

Googling Eurovision

Modeling Voting in the Eurovision Song Contest Using Google PageRank



Michael Gordon
Box 5638
Davidson College
MAT210 - Mathematical Modeling
Dr. Chartier
May 3, 2007

Abstract

In the Eurovision Song Contest, each competing European country submits one song, and each other country assigns its top ten songs a set of points, voting for its favorite songs using a modified Borda count. This paper describes the creation of a model using a modified version of Google's PageRank algorithm, a Markov chain model of a network, which is explained and demonstrated. The algorithm is changed by making the weights of the edges in the network the number of votes each country in the contest gives to another. The 2005 and 2006 Eurovision Song Contests are recreated with the model, and the proper parameters for the model are found experimentally by comparing results to past contests. The model is then used to predict the winner of Eurovision Song Contest 2007 Helsinki, which will take place on May 12.

Eurovision Song Contest logo from <http://upload.wikimedia.org/wikipedia/en/3/3a/Eurovision_Song_Contest_logo.svg>

Contents

1	What is the Eurovision Song Contest?	1
2	Model Development	1
2.1	Mathematics of Eurovision	1
2.2	What to Consider	2
2.3	Simplifying Assumptions	3
2.4	Data	3
2.5	PageRank	4
2.6	Using PageRank to Model the Eurovision Song Contest	7
3	Results	7
3.1	ESC 2005 Kiev and 2006 Athens: Playing With Parameters	7
3.2	ESC 2007 Helsinki: Who Will Win?	10
3.3	Flaws	12
4	Conclusion and Future Work	13
	Appendices	14
A	Representative Matlab Code	15
B	Representative Data: Language-Preference Voting Averages	19

“You gotta turn it on, you gotta put it out, you gotta be sure that it’s something everybody’s gonna talk about. Before you decide, the time’s arrived for making your mind up.”
–from “Making Your Mind Up” by Bucks Fizz, winner of Eurovision Song Contest 1981 Dublin

1 What is the Eurovision Song Contest?

In 1954, members of the European Broadcasting Union (EBU) created an international broadcasting network with various European countries’ broadcasters as its members. During a time when few Europeans owned TVs and little money existed in recent post-World War II Europe to create the national television broadcasting infrastructure that existed for radio, the national broadcasters of eight European countries came together to create a unified European television network, which was later called “Eurovision.”[1] Following the success of its first season of programming, Marcel Benzençon created a song contest for all of Europe, called “Le Grand-Prix Eurovision de la Chanson Européenne.”[2] The first Eurovision Song Contest, as it is now called¹, was held in Lugano, Switzerland on May 24, 1956 with seven countries and fourteen songs.[3] After 1956, each country was only allowed one entry.

The premise of the contest is to select the best song from Europe in that year. Each participating country casts votes for other countries’ songs and the song garnering the most votes is crowned the winner of the Eurovision Song Contest. The method for a country to vote, the way that the number of votes is determined, and the number of votes that can be cast varies by year, and will be explored more thoroughly in Section 2.1 below.

Today, the ESC is still an annual trans-European event, having been held every year since 1956. In 2007, the contest will be held in Helsinki, Finland on May 12. In a stark contrast from the first competition, 42 countries will compete in ESC Helsinki 2007. Twenty-four of these are will compete in the final round. Of the final round competitors, 14 are guaranteed and the remaining 18 will compete in the semi-final round on May 10, with the top 10 advancing to the final. [4]

2 Model Development

2.1 Mathematics of Eurovision

In 1975, the administrators of Eurovision created a new voting system for Eurovision based on the mathematical Borda count. Each country would create a list of its top ten songs. It would give its favorite song 12 points, its second favorite 10, third favorite 8, fourth favorite 7, and so on down to its tenth favorite receiving 1 point.[5] Thus, each country casts a total of 58 points in any given contest.

Before 2004, only countries qualifying for the final round (the methods of qualifying vary by year) could vote in the final. In 2004, the contest featured a semi-final round for the first time, and all competing countries could vote in the semi-final and final. Since the number of participating countries changes from year to year, the maximum number of point that a country could receive changes each year. In 2007, the highest possible score is $12 \times 41 = 492$.

¹As this paper will address only the contest and not the Eurovision network, the Eurovision Song Contest will often be shortened to “Eurovision” or “ESC.”

