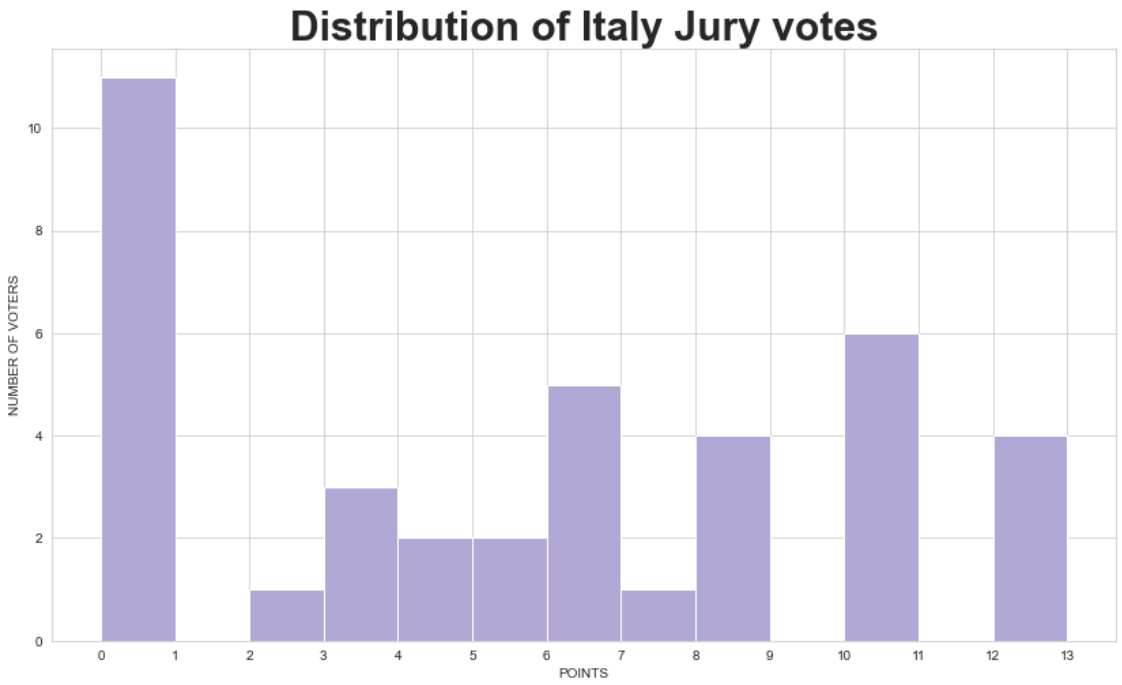
Though overall the public vote was significantly more favorable to Italy than the jury vote, we can notice that only 5 countries have awarded the maximum 12 points in the televote to Italy.

[48]:
```
# Calculate how many of each of type of points (0 - 12, excluding 9,11) Italy␣
↪received in the televote
df['count'] = 1

df.groupby(['Italy Televotes']).count()['count']
```

[48]:
```
Italy Televotes
0      1
2      1
3      2
5      2
6      2
7      6
8      7
10    13
12     5
Name: count, dtype: int64
```

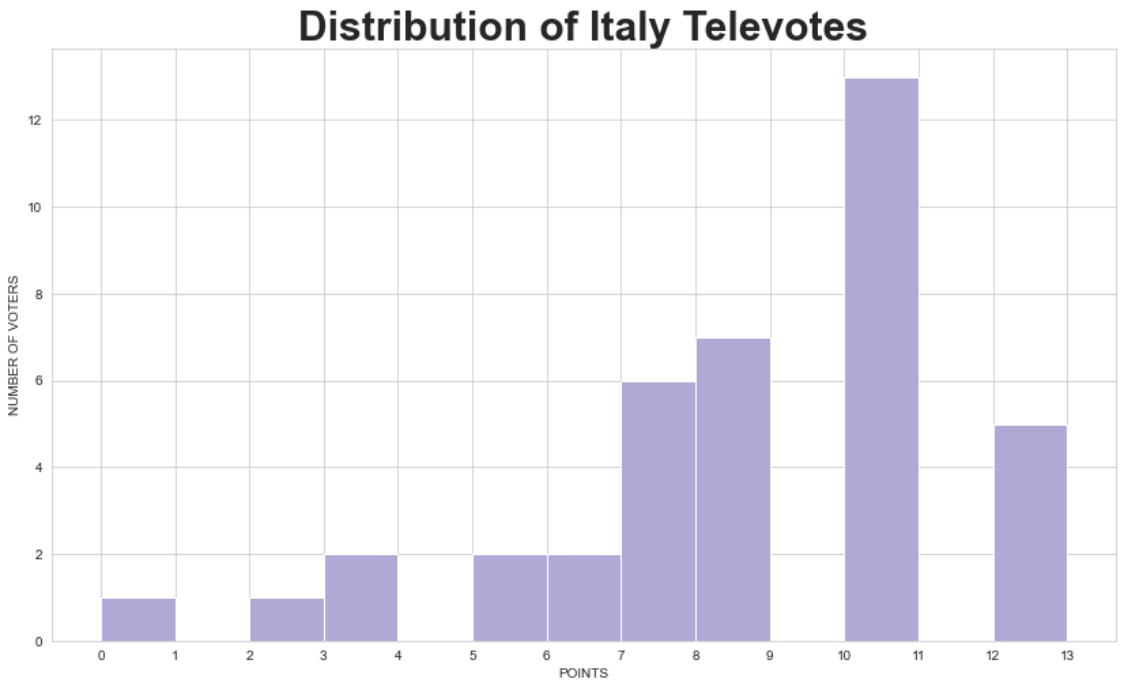
I've decided to create histograms with the Italy jury votes and the televotes, for a better visualization of the significant differences between the two. We can notice the significant proportion of 0 points received by Italy in the professional jury vote and also the significant number of 10 points received in the televote.

[128]:
```
# Histogram showing the distribution of Italy Juryvotes
fig=plt.figure(figsize=(14,8))

# Added extra bins 9, 11 and 13 for accuracy of the histogram
## 9 and 11 points are never offered in Eurovision, 12 is maximum
bins = [0,1,2,3,4,5,6,7,8,9,10,11,12,13]

plt.hist(df['Italy Juryvotes'], bins = bins, color = '#b2a8d5', align = 'mid')

plt.xticks(bins)

plt.ylabel('NUMBER OF VOTERS')
plt.xlabel('POINTS')
plt.title('Distribution of Italy Jury votes',
          fontdict={'fontweight':'bold',
                    'fontsize': 30})

plt.show()
```



[125]:
```
# Histogram showing the distribution of Italy Televotes
fig=plt.figure(figsize=(14,8))

# Added extra bins 9, 11 and 13 for accuracy of the histogram
## 9 and 11 points are never offered in Eurovision, 12 is maximum
bins = [0,1,2,3,4,5,6,7,8,9,10,11,12,13]
```

```
plt.hist(df['Italy Televotes'], bins = bins, color = '#b2a8d5', align = 'mid')

plt.xticks(bins)

plt.ylabel('NUMBER OF VOTERS')
plt.xlabel('POINTS')
plt.title('Distribution of Italy Televotes',
          fontdict={'fontweight':'bold',
                    'fontsize': 30})

plt.show()
```