In recent years, votes have been determined by the residents of the country by a process known as televoting. The residents of each participating country cast votes for their one favorite songs by calling designated phone numbers. Televoting was implemented during 1997 and 1998. Before televoting, a country's votes were determined by a national jury. The change from a jury to televoting is can be particularly noticable in the way the country votes.[6] This phenomenon will be further examined in Section 3.3.

Eurovision participants are typically added only a few every year, if any, with the largest expansion coming shortly after the collaps of the Soviet Union and the fall of the Iron Curtain.[7] This year's contest in Helsinki will see the entry of two new countries.² Since the participants in Eurovision do not change drastically from year to year, it is easy to keep track of the votes from one country to another and perform simple statistical analyses on these vote totals.

2.2 What to Consider

After examining the background of the Eurovision Song Contest, I will now turn to modeling the contest.

While the tension of competition is certainly one of the most exciting parts of Eurovision, it is also interesting to attempt to predict the winners. The Eurovision fan site Oikotimes.com collects the results of numerous Eurovision polls across the Internet to attempt to predict the winner. British betting company Ladbrokes conducts betting on the winner of the contest. Beyond the Eurovision fan community, about 120 million people watched the contest in 2005.[6] Clearly, the contest interests a large number of people in Europe and around the world, including the author.

Some academics have previously examined the contest, looking for factors that influence Eurovision voting. In 2006, Laura Spierdijk and Michel Vellekoop took earlier studies along with a statistical analysis of past contests as well as the socio-cultural and geographic makeup of the competitors. They found a strong influence on the voting from geography – countries vote for other countries near them. Similarly, Derek Gatherer published a paper in the *Journal of Artificial Societies and Social Simulation* in which he used statistical analysis and Monte Carlo simulation to model the results of various Eurovision Song Contests.[5] He found distinct “collusive voting groups,” or groups which often give high numbers of votes to each other. Spierdijk and Vellekoop also looked at religion, ethnicity, and language as possible factors. Given these papers, one of which even uses the basic mathematical modeling technique of simulation, it does not seem unreasonable at all to attempt to model the Eurovision Song Contest.

Voting in the Eurovision Song Contest is a discrete phenomenon. As Fenn, et. al. put it, “each voting country A allocates a set of points $\{1, 2, 3, 4, 5, 6, 7, 8, 10, 12\}$ to the ten other countries $\{B, C, D, E, F, G, H, I, J, K\}$ which are a subset of the entire set S of competing countries.”[8] In other words, the number of votes assigned to each country is discrete. Then the scoreboard is determined by a series of votes, and thus the voting can be modeled discretely.

To model the contest, I chose to consider two major factors: patterns in country-to-country voting and patterns in language-based voting. The model combines Monte Carlo simulation with a modification of Google's PageRank algorithm. For the first factor, country-to-country voting, the premise is that, since the modified Borda count method of voting has been used for over thirty years, there is sufficient data to predict the number of points that one country will give to another.

²The Czech Republic and Georgia are scheduled to compete for the first time this year. Serbia and Montenegro will each have one entry; in the past, they have participated jointly.

In fact, I used basic statistics to model the number of points that each country will give to another. The second factor, language-based voting, is based on the idea that a country is more likely to vote for songs in a particular language. Until 1999, countries were required to perform in one of their official languages[9], so the language-based voting may be similar to geographic voting. However, language is still an aspect of culture and some countries may have multiple official languages, so the choice of language may still be significant.

2.3 Simplifying Assumptions

In any model of a real-world phenomenon, simplifying assumptions must be made in order to allow for the phenomenon to be modeled using mathematical techniques. The most basic assumption, and the one that makes the model unlikely to predict the winner, is that *past voting tendencies and language are the only factors influencing countries' votes*. Second, I assume that each country will follow those past voting tendencies, both in terms of geography and language. For example, if Cyprus were to be successfully reunified tomorrow and all animosity between Turks and Greek Cypriots erased, I would still assume that the Cyprus and Turkey would be unlikely to exchange large numbers of points. Along the same lines, the model assumes that no changes in preferences have occurred over the period from which the data comes. If there were any change in culture, it would not be detected, and the overall averages would not be detected. If two countries were at odds, culturally or politically, from 1975 to 1999, but since have been allies, the low scores from the previous, longer period would weigh strongly in the averages.