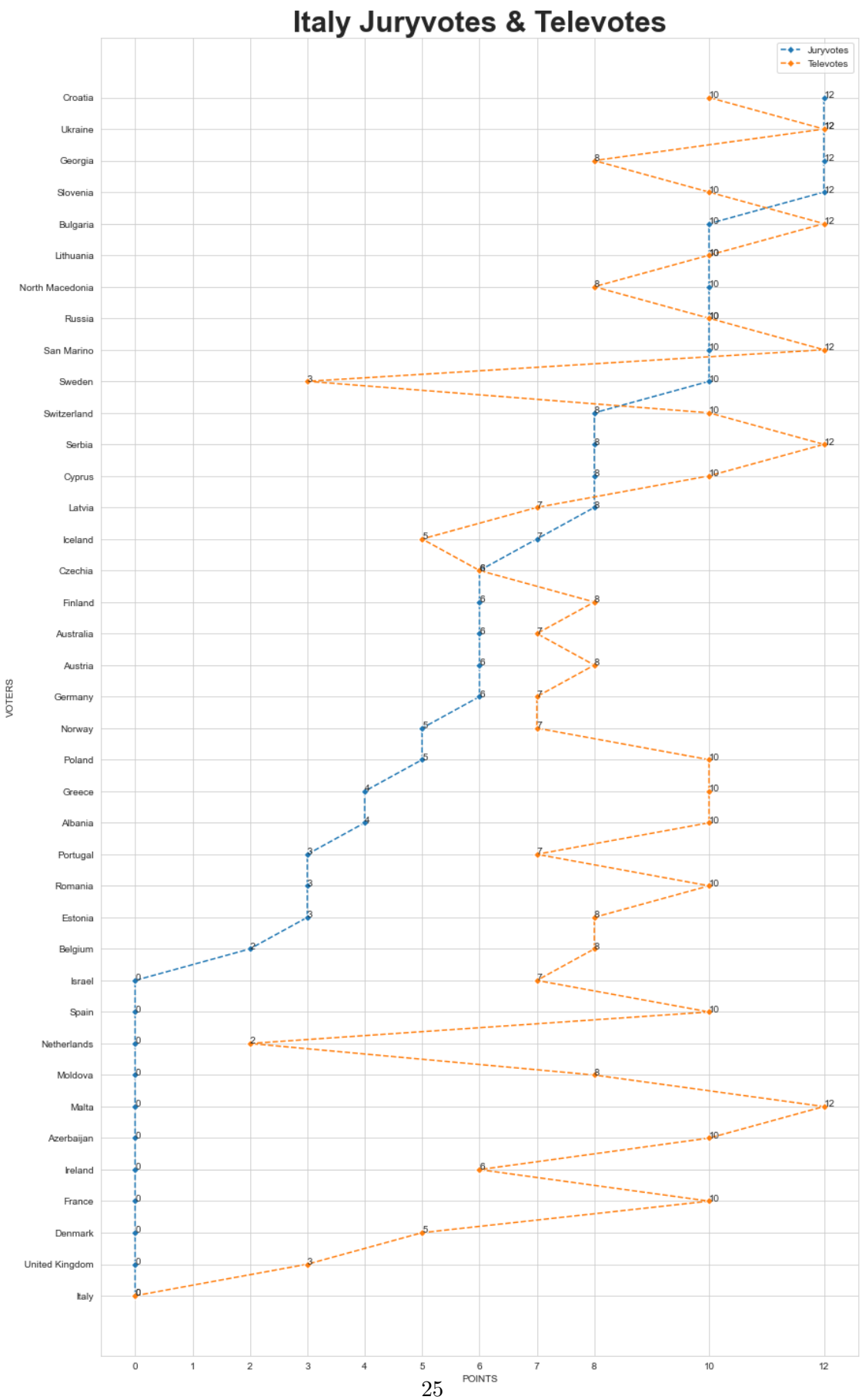


I've created a graphical visualization of all points received by Italy, having the y axis ordered by the lowest to the highest number of jury votes offered.

I've decided to organize them based on the jury votes because the professional evaluation of the artistic performance seemed more objective, being a qualitative result as opposed to a quantitative one. This is questionable, in any case and the decision to organise them like this was made more for a better visualization than the prevalence of one type of vote over the other.

[126]:
```
# Plot to show the overall votes Italy received from all countries
fig = plt.figure(figsize=(14, 25))
ax = fig.add_subplot(111)

plt.title('Italy Juryvotes & Televotes',
```

```
          fontdict={'fontweight':'bold',
                    'fontsize': 30})

# Part of arranging the voters based on their juryvote
numberbyname = {}
for i, j in df['Voters'].items():
    numberbyname[j] = i

df = df.sort_values(['Italy Juryvotes'])

plt.plot(df['Italy Juryvotes'],
         df.Voters,
         label='Juryvotes',
         marker='D',
         markeredgecolor='white',
         linestyle="--")
plt.plot(df['Italy Televotes'],
         df.Voters,
         label='Televotes',
         marker='D',
         markeredgecolor='white',
         linestyle="--")

plt.xticks([0, 1, 2, 3, 4, 5, 6, 7, 8, 10, 12])
plt.xlabel('POINTS')
plt.ylabel('VOTERS')

# Annotating the votes on the specific line (juryvotes or televotes)
k = 0
for i,j in df['Italy Juryvotes'].items():
    ax.annotate(str(j), xy=(j, k))
    k = k + 1

k = 0
for i,j in df['Italy Televotes'].items():
    ax.annotate(str(j), xy=(j, k))
    k = k + 1

plt.legend()

# If we want to save the image, we ca use plt.
 ↪savefig('Italy_Juryvotes_Televotes.png', dpi=300)

plt.show()
```


Italy Juryvotes & Televotes
Juryvotes
Televotes
VOTERS
POINTS
Croatia
Ukraine
Georgia
Slovenia
Bulgaria
Lithuania
North Macedonia
Russia
San Marino
Sweden
Switzerland
Serbia
Cyprus
Latvia
Iceland
Czechia
Finland
Australia
Austria
Germany
Norway
Poland
Greece
Albania
Portugal
Romania
Estonia
Belgium
Israel
Spain
Netherlands
Moldova
Malta
Azerbaijan
Ireland
France
Denmark
United Kingdom
Italy
0 1 2 3 4 5 6 7 8 10 12

To notice in this graph is that Italy is marked with 0 points only because it was not allowed to vote for itself in the competition.

The findings of this graphical visualization are the following: * both the UK and Netherlands public and jury were not impressed by Italy's performance, given 0 points from the jury and close to 0 points in the public vote (3 and 2) * we notice the significant proportion of 0 votes offered by 10 countries in the professional jury * Ukraine was perfectly convinced that Italy should win, both in the jury voting and according to the televote * Malta had the most contradictory views: 0 points from the jury versus 12 points from the public * the jury voted the same as the public in the following cases: the 6 points from the Check Republic, and the 10 points from Russia and Lithuania. All the other countries had different types of points to offer to Italy. * there is a significant difference between the lowest jury votes and the votes from the public: the public voted significantly different than the jury vote, who was particularly harsh to Italy.

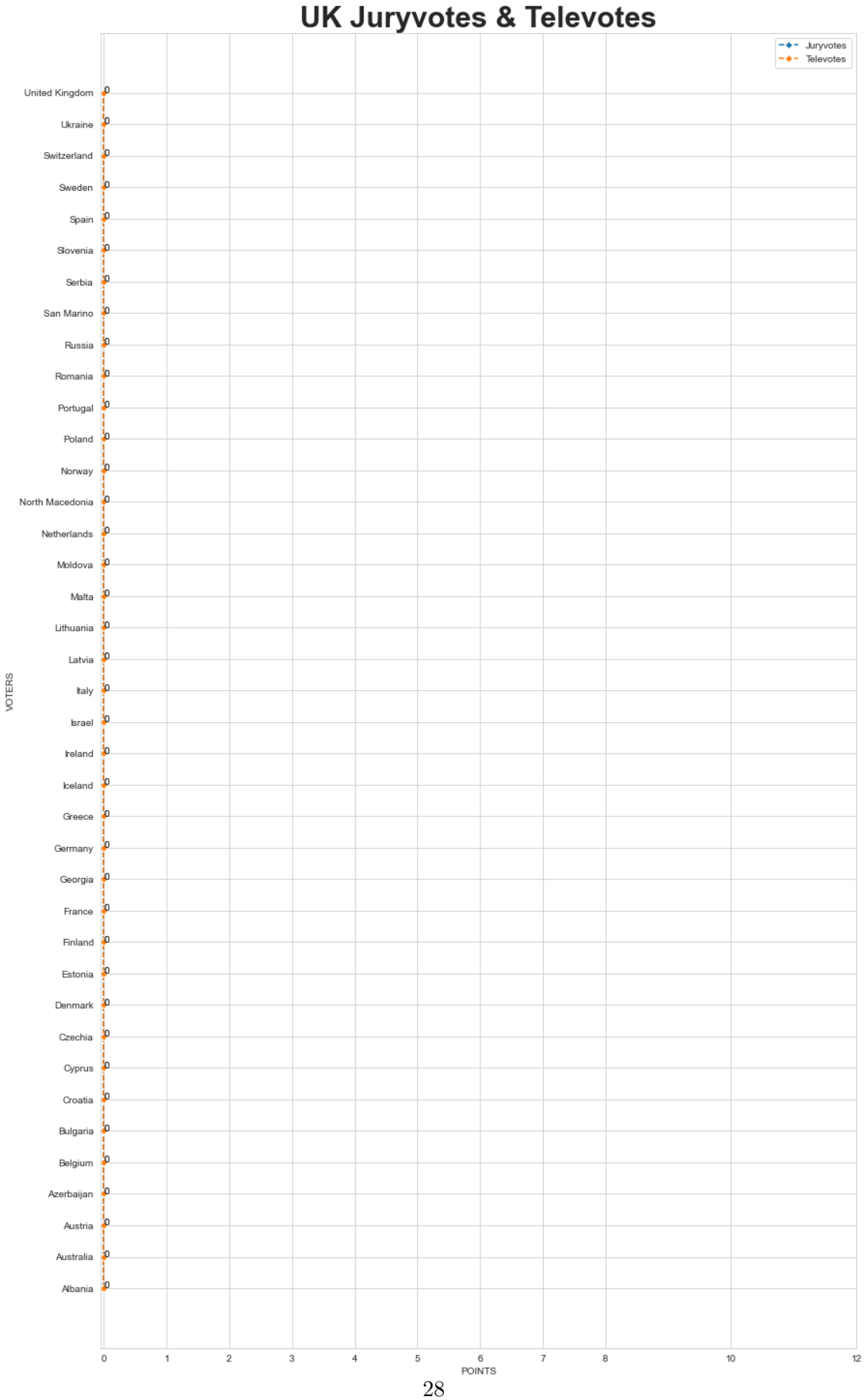
After noticing all these differences between the way the public and the jury voted for Italy, I've decided to make a 180 degrees turn and look at the bottom of the results. We can see a perfect equilibrium in the case of UK: both the jury and the public has given 0 points for the performance.

This is particularly interesting considering the significant differences we see in Italy's case. To notice such lack of division in awarding the null points can be surprising.

However, I think that the null points do not reflect clearly an equilibrium in the evaluations made by voters. We can remain wondering what would have been the results if the countries had to actually award some points to UK and how much these points would coincide.

[127]:
```
# Plotting UK overall null points
fig = plt.figure(figsize=(14, 25))
ax = fig.add_subplot(111)

plt.title('UK Juryvotes & Televotes',
          fontdict={'fontweight':'bold',
                    'fontsize': 30})

df = df.sort_values('Voters')

plt.plot(df['United Kingdom Juryvotes'],
         df.Voters,
         label='Juryvotes',
         marker='D',
         markeredgecolor='white',
         linestyle="--")
plt.plot(df['United Kingdom Televotes'],
         df.Voters,
         label='Televotes',
         marker='D',
         markeredgecolor='white',
         linestyle="--")
```

```
plt.xticks([0, 1, 2, 3, 4, 5, 6, 7, 8, 10, 12])
plt.xlabel('POINTS')
plt.ylabel('VOTERS')

for i,j in df['United Kingdom Juryvotes'].items():
    ax.annotate(str(j), xy=(j, i))

for i,j in df['United Kingdom Televotes'].items():
    ax.annotate(str(j), xy=(j, i))

plt.legend()

#plt.savefig('Italy_Juryvotes_Televotes.png', dpi=300)

plt.show()
```


UK Juryvotes & Televotes
Juryvotes
Televotes
United Kingdom
Ukraine
Switzerland
Sweden
Spain
Slovenia
Serbia
San Marino
Russia
Romania
Portugal
Poland
Norway
North Macedonia
Netherlands
Moldova
Malta
Lithuania
Latvia
Italy
Israel
Ireland
Iceland
Greece
Germany
Georgia
France
Finland
Estonia
Denmark
Czechia
Cyprus
Croatia
Bulgaria
Belgium
Azerbaijan
Austria
Australia
Albania
0
VOTERS
0 1 2 3 4 5 6 7 8 10 12
POINTS

7 Analysis of the votes countries received in the overall competition

I've decided to use UK's null points in an original way in order to make heatmaps showing th votes received by countries in the jury vote, the televote and overall.

I've created dataframes which will be used separately for the corresponding heatmaps.

[59]:
```
# Create a dataframe with juryvotes and print the columns
df_juryvotes = df[['Voters', 'Albania Juryvotes', 'Azerbaijan Juryvotes',
       'Belgium Juryvotes', 'Bulgaria Juryvotes', 'Cyprus Juryvotes',
       'Finland Juryvotes', 'France Juryvotes', 'Germany Juryvotes',
       'Greece Juryvotes', 'Iceland Juryvotes', 'Israel Juryvotes',
       'Italy Juryvotes', 'Lithuania Juryvotes', 'Malta Juryvotes',
       'Moldova Juryvotes', 'Netherlands Juryvotes', 'Norway Juryvotes',
       'Portugal Juryvotes', 'Russia Juryvotes', 'San Marino Juryvotes',
       'Serbia Juryvotes', 'Spain Juryvotes', 'Sweden Juryvotes',
       'Switzerland Juryvotes', 'Ukraine Juryvotes',
       'United Kingdom Juryvotes']]
print(df_juryvotes.columns)
```

```
Index(['Voters', 'Albania Juryvotes', 'Azerbaijan Juryvotes',
       'Belgium Juryvotes', 'Bulgaria Juryvotes', 'Cyprus Juryvotes',
       'Finland Juryvotes', 'France Juryvotes', 'Germany Juryvotes',
       'Greece Juryvotes', 'Iceland Juryvotes', 'Israel Juryvotes',
       'Italy Juryvotes', 'Lithuania Juryvotes', 'Malta Juryvotes',
       'Moldova Juryvotes', 'Netherlands Juryvotes', 'Norway Juryvotes',
       'Portugal Juryvotes', 'Russia Juryvotes', 'San Marino Juryvotes',
       'Serbia Juryvotes', 'Spain Juryvotes', 'Sweden Juryvotes',
       'Switzerland Juryvotes', 'Ukraine Juryvotes',
       'United Kingdom Juryvotes'],
      dtype='object')
```

[130]:
```
# Group the juryvotes by UK's juryvotes
votes = df_juryvotes.groupby(by=['United Kingdom Juryvotes']).sum()
votes.head()
```

[130]:

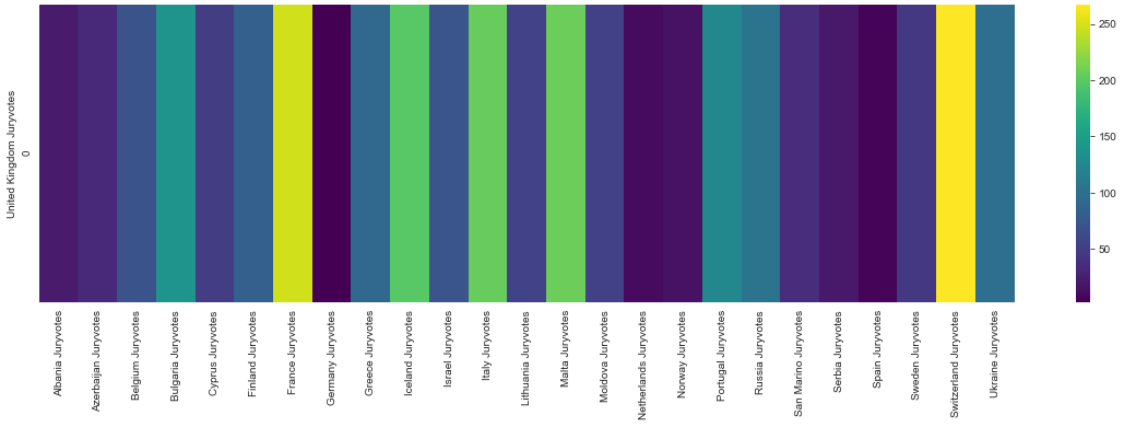
United Kingdom Juryvotes	Albania Juryvotes	Azerbaijan Juryvotes	Belgium Juryvotes	Bulgaria Juryvotes
0	22	32	71	140

```
                          Cyprus Juryvotes  Finland Juryvotes  \
United Kingdom Juryvotes
0                                       50                 83

                          France Juryvotes  Germany Juryvotes  \
United Kingdom Juryvotes
0                                      248                  3

                          Greece Juryvotes  Iceland Juryvotes  …  \
United Kingdom Juryvotes                                      …
0                                       91                198  …

                          Netherlands Juryvotes  Norway Juryvotes  \
United Kingdom Juryvotes
0                                            11                15

                          Portugal Juryvotes  Russia Juryvotes  \
United Kingdom Juryvotes
0                                        126               104

                          San Marino Juryvotes  Serbia Juryvotes  \
United Kingdom Juryvotes
0                                          37                20

                          Spain Juryvotes  Sweden Juryvotes  \
United Kingdom Juryvotes
0                                       6                46

                          Switzerland Juryvotes  Ukraine Juryvotes
United Kingdom Juryvotes
0                                           267                 97

[1 rows x 25 columns]
```

The heatmap displayed here is showing the jury vote result, mapped on the axis of the 0 point received by UK. We can notice the following aspects: * Switzerland was the winner of the jury vote and France was on the second place * Germany, Netherlands and SPain received particularly low points from the jury vote * Italy is close to Iceland and Malta in terms of results obtained