The model assumes also that all countries compete in the final round. I do not simulate the semi-final round. This is, in some ways, realistic, as qualifiers from the semi-finals have just as much chance to win as do automatic qualifiers and even countries that do not qualify for the final round vote in the finals. Similarly, the model assumes that there are no new entrants in the 2007 contest. Since no past voting data exists on these countries, their voting patterns cannot be predicted in the same method as all other countries. Additionally, no other countries have voted for the new entrants, so the number of votes they will receive from other countries can only be predicted by the language factor. This will result in a very small number of votes for the country comparatively in the model, so even if these countries were included, the results would be essentially useless.

In terms of modeling the contest, the model assumes that each language is in only one language, whichever language in which most of the song is written. This is often not the case in actuality. Ruslana's winning entry for Ukraine in 2004, "Wild Dances," was in both English and Ukrainian.[10] The inclusion of another language may have appealed to a larger audience and helped her win the contest. To model voting, however, this simplification was made. It should be noted that until 1999, all songs were required to be in an official language of the country submitting the song. This model assumes that all countries have always been free to submit a song in any language.

I will assume that the averages of the data is distributed normally. This assumption is justified by the Central Limit Theorem, which states that as the number of data points approaches infinity, the average of the data points will approach the normal distribution.

2.4 Data

In order to model voting in the ESC using the method briefly described above, data on country-to-country voting and language-based voting are required. The country-to-country voting data was

graciously provided by Jarmo Pentillä of ESCStats.com.³ Pentillä’s data contains the voting record for each Eurovision Song Contest since 1975, when the current voting system was implemented. Both semi-final and final rounds are included in voting data. Each year has essentially a score sheet, All of these data points are averaged and the standard deviations found using Microsoft Excel. Of interest is that Serbia and Montenegro has the highest average number of votes with 221, though the country has only participated three times. This small number of data points leads to a very high standard deviation of 72.746 votes. Additionally, some countries have never competed in the contest together. For example, Morocco competed only once (1980), and thus have no contests in common with many countries. All of these averages, which would otherwise result in a “divide by zero” error, were set to be zero. I used all the data when modeling past contests, including data from that contest, in order to maximize the set. Finally, as stated above, I assume that the averages are distributed normally, so that a random number can be drawn from the normal distribution with the calculated standard deviation around the calculated mean.

The data for language-based voting was gathered from Pentillä’s spreadsheets and from The Diggiloo Thrush, [7], a Website containing information about each Eurovision entry. I compiled a list of each country’s song’s languages from 1996–2006, then calculated the total number of points each country gave to each language or group of languages in the final round.⁴ (Votes in the semi-final were not considered.) If a song was in multiple languages, then the number of points given to that song by the country being considered was given to both language groups. For example, The Ukrainian entry in 2004, “Wild Dances” by Ruslana, was in both English and Ukrainian and Estonia gave 12 points to Ukraine that year. The vote count for both English and Slavic for Estonia in 2004 would be incremented by 12. Thus, it is possible for each country to give more than 58 votes in each year. I found the averages and standard deviations of the data, assuming that the averages of data are normally distributed. The most interesting fact about the data is the extreme favor of songs in English, with a mean of 36.305 votes to songs in English each year, versus the next highest language group, Slavic, with an average of 8.63 votes to songs in Slavic languages each year. Since each country can give a song at most 12 points, this means that, if language-based voting is given equal standing to country-to-country voting history, this means that two countries would give that song full points! Clearly this will require correction.

2.5 PageRank

The following section of this paper will explain Google’s PageRank algorithm, which was used to model voting in the Eurovision Song Contest. Section 2.5.1 will discuss the pure Google PageRank algorithm, and Section 2.5.2 will explain the modifications made to the algorithm to model Eurovision voting.

2.5.1 Google’s PageRank

The world-famous search engine Google uses a mathematical method to determine which Webpages are ranked higher than others. Google’s spiders crawl the Web to build an adjacency matrix de-

³The author would like to extend a further thanks for Pentillä for his assistance with comprehension of the spreadsheets, as well as the other useful data compiled on ESCStats.com.

⁴The languages and groups of languages used were English, Spanish, Italian, Portuguese, German, Scandinavian (Danish, Norwegian, Finnish, Swedish), Slavic (Russian, Ukrainian, Macedonian, Croatian, Polish, and Albanian), Turkish, Hebrew, and Other.

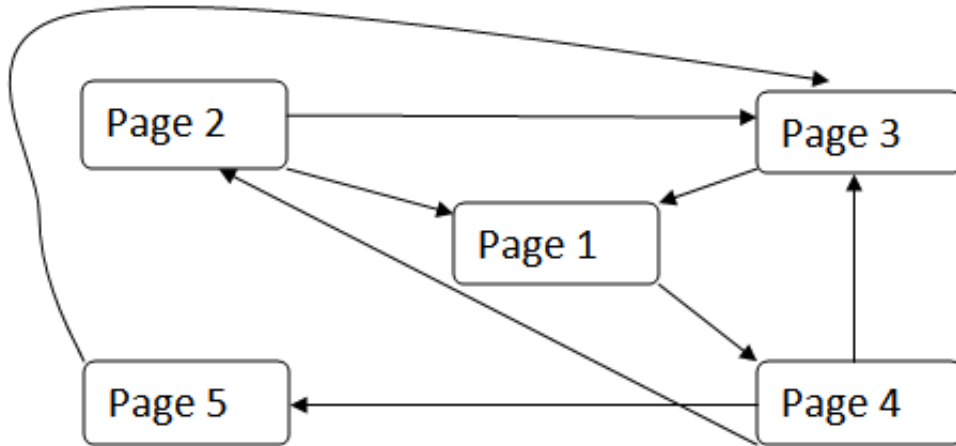


Figure 1: Example Five-Page Website Network. Adapted from [12].

noting which pages link to which other ones. The PageRank algorithm then modifies the adjacency matrix slightly and finds the left eigenvector by the power method. The method is detailed in the example below.

Suppose we have the 5-page Webpage network shown in Figure 1, where each arrow represents a link from one Webpage to another⁵. This network graph can be represented as the adjacency matrix:

$$\mathbf{G} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Let r_i be the sum of the entries in row i of \mathbf{G} . Then the probability that a visitor to the page represented by row i will next visit the page represented by column j is in the stochastic matrix \mathbf{H} , where $H_{i,j} = \frac{G_{i,j}}{r_i}$, below:

$$\mathbf{H} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 1/2 & 0 & 1/2 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & 1/3 & 0 & 1/3 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

However, observe that currently, the only way to visit a page is to follow a link to it. For example, for a user who starts at Page 1 to reach Page 3, he/she must first travel through Page 4. To resolve this problem, PageRank creators Sergey Brin and Lawrence Page stated that the probability p that a user would follow a link on a page was 0.85. The other 15% of the time, the browsing user will “teleport” to a page on the Internet, or, in this example, one of the 5 pages. The user may teleport

⁵The example and explanation of the PageRank method was partially taken from [12] and [11].

back to the page he/she just visited. With this in mind, we can form the transition matrix \mathbf{M} by taking:

$$M_{i,j} = p \left(\frac{G_{i,j}}{r_i} \right) + \frac{1-p}{n} \quad (1)$$

Thus, we obtain:

$$\mathbf{M} = \begin{pmatrix} .0300 & .0300 & .0300 & .8800 & .0300 \\ .4550 & .0300 & .4550 & .0300 & .0300 \\ .8800 & .0300 & .0300 & .0300 & .0300 \\ .0300 & .3133 & .3133 & .0300 & .3133 \\ .0300 & .0300 & .8800 & .0300 & .0300 \end{pmatrix}$$

The PageRank of page i is defined to be the normalized steady-state vector of the Markov process with initial vector \mathbf{v}_0 with each entry equal to $1/n$, where n is the number of Webpages in the network. The Perron-Frobenius Theorem guarantees convergence of this process. The steady-state vector is the dominant eigenvector, which in this case, will always have the eigenvalue of 1. We can then use the power method, which is that $v_k = v_0 \mathbf{M}^k$, repeatedly to find a vector to which the method converges. Using this method, the PageRank of this network is:

$$\boldsymbol{\pi} = \begin{pmatrix} 0.2841 \\ 0.1935 \\ 0.1507 \\ 0.2990 \\ 0.0727 \end{pmatrix}$$

The i th entry in $\boldsymbol{\pi}$ is the PageRank of page i . Observe in $\boldsymbol{\pi}$ that the largest entry is the fourth entry and the second largest is the first. This means that page 4 has the highest PageRank, followed by page 1.