[131]:
```
# Draw a heatmap showing overall juryvotes on the scale of the 0 points for UK
plt.figure(figsize=(20,5))
sns.heatmap(votes, cmap='viridis')
```

[131]:
```
<AxesSubplot:ylabel='United Kingdom Juryvotes'>
```



[63]:
```
# Create a dataframe with televotes and print the columns
df_televotes = df[['Voters', 'Albania Televotes', 'Azerbaijan Televotes',
       'Belgium Televotes', 'Bulgaria Televotes', 'Cyprus Televotes',
       'Finland Televotes', 'France Televotes', 'Germany Televotes',
       'Greece Televotes', 'Iceland Televotes', 'Israel Televotes',
       'Italy Televotes', 'Lithuania Televotes', 'Malta Televotes',
       'Moldova Televotes', 'Netherlands Televotes', 'Norway Televotes',
       'Portugal Televotes', 'Russia Televotes', 'San Marino Televotes',
       'Serbia Televotes', 'Spain Televotes', 'Sweden Televotes',
       'Switzerland Televotes', 'Ukraine Televotes',
       'United Kingdom Televotes']]
print(df_televotes.columns)
```

```
Index(['Voters', 'Albania Televotes', 'Azerbaijan Televotes',
       'Belgium Televotes', 'Bulgaria Televotes', 'Cyprus Televotes',
       'Finland Televotes', 'France Televotes', 'Germany Televotes',
       'Greece Televotes', 'Iceland Televotes', 'Israel Televotes',
       'Italy Televotes', 'Lithuania Televotes', 'Malta Televotes',
       'Moldova Televotes', 'Netherlands Televotes', 'Norway Televotes',
       'Portugal Televotes', 'Russia Televotes', 'San Marino Televotes',
       'Serbia Televotes', 'Spain Televotes', 'Sweden Televotes',
       'Switzerland Televotes', 'Ukraine Televotes',
       'United Kingdom Televotes'],
      dtype='object')
```

[132]:
```
# Group the televotes by UK's televotes
votes = df_televotes.groupby(by=['United Kingdom Televotes']).sum()
votes.head()
```

[132]:

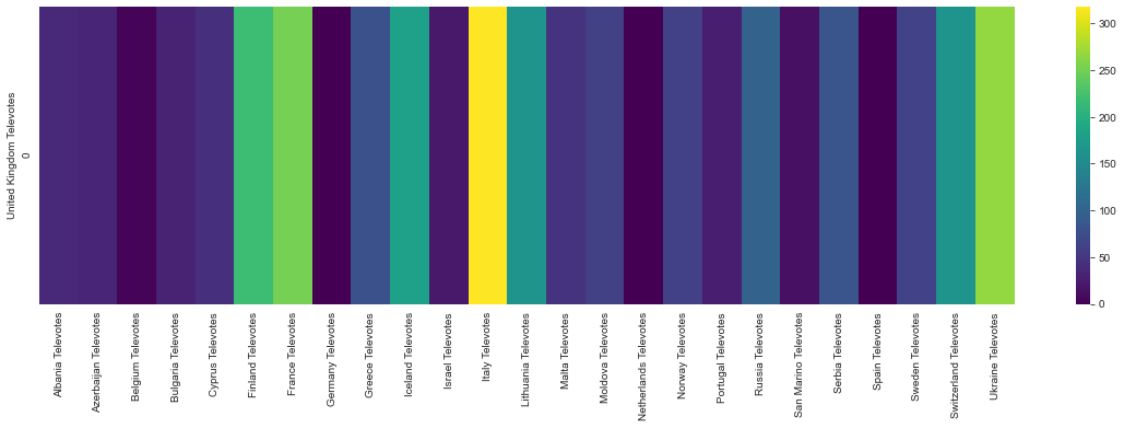
United Kingdom Televotes	Albania Televotes	Azerbaijan Televotes	\
0	35	33	

```
                         Belgium Televotes  Bulgaria Televotes  \
United Kingdom Televotes
0                                        3                  30

                         Cyprus Televotes  Finland Televotes  \
United Kingdom Televotes
0                                       44                218

                         France Televotes  Germany Televotes  \
United Kingdom Televotes
0                                      251                  0

                         Greece Televotes  Iceland Televotes  …  \
United Kingdom Televotes                                       …
0                                       79                180  …

                         Netherlands Televotes  Norway Televotes  \
United Kingdom Televotes
0                                            0                60

                         Portugal Televotes  Russia Televotes  \
United Kingdom Televotes
0                                        27               100

                         San Marino Televotes  Serbia Televotes  \
United Kingdom Televotes
0                                          13                82

                         Spain Televotes  Sweden Televotes  \
United Kingdom Televotes
0                                       0                63

                         Switzerland Televotes  Ukraine Televotes
United Kingdom Televotes
0                                          165                267

[1 rows x 25 columns]
```

In this second heatmap we can notice the public vote. The remarks to make are the following: * Italy is the winner of the public vote * France is on the third place, close to the second place awarded by the jury * Switzerland is not close to the podium according to the public, though it has won the jury vote * Germany, Netherlands and Spain have received 0 points as well, which is close to what the jury vote was in their case as well. It seems that the jury vote and the public vote seem to be particularly similar in the case of the countries from the last places of the competition. * we can notice some surprises in the public vote: Ukraine and Finland scored particularly well here, as opposed to their results from the professional jury

[133]:
```
# Draw a heatmap showing overall televotes on the scale of the 0 points for UK
plt.figure(figsize=(20,5))
sns.heatmap(votes,cmap='viridis')
```

[133]: <AxesSubplot:ylabel='United Kingdom Televotes'>



This last heatmap is showing the overall votes awarded in the competition. We can notice the following: * the most votes were awarded in the case of Italy, from the public * Ukraine received a significant number of points, similar to ones that Switzerland received in the jury vote or France from both the jury and the public * in the case of some countries, we can notice significant disproportions from the votes given by the jury and public: Malta (low public votes vs. high jury votes), Finland (low jury votes vs. high public votes)

[66]:
```
# Group overall votes by UK juryvotes and televotes
votes = df.groupby(by=['United Kingdom Juryvotes','United Kingdom Televotes']).
 ↪sum()
votes.head()
```

[66]:

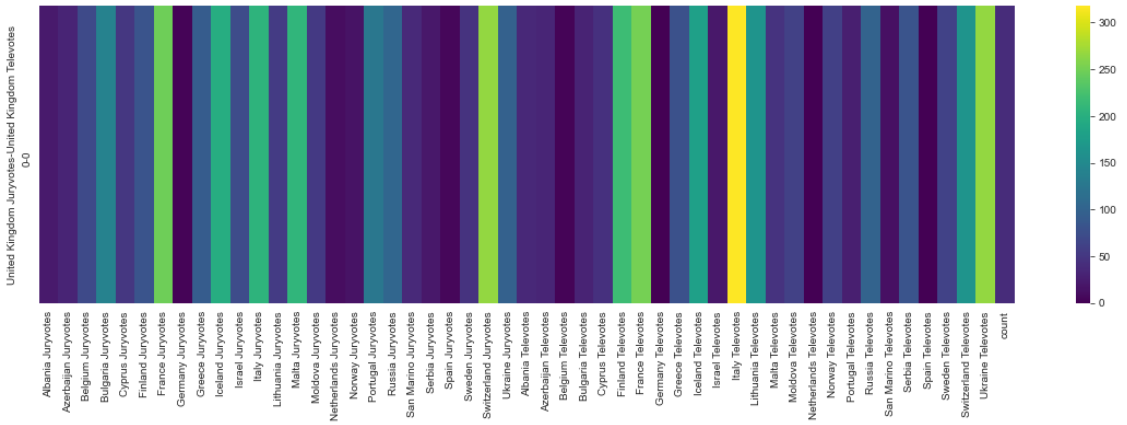
```
                                                 Albania Juryvotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                      22

                                                 Azerbaijan Juryvotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                         32

                                                 Belgium Juryvotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                      71

                                                 Bulgaria Juryvotes  \
United Kingdom Juryvotes United Kingdom Televotes
```

```
0                        0                                    140

                                                   Cyprus Juryvotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                     50

                                                   Finland Juryvotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                     83

                                                   France Juryvotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                    248

                                                   Germany Juryvotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                      3

                                                   Greece Juryvotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                     91

                                                   Iceland Juryvotes  …  \
United Kingdom Juryvotes United Kingdom Televotes                        …
0                        0                                    198  …

                                                   Norway Televotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                     60

                                                   Portugal Televotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                     27

                                                   Russia Televotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                    100

                                                   San Marino Televotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                     13

                                                   Serbia Televotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                     82

                                                   Spain Televotes  \
```

```
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                  0

                                                 Sweden Televotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                 63

                                                 Switzerland Televotes  \
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                     165

                                                 Ukraine Televotes  count
United Kingdom Juryvotes United Kingdom Televotes
0                        0                                 267     39

[1 rows x 51 columns]
```

[69]:
```
# Draw a heatmap showing overall televotes and juryvotes grouped by the null points of UK
plt.figure(figsize=(20,5))
sns.heatmap(votes,cmap='viridis')
```

[69]:
```
<AxesSubplot:ylabel='United Kingdom Juryvotes-United Kingdom Televotes'>
```

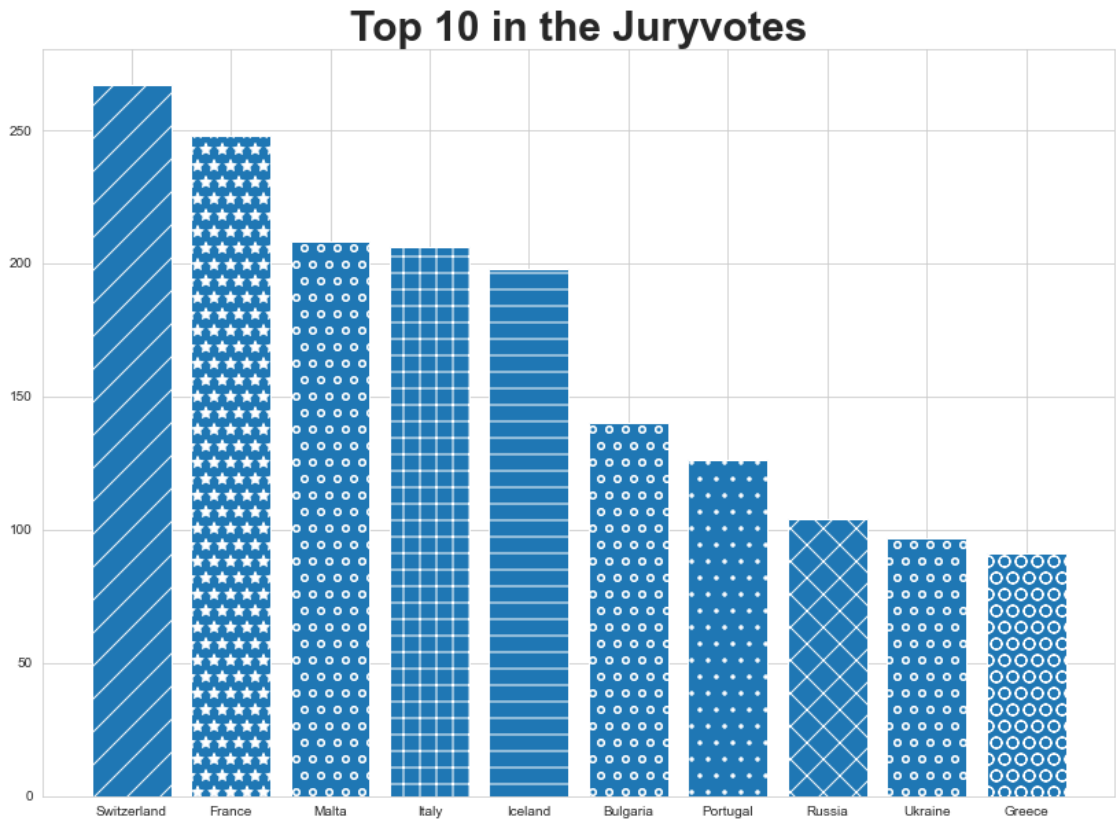


8 Analysis of the votes received by the Top 10 of the competition

I've created bar charts in order to point out the top 10 of the competition according to the jury, the public and overall.

[70]:
```
# Bar chart showing the top 10 according to the juryvotes
fig=plt.figure(figsize=(14,10))
```

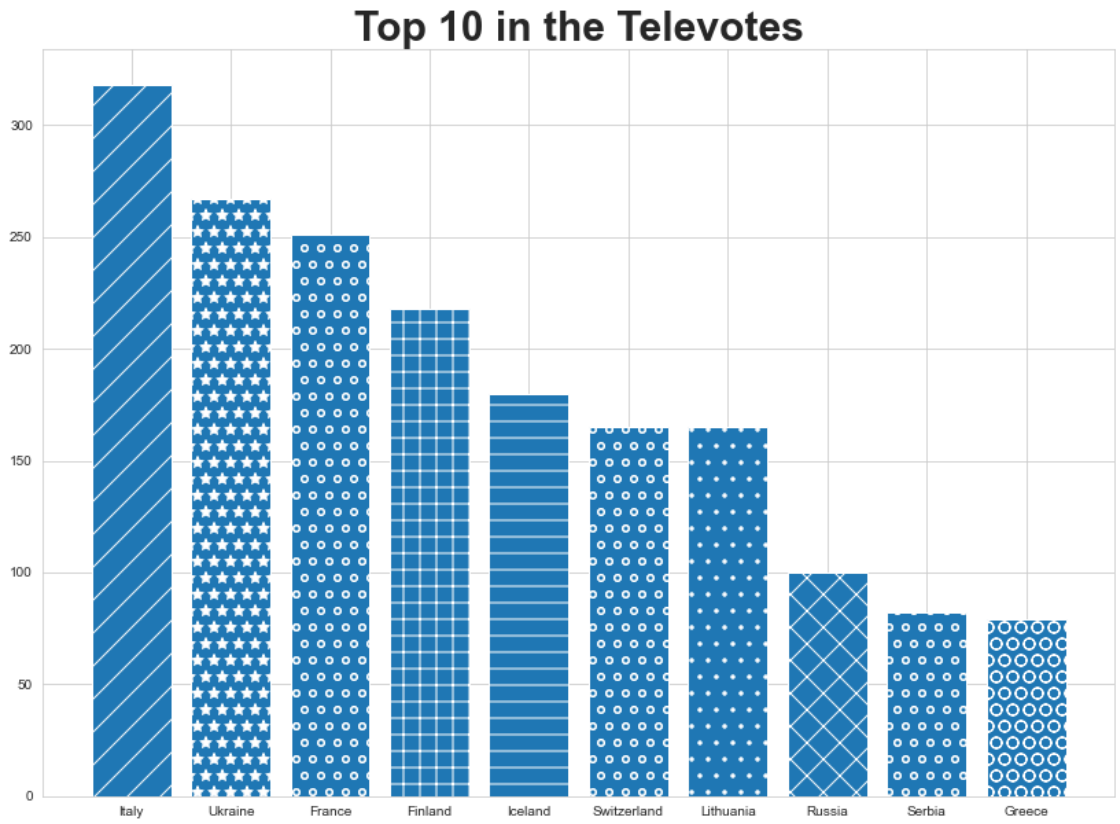
```
values = [df['Switzerland Juryvotes'].sum(),
          df['France Juryvotes'].sum(),
          df['Malta Juryvotes'].sum(),
          df['Italy Juryvotes'].sum(),
          df['Iceland Juryvotes'].sum(),
          df['Bulgaria Juryvotes'].sum(),
          df['Portugal Juryvotes'].sum(),
          df['Russia Juryvotes'].sum(),
          df['Ukraine Juryvotes'].sum(),
          df['Greece Juryvotes'].sum()]

labels = ['Switzerland',
          'France',
          'Malta',
          'Italy',
          'Iceland',
          'Bulgaria',
          'Portugal',
          'Russia',
          'Ukraine',
          'Greece']

bars = plt.bar(labels, values)

patterns = ['/', '*', 'o', '+', '-', 'o', '.', 'x', 'o', 'O']
for bar in bars:
    bar.set_hatch(patterns.pop(0))

plt.title('Top 10 in the Juryvotes',
         fontdict={'fontweight':'bold',
                   'fontsize': 30})

plt.show()
```



In the bar chart with the jury vote, we can notice the followings: * Switzerland won the jury votes, which is different than the overall winner of the competition, Italy * Malta is on the third position, which is significantly higher than the final result * Italy occupies only the forth position * Portugal and Bulgaria scored well in the jury vote, being in the top 10 * Ukraine made it to the top 10, but not in the top 3