2.5.2 Modifying PageRank

In order to model voting in the Eurovision Song Contest, I modified the Google PageRank algorithm in a manner similar to Govan and Meyer in [11], in which the authors use PageRank to predict the winners of National Football League games. They state that when a team loses a game by x points, that team “votes” for the team to which it lost by x points. We can then construct a weighted digraph of the football season, with the edges going from each losing team to each winning team and the weight of each edge being the margin of victory. The weighted digraph can be turned into an adjacency matrix \mathbf{G} , where each non-zero entry $G_{i,j}$ is the weight of the edge from Team i to Team j . Additionally, there is a probability that a team will lose to another team in some game. In order to allow for this possibility, we use the Formula 1 to create a transition matrix, where p is the probability that a team will follow the graph precisely. For example, if Chicago defeats Denver by 10 points and Denver defeats Carolina by 5 points, with probability p , Chicago will defeat Carolina. Finally, we find a “personalization vector” \mathbf{v} , which may represent some statistic, such as defensive effectiveness, or be as basic as each entry being $1/n$. The left eigenvector corresponding to the dominant eigenvalue of 1 will be the steady-state solution to this Markov chain, and will be the PageRank vector $\boldsymbol{\pi}$. The i th entry in $\boldsymbol{\pi}$ is the PageRank of the team whose record is represented by the i th row and column in G .

2.6 Using PageRank to Model the Eurovision Song Contest

With this model of the National Football League in mind, I created a model of the Eurovision Song Contest. The model uses PageRank to simulate the ESC and was programmed in MATLAB. A separate simulation was programmed for each year simulated. The program creates matrices of averages of the each participating country's past voting tendencies, both country-to-country voting, called \mathbf{V} , and language preference voting, called \mathbf{L} , then randomizes these tendencies around its calculated standard deviation assuming that it is normal. As stated in Section 2.4, the votes for songs in English is extremely large, partially because most songs since the rule change in 1998 have been in English.[7] In fact, if a country which sang in English were to be given the full number of average points from a country to a song in English, these votes would outweigh the voting power of two countries. To resolve this issue, the additional votes given from countries to songs in English is scaled by a given English Scaling Factor, which will be experimentally determined by comparing simulated results with known results of past contests. Language preference voting may be more or less of a factor than country-to-country voting, so \mathbf{L} is scaled by an experimentally-determined Language Voting Factor. A final voting matrix \mathbf{V} is formed by adding \mathbf{L} to the current \mathbf{V} . Then the PageRank vector is determined using modified MATLAB code written by Cleve Moler available at <http://www.mathworks.com/moler/ncm/pagerank.m>. The final ranking of the countries is output and, for past contests, compared to the actual results using Kendall's Tau rank correlation test. Kendall's Tau is essentially a normalized measure of how many swaps are needed to turn the experimental list into the actual list using bubble sort. More specifically, it is defined in terms of concordant and discordant pairs of data points. A pair of data points $(x_i, y_i), (x_j, y_j)$ from lists x and y is concordant if either $x_i \geq x_j$ and $y_i \geq y_j$ or $x_i \leq x_j$ and $y_i \leq y_j$ and discordant otherwise. Kendall's Tau, or τ , can now be defined as:

$$\tau = \frac{\text{concordant} - \text{discordant}}{\sqrt{\text{concordant} + \text{discordant} + \text{extra-}y} + \sqrt{\text{concordant} + \text{discordant} + \text{extra-}x}} \quad (2)$$

Thus, the value ranges between -1 and 1, and values closest to 1 are the best. Kendall's Tau is implemented in this simulation using an adapted version of the code in [13]. (The code is included in the Appendix A.) After each round of simulations, the average Kendall's Tau and most common first place team is printed. The algorithm is summarized in Algorithm 2.6.

The simulation is a MATLAB function with four parameters. First is the number of pseudo-random simulations to perform. The second parameter is the EuroFollowing Factor. Defined as p in the sections above about Google PageRank, the EuroFollowing Factor is the probability that a country will follow its previous voting tendencies. The last two parameters are the English Scaling Factor and the Language Voting Factor.

3 Results

3.1 ESC 2005 Kiev and 2006 Athens: Playing With Parameters

In order to be able to accurately simulate future contest, we first need to determine the value of the parameters necessary to most accurately simulate Eurovision voting. Therefore, the objective of running simulations of past contests is to find the correct parameters to simulate future contests, such as Eurovision Song Contest 2007 Helsinki.