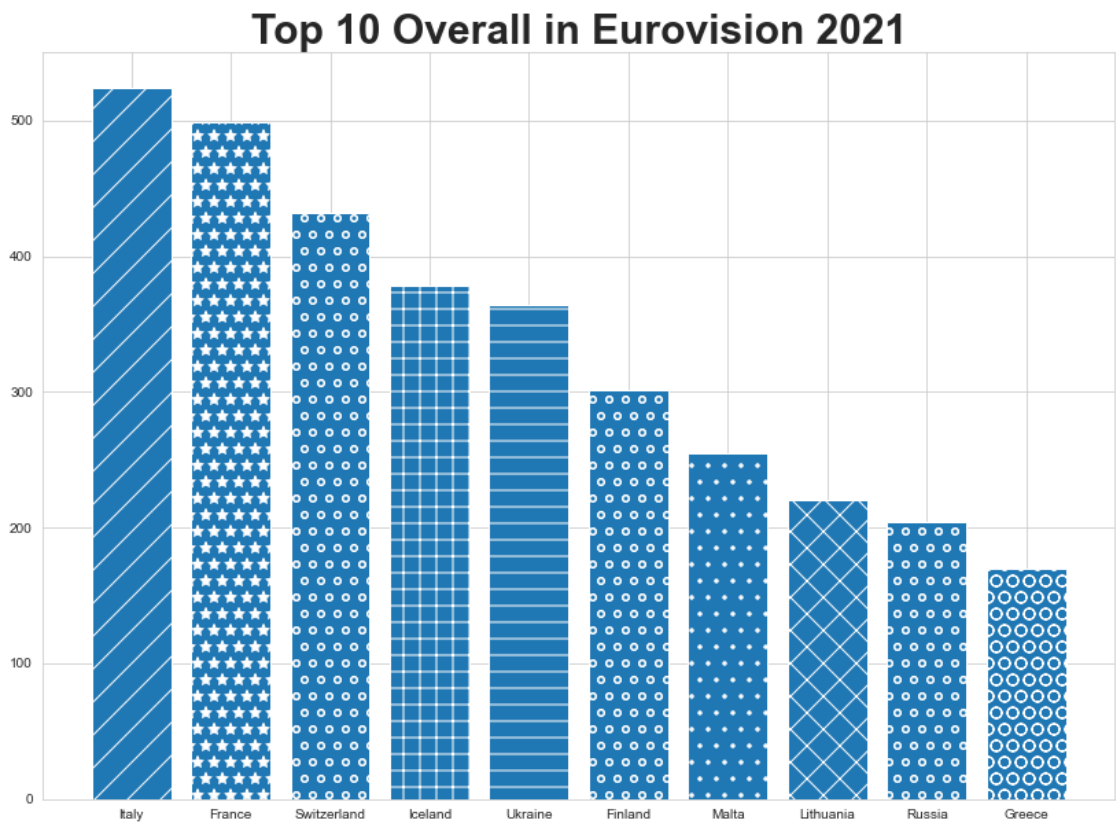
[71]:
```
# Bar chart showing the top 10 according to the televotes
fig=plt.figure(figsize=(14,10))

values = [df['Italy Televotes'].sum(),
          df['Ukraine Televotes'].sum(),
          df['France Televotes'].sum(),
          df['Finland Televotes'].sum(),
          df['Iceland Televotes'].sum(),
          df['Switzerland Televotes'].sum(),
          df['Lithuania Televotes'].sum(),
          df['Russia Televotes'].sum(),
          df['Serbia Televotes'].sum(),
          df['Greece Televotes'].sum()]

labels = ['Italy',
```

```
          'Ukraine',
          'France',
          'Finland',
          'Iceland',
          'Switzerland',
          'Lithuania',
          'Russia',
          'Serbia',
          'Greece']

bars = plt.bar(labels, values)

patterns = ['/', '*', 'o', '+', '-', 'o', '.', 'x', 'o', 'O']
for bar in bars:
    bar.set_hatch(patterns.pop(0))

plt.title('Top 10 in the Televotes',
         fontdict={'fontweight':'bold',
                   'fontsize': 30})

plt.show()
```



In the televotes, we can notice the followings: * Italy won, followed by Ukraine - in both cases this was a different podium than what the professional jury decided * Finland is in the top, though it did not score well in the jury vote * Switzerland did not score well in the public vote * Serbia made a surprising entry in the Top 10 and Lithuania made it as well, as opposed to the jury vote * Greece is on the 10th place in both cases

[72]:
```
# Bar chart showing top 10 overall, according to both the televotes and the␣
 ↪juryvotes
fig=plt.figure(figsize=(14,10))

values = [df['Italy Juryvotes'].sum() + df['Italy Televotes'].sum(),
          df['France Juryvotes'].sum() + df['France Televotes'].sum(),
          df['Switzerland Juryvotes'].sum() + df['Switzerland Televotes'].sum(),
          df['Iceland Juryvotes'].sum() + df['Iceland Televotes'].sum(),
          df['Ukraine Juryvotes'].sum() + df['Ukraine Televotes'].sum(),
          df['Finland Juryvotes'].sum() + df['Finland Televotes'].sum(),
          df['Malta Juryvotes'].sum() + df['Malta Televotes'].sum(),
          df['Lithuania Juryvotes'].sum() + df['Lithuania Televotes'].sum(),
          df['Russia Juryvotes'].sum() + df['Russia Televotes'].sum(),
          df['Greece Juryvotes'].sum() + df['Greece Televotes'].sum()]

labels = ['Italy',
          'France',
          'Switzerland',
          'Iceland',
          'Ukraine',
          'Finland',
          'Malta',
          'Lithuania',
          'Russia',
          'Greece']

bars = plt.bar(labels, values)

patterns = ['/', '*', 'o', '+', '-', 'o', '.', 'x', 'o', 'O']
for bar in bars:
    bar.set_hatch(patterns.pop(0))

plt.title('Top 10 Overall in Eurovision 2021',
         fontdict={'fontweight':'bold',
                   'fontsize': 30})

plt.show()
```



The final results from the competiton is showing that the public has had a major impact in the overall results. We see that: * Switzerland lost the first place awarded by the jury in favour of Italy * Ukraine, Lithuania and Finland are in the top 10, both being there thanks to the public vote * Malta lost the top 3 awarded by the jury

9 Analysis of the 12 points votes by geographical distribution

In order to make the analysis of the maximum 1 points awarded by the voters, I calculate first the longitude and the latitude, for further mapping them.

[74]:
```
# Finding the geolocation of countries for mapping them further
location = geolocator.geocode('Albania')
print((location.latitude, location.longitude))
```

```
(41.000028, 19.9999619)
```

[75]:
```
location = geolocator.geocode('Azerbaijan')
print((location.latitude, location.longitude))
```

```
(40.3936294, 47.7872508)
```

[76]:
```
location = geolocator.geocode('Belgium')
print((location.latitude, location.longitude))
```

```
(50.6402809, 4.6667145)
```

[77]:
```
location = geolocator.geocode('Bulgaria')
print((location.latitude, location.longitude))
```

```
(42.6073975, 25.4856617)
```

[78]:
```
location = geolocator.geocode('Cyprus')
print((location.latitude, location.longitude))
```

```
(34.9823018, 33.1451285)
```

[79]:
```
location = geolocator.geocode('Finland')
print((location.latitude, location.longitude))
```

```
(63.2467777, 25.9209164)
```

[80]:
```
location = geolocator.geocode('France')
print((location.latitude, location.longitude))
```

```
(46.603354, 1.8883335)
```

[81]:
```
location = geolocator.geocode('Germany')
print((location.latitude, location.longitude))
```

```
(51.0834196, 10.4234469)
```

[82]:
```
location = geolocator.geocode('Greece')
print((location.latitude, location.longitude))
```

```
(38.9953683, 21.9877132)
```

[108]:
```
location = geolocator.geocode('Iceland')
print((location.latitude, location.longitude))
```

```
(64.9841821, -18.1059013)
```

[84]:
```
location = geolocator.geocode('Israel')
print((location.latitude, location.longitude))
```

```
(31.5313113, 34.8667654)
```

[85]:
```
location = geolocator.geocode('Italy')
print((location.latitude, location.longitude))
```

```
(42.6384261, 12.674297)
```

[86]:
```
location = geolocator.geocode('Lithuania')
print((location.latitude, location.longitude))
```

```
(55.3500003, 23.7499997)
```

[87]:
```
location = geolocator.geocode('Malta')
print((location.latitude, location.longitude))
```

```
(35.8885993, 14.4476911)
```

[88]:
```
location = geolocator.geocode('Moldova')
print((location.latitude, location.longitude))
```

```
(47.2879608, 28.5670941)
```

[89]:
```
location = geolocator.geocode('Netherlands')
print((location.latitude, location.longitude))
```

```
(52.5001698, 5.7480821)
```

[90]:
```
location = geolocator.geocode('Norway')
print((location.latitude, location.longitude))
```

```
(60.5000209, 9.0999715)
```

[91]:
```
location = geolocator.geocode('Portugal')
print((location.latitude, location.longitude))
```

```
(40.0332629, -7.8896263)
```

[92]:
```
location = geolocator.geocode('Russia')
print((location.latitude, location.longitude))
```

```
(64.6863136, 97.7453061)
```

[93]:
```
location = geolocator.geocode('San Marino')
print((location.latitude, location.longitude))
```

```
(43.9458623, 12.458306)
```

[94]:
```
location = geolocator.geocode('Serbia')
print((location.latitude, location.longitude))
```

```
(44.1534121, 20.55144)
```

[95]:
```
location = geolocator.geocode('Spain')
print((location.latitude, location.longitude))
```

```
(39.3260685, -4.8379791)
```

[96]:
```
location = geolocator.geocode('Sweden')
print((location.latitude, location.longitude))
```

```
(59.6749712, 14.5208584)
```

[97]:
```
location = geolocator.geocode('Switzerland')
print((location.latitude, location.longitude))
```

```
(46.7985624, 8.2319736)
```

[98]:
```
location = geolocator.geocode('Ukraine')
print((location.latitude, location.longitude))
```

```
(49.4871968, 31.2718321)
```

[99]:
```
location = geolocator.geocode('United Kingdom')
print((location.latitude, location.longitude))
```

```
(54.7023545, -3.2765753)
```

[100]:
```
location = geolocator.geocode('Romania')
print((location.latitude, location.longitude))
```

```
(45.9852129, 24.6859225)
```

In the jury votes distribution, we can notice the following aspects: * Greece gave 12 points to Cyprus and Cyprus gave 12 points to Greece * Moldova received 12 points from Russia and from Bulgaria * Serbia received 12 points from North Macedonia

These observations are surprising particularly in the cases where the one awarded the 12 points is not on the top 10 overall or acording to the jury in general. It is the case of Moldova and Cyprus, both being voted by professional juries from countries which are geographically close to them. This raises a question regarding the political aspects and biases of the voting system from Eurovision. If the public can be influenced freely by various subjective factors and there is little control on that aspect, in the case of the jury there needs to be an objective view, coming from professionals.

Various analysis on Eurovision trends in voting are focused on the public vote, but in the case of the jury it is particularly interesting to notice that politics may play a factor in its decision, in some isolated cases which seem to stand out in general from the overall results.

[134]:
```
# Create a map and pin countries receiving 12 points, with popups and info␣
↪messages for each
m = folium.Map(location = [45.9852129, 24.6859225], tiles = 'Stamen Toner',␣
↪zoom_start=3)

# France
folium.Marker(location = [46.603354, 1.8883335],
              popup = 'France received 12 points from: Germany, Ireland,␣
↪Netherlands, San Marino, Serbia, Spain, Switzerland, United Kingdom',
              tooltip = 'Click for Jurivotes',
              icon = folium.Icon(color = 'blue')).add_to(m)

# Switzerland
folium.Marker(location = [46.7985624, 8.2319736],
```

```
            popup = 'Switzerland received 12 points from:  Albania, Belgium,␣
 ↪Denmark, Estonia, Finland, Iceland, Israel, Latvia',
            tooltip = 'Click for Jurivotes',
            icon = folium.Icon(color = 'blue')).add_to(m)

# Italy
folium.Marker(location = [42.6384261, 12.674297],
            popup = 'Italy received 12 points from: Croatia, Goergia,␣
 ↪Slovenia, Ukraine',
            tooltip = 'Click for Jurivotes',
            icon = folium.Icon(color = 'blue')).add_to(m)

# Malta
folium.Marker(location = [35.8885993, 14.4476911],
            popup = 'Malta received 12 points from: Australia, Norway,␣
 ↪Romania, Sweden',
            tooltip = 'Click for Jurivotes',
            icon = folium.Icon(color = 'blue')).add_to(m)

# Bulgaria
folium.Marker(location = [42.6073975, 25.4856617],
            popup = 'Bulgaria received 12 points from: Moldova, Portugal',
            tooltip = 'Click for Jurivotes',
            icon = folium.Icon(color = 'blue')).add_to(m)

# Greece
folium.Marker(location = [38.9953683, 21.9877132],
            popup = 'Greece received 12 points from: Cyprus, France',
            tooltip = 'Click for Jurivotes',
            icon = folium.Icon(color = 'blue')).add_to(m)

# Moldova
folium.Marker(location = [47.2879608, 28.5670941],
            popup = 'Moldova received 12 points from: Bulgaria, Russia',
            tooltip = 'Click for Jurivotes',
            icon = folium.Icon(color = 'blue')).add_to(m)

# Albania
folium.Marker(location = [41.000028, 19.9999619],
            popup = 'Albania received 12 points from: Malta',
            tooltip = 'Click for Jurivotes',
            icon = folium.Icon(color = 'blue')).add_to(m)

# Cyprus
folium.Marker(location = [34.9823018, 33.1451285],
            popup = 'Cyprus received 12 points from: Greece',
            tooltip = 'Click for Jurivotes',
```

```
              icon = folium.Icon(color = 'blue')).add_to(m)

# Iceland
folium.Marker(location = [64.9841821, -18.1059013],
              popup = 'Iceland received 12 points from: Austria',
              tooltip = 'Click for Jurivotes',
              icon = folium.Icon(color = 'blue')).add_to(m)

# Lithuania
folium.Marker(location = [55.3500003, 23.7499997],
              popup = 'Lithuania received 12 points from: Italy',
              tooltip = 'Click for Jurivotes',
              icon = folium.Icon(color = 'blue')).add_to(m)

# Portugal
folium.Marker(location = [40.0332629, -7.8896263],
              popup = 'Portugal received 12 points from: Czech Republic',
              tooltip = 'Click for Jurivotes',
              icon = folium.Icon(color = 'blue')).add_to(m)

# Russia
folium.Marker(location = [64.6863136, 97.7453061],
              popup = 'Russia received 12 points from: Azerbaijan',
              tooltip = 'Click for Jurivotes',
              icon = folium.Icon(color = 'blue')).add_to(m)

# San Marino
folium.Marker(location = [43.9458623, 12.458306],
              popup = 'San Marino received 12 points from: Poland',
              tooltip = 'Click for Jurivotes',
              icon = folium.Icon(color = 'blue')).add_to(m)

# Serbia
folium.Marker(location = [44.1534121, 20.55144],
              popup = 'Serbia received 12 points from: North Macedonia',
              tooltip = 'Click for Jurivotes',
              icon = folium.Icon(color = 'blue')).add_to(m)

# Ukraine
folium.Marker(location = [49.4871968, 31.2718321],
              popup = 'Ukraine received 12 points from: Lithuania',
              tooltip = 'Click for Jurivotes',
              icon = folium.Icon(color = 'blue')).add_to(m)

m
```

[134]: <folium.folium.Map at 0x7fefd5616550>

In the televoting, we can notice the follwing aspects: * Lithuania was voted by Latvia * Serbia received 12 points from North Macedonia, the same as in the case of the jury vote * Moldova was voted by Romania. However, Romania gave 12 points to Malta in the jury vote * Russia gave 12 points to Moldova as well as in the jury vote * Finland received 12 points from Estonia, Iceland and Sweden * Iceland received 12 points from Denmark and Finland * Cyprus received 12 points from Greece and Greece received 12 points from Cyprus

These votes received by the countries from the public of a neighbour state can have various roots, considering that there is little control or expectations in the public capacity to evaluate a performance. One one side, personal taste in music is in itself subjective and on the other, so do voting for a neighbour country can be subjective. It can be due to political reasons, exposure of the artist in that area, a big fan base located in a particular country, significant number of immigration happening in a certain country. These reasons can be all explored in further analysis. For now, what is particularly suspect is the votes received by Moldova from Russia and by Cyprus from Greece. Greece was in the top 10 in both the jury and the public vote, so this is why it is not mentioned.