Algorithm 1 Algorithm for the Eurovision Song Contest model

- Load country-to-country voting data into \mathbf{V}
 - Load language-preference voting data into \mathbf{L}
 - Randomize entries of \mathbf{V} and \mathbf{L} using their means and standard deviations
 - Scale the entries of \mathbf{L} corresponding to votes for songs in English by ESF
 - Scale all entries of \mathbf{L} by LVF
 - $\mathbf{V} = \mathbf{V} + \mathbf{L}$
 - Perform PageRank on \mathbf{V}
 - If a past contest, compare results to actual by calculating Kendall’s Tau
 - Output results
-

For all simulations, I chose to use a EuroFollowing Factor of 0.85, as this is approximately the p used by Google, the default used by Cleve Moler in his code, and used by Govan and Meyer in [11]. By running multiple simulations of over 100 tests, I found that changing the EuroFollowing Factor had no overall effect on the accuracy of the estimates.

I first ran simulations of Eurovision Song Contest 2005 Kiev. The contest was won by Helena Paparizou from Greece with “My Number One,” entirely in English. I first guessed at the English Scaling Factor (ESF) of 0.3. This brings the average additional votes from a country to a song in English to 10, so as not to outweigh the voting power of any one country. With an ESF of 0.3, I ran 100 simulations with each LVF from 0.1 to 1.0 in increments of 0.1 to find one with the lowest Kendall’s Tau distance to the actual results from the Kiev contest. The best was obtained with an LVF of 0.9. The results table, as compared to the actual, can be seen in Table 1. In Figure 2, the graph of the ESF along the x -axis and Kendall’s Tau along the y is shown.

I then ran the same simulation, except allowing the LVF and ESF to vary, each running from 0.1 to 1.0 in increments of 0.1 independently, so that each combination of LVF and ESF was tested. Using this method, I found the best LVF and ESF for the 2005 contest to be LVF = 0.6 and ESF = 0.8. The results are shown in Table 2.

Finally, I tested the to see if the same parameters are accurate for ESC 2006 Athens. The same method of finding the LVF and ESF was used as above. I found that for ESF = 0.6 and LVF = 0.8, $\tau = 0.13$, while for ESF = 0.9 and LVF = 0.2, $\tau = 0.16$. The difference in these τ values is fairly small, especially considering the results are not at all stable – in different runs of simulations, the results change greatly. (This phenomenon will be discussed further in Section 3.3.) The best result from 100 runs of simulations with these parameters is shown in Table 3. Possibly the most interesting thing about these results is that the ESF is incredibly high in 2006. This means that we would have expected non-English songs to come in higher based on country-to-country voting, and to compensate, the number of votes given to English songs was increased greatly. While the winner of ESC 2006 Athens was Finland, their song, “Hard Rock Hallelujah” by Lordi, was not

Simulation	Actual
Ireland	Greece
Serbia	Malta
Sweden	Romania
Greece	Israel
Estonia	Latvia
Hungary	Moldova
Bosnia	Serbia & Montenegro
Russia	Switzerland
Norway	Norway
Malta	Denmark
Finland	Croatia
Moldova	Hungary
Kendall's Tau Distance Simulation to Actual: 0.29	

Table 1: Simulation versus actual results of 2005 ESC (top 12 teams) with ESF = 0.3 and LVF = 0.9

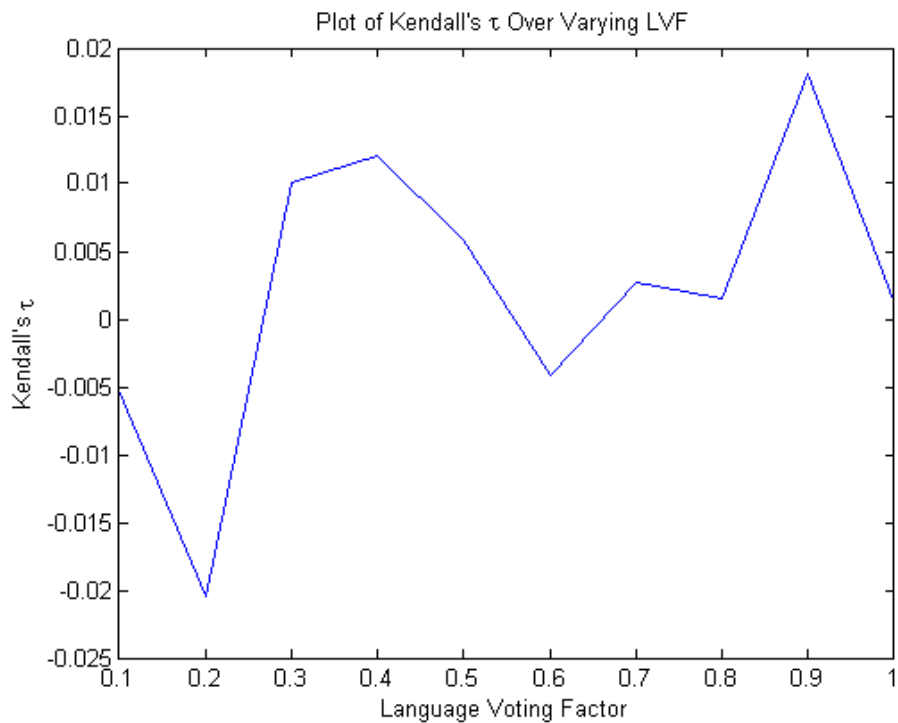


Figure 2: Graph of Kendall's Tau varying over different values of ESF

Simulation	Actual
Ireland	Greece
Sweden	Malta
Greece	Romania
UK	Israel
Israel	Latvia
Norway	Moldova
Moldova	Serbia & Montenegro
Switzerland	Switzerland
Malta	Norway
Bosnia	Denmark
Finland	Croatia
Denmark	Hungary
Kendall's Tau Distance Simulation to Actual: 0.29	

Table 2: Simulation versus actual results of 2005 ESC (top 12 teams) with $ESF = 0.6$ and $LVF = 0.8$

predicted to be a winner due to its entirely different style from previous Eurovision winners.[14]

3.2 ESC 2007 Helsinki: Who Will Win?

The ultimate goal of this mathematical model is to predict the results of Eurovision Song Contest 2007 Helsinki. Unfortunately, with the results in the previous section, it is difficult to decide what ESF and LVF to use to make the prediction. Additionally, as stated in Sections 2.4 and 3.3, the data, due to its small size and wide variations, has an extremely large variance, and thus the random voting numbers generated by the simulation vary widely. This is even more pronounced in all 39 countries, as their placings could vary wildly.

As a result, I have chosen to run two simulations – one with $ESF = 0.6$, $LVF = 0.8$ and one with $ESF = 0.9$, $LVF = 0.2$. The A representative sample of the placings of the top 12 countries in each are reported in Table 4.

In both simulations, though, the country coming in first the greatest number of times was Russia with, over multiple runs of 100 simulations each, an average of 37 appearances in the first simulation and 33 appearances in the second simulation. Observe that some countries, like the UK and Ireland, unsurprisingly consistently place high due to the fact that they sing in English and have won numerous contests with large numbers of points. Also note that Serbia, unlike in previous years, does not place high in either ESC simulation due to the fact that their entry, Marija Šerifović's "Molitva," is in Serbian.

We can compare these to the current online polls and betting odds. The Website Oikotimes.com, which is a Eurovision news site, compiles and updates a list of online polls' and bookmakers odds' choices for the winner of the upcoming Eurovision. The results are reported in Table 5. It is worth noting that Serbia, along with Spain and Switzerland, top the list with 8 poll results for them each. It is not surprising that these countries did not place highly on the representative sample lists. Spain and Switzerland each have only two victories in the history of Eurovision, and Switzerland is

Simulation	Actual
Ukraine	Finland
Russia	Russia
Greece	Bosnia & Herzegovina
Ireland	Romania
Sweden	Sweden
Armenia	Lithuania
Denmark	Ukraine
UK	Armenia
Moldova	Greece
Cyprus	Ireland
Finland	Turkey
Iceland	Macedonia
Kendall's Tau Distance Simulation to Actual: 0.37	

Table 3: Simulation versus actual results of 2006 ESC (top 12 teams) with $ESF = 0.9$ and $LVF = 0.2$

ESF = 0.6, LVF = 0.8	ESF = 0.9, LVF = 0.2
Russia	Sweden
Estonia	Ireland
Armenia	Armenia
Sweden	Moldova
UK	UK
Albania	Greece
Netherlands	Turkey
Ireland	Albania
Greece	Malta
Denmark	Switzerland
Malta	Russia
Moldova	Denmark

Table 4: Representative sample simulation results from simulations of ESC 2007 Helsinki

Country	Number of Appearances in Polls as Winner
Serbia	8
Spain	8
Switzerland	8
Sweden	6
Cyprus	5
Greece	3