[112]:
```
# Create a map and pin countries receiving 12 points, with popups and info␣
 ↪messages for each
m = folium.Map(location = [45.9852129, 24.6859225], tiles = 'Stamen Toner',␣
 ↪zoom_start=3)

# Italy
folium.Marker(location = [42.6384261, 12.674297],
              popup = 'Italy received 12 points from: Bulgaria, Malta, San␣
 ↪Marino, Serbia, Ukraine',
              tooltip = 'Click for Televotes',
              icon = folium.Icon(color = 'red')).add_to(m)

# Lithuania
folium.Marker(location = [55.3500003, 23.7499997],
              popup = 'Lithuania received 12 points from: Germany, Ireland,␣
 ↪Latvia, Norway, United Kingdom',
              tooltip = 'Click for Televotes',
              icon = folium.Icon(color = 'red')).add_to(m)

# Serbia
folium.Marker(location = [44.1534121, 20.55144],
              popup = 'Serbia received 12 points from: Austria, Croatia, North␣
 ↪Macedonia, Slovenia, Switzerland',
              tooltip = 'Click for Televotes',
              icon = folium.Icon(color = 'red')).add_to(m)

# Ukraine
folium.Marker(location = [49.4871968, 31.2718321],
              popup = 'Ukraine received 12 points from: France, Israel, Italy,␣
 ↪Lithuania, Poland',
              tooltip = 'Click for Televotes',
```

```
                icon = folium.Icon(color = 'red')).add_to(m)

# France
folium.Marker(location = [46.603354, 1.8883335],
                popup = 'France received 12 points from: Belgium, Netherlands,␣
 ↪Portugal, Spain',
                tooltip = 'Click for Televotes',
                icon = folium.Icon(color = 'red')).add_to(m)

# Finland
folium.Marker(location = [63.2467777, 25.9209164],
                popup = 'Finland received 12 points from: Estonia, Iceland,␣
 ↪Sweden',
                tooltip = 'Click for Televotes',
                icon = folium.Icon(color = 'red')).add_to(m)

# Iceland
folium.Marker(location = [64.9841821, -18.1059013],
                popup = 'Iceland received 12 points from: Australia, Denmark,␣
 ↪Finland',
                tooltip = 'Click for Televotes',
                icon = folium.Icon(color = 'red')).add_to(m)

# Cyprus
folium.Marker(location = [34.9823018, 33.1451285],
                popup = 'Cyprus received 12 points from: Greece, Russia',
                tooltip = 'Click for Televotes',
                icon = folium.Icon(color = 'red')).add_to(m)

# Greece
folium.Marker(location = [38.9953683, 21.9877132],
                popup = 'Greece received 12 points from: Cyprus, Goergia',
                tooltip = 'Click for Televotes',
                icon = folium.Icon(color = 'red')).add_to(m)

# Moldova
folium.Marker(location = [47.2879608, 28.5670941],
                popup = 'Moldova received 12 points from: Czech Republic, Russia',
                tooltip = 'Click for Televotes',
                icon = folium.Icon(color = 'red')).add_to(m)

# Israel
folium.Marker(location = [31.5313113, 34.8667654],
                popup = 'Israel received 12 points from: Azerbaijan',
                tooltip = 'Click for Televotes',
                icon = folium.Icon(color = 'red')).add_to(m)
```

```
# Russia
folium.Marker(location = [64.6863136, 97.7453061],
              popup = 'Russia received 12 points from: Moldova',
              tooltip = 'Click for Televotes',
              icon = folium.Icon(color = 'red')).add_to(m)

# Switzerland
folium.Marker(location = [46.7985624, 8.2319736],
              popup = 'Switzerland received 12 points from: Albania',
              tooltip = 'Click for Televotes',
              icon = folium.Icon(color = 'red')).add_to(m)

m
```

[112]: <folium.folium.Map at 0x7fefd0d53d60>

10 Conclusions and possible future analysis explorations

The conclusions of this project will be done from the perspective of the initial questions to answer in this analysis.

Regarding the correlation between the jury vote and the televote regarding choosing the winner of Eurovision 2021, we can draw the following conclusions: * Italy has won with a significant proportion of votes coming from the public. * Except Ukraine, other countries had a different perspective on who should be the winner of the competition, either in the jury vote or the televote. Ukraine was the only country that positioned Italy in the first place, both by the jury and the televote. * The second most votes of one kind that Italy received was 10 zeros in the jury vote, which is surprising considering that it has won the competition. * The most votes of one kind that Italy received was 13 of 10 in the televote, which means that even if the public had selected the overall winner in a significant proportion, it did not see it as the winner, but in second place. The sum of them turned out to be quite significant and overall it contributed to Italy winning the competition. * Except from UK and Netherlands which had an alignment of views between the public and the jury, giving 0 points or slightly more, the other countries awarding 0 points in the jury vote have had a large difference in the public votes, this going to as far as Malta's case with 0 points from the jury versus 12 points from the public. * The jury was particularly harsh to Italy, as opposed to the same countries in the public vote who placed Italy among the firsts. * The null points received by the UK, though they show an 100% alignment between the jury and the public vote, do not represent a clear evaluation of the alignment of views between the public and the jury. In order to conclude this, we would need to see what points would the jury and the public award to the UK, on a different scale, in another analysis, separate from this present one.

From analysing the overall results from the competition, we can conclude the following aspects regarding the jury vote and the televote correlation: * Italy was not the winner according to the professional jury, but Switzerland, who received the highest score in the jury vote, was on the 6th place in the televote. * France was equilibrated in the jury and the public views, both seeing it either on the second or third place. * Italy was only in fourth place in the jury vote, with results close to Iceland and Malta. * Germany, Netherlands and Spain have received low scores (0 or close to 0) both in the jury and the professional vote, similar to the UK. This means that there was a

very low difference between the jury and the televote in picking the last 4 of the competition. * The most votes received in the competition are the televotes that Italy received, which has placed it in the end in first place. * Ukraine and Finland scored particularly well in the public vote, as opposed to the jury vote which did not see them on the top.

From analysing the correlation from the jury and the televotes in relation to picking the top 10 of the competition, we can conclude the followings: * Portugal and Bulgaria scored well only in the jury vote, as opposed to the public who did not see them in the top 10. * Malta was in third place according to the jury vote, as opposed to the public vote who did not see it finishing in the top 10. * Greece was on the 10th place both in the jury and the televote. * Overall, the jury and the televote had a different view on who should be in the top 3, but the majority of the countries were both present in the top 10 from the jury and the televote, though on different positions: Iceland, Italy, France, Switzerland, Russia, Ukraine, Greece

After the analysis done on the 12 points awarded to countries based on geographical position, we can conclude the followings: * There is a similarity between the voting done in the public and the jury vote about the tendency to vote some countries which overall are not in the top 10 of the competition, but are geographically close. This is the case of Cyprus receiving 12 points from Greece both in the jury and the televote. Greece receiving 12 points from Cyprus in both the votings was overall excluded, because Greece was on the top 10 overall in the competition and was voted also by other countries. Moldova receiving 12 points from Russia both in the public and the televote is also an example. * The case of Romania giving 12 points to Moldova in the public vote is also something to point out. However, Romania awarded 12 points from the jury to Malta, which overall had great jury points from other countries as well. * Finland received 12 points in the public vote from Estonia, Iceland and Sweden, all geographically close. Also, Iceland received 12 points from the public vote from Denmark and Finland.

It would be interesting to explore further what is at the base of these 12 points awarded to countries that are geographically closer to each other. Particularly, I would be interested to understand why the professional juries voted this way, considering their expertise. * Extending the 12 points analysis to a full analysis of all votes from the jury and the public. The 12 points was just a surface and a right beginning for this analysis. * We can also make a parallel between the immigration statistics and the public vote. Some countries can receive votes from their citizens who live abroad and send the vote from the new country. * We could make an analysis on the popularity of the artists in certain areas / countries and the public vote they received. There is the potential that international fans have voted for some artists solely based on their affiliation. * Extending the analysis of the top 10 to the entire competition from Eurovision would be interesting to explore in order to point out the correlation between the voting in the televote and the public vote. The analysis could also be extended to the semi-finals and also to previous years.

11 Bibliography

- https://en.wikipedia.org/wiki/Eurovision_Song_Contest_2021 , last checked on 01/07/2021
- https://en.wikipedia.org/robots.txt , last checked on 01/07/2021
- 'Eurovision 2021 song competition voting patterns', by Prajit T.R, https://www.kaggle.com/prajittr/eurovision-2021-song-competition-voting-patterns , last checked on 01/07/2021
- 'Neighbour Pattern In The Votes By Countries', by Ashesh Jain, https://www.kaggle.com/asheshtadaiya/neighbour-pattern-in-the-votes-by-countries ,

last checked on 01/07/2021

- https://eurovision.tv/event/rotterdam-2021 , last checked on 01/07/2021
- https://creativecommons.org/publicdomain/zero/1.0/ , last checked on 01/07/2021

[ ]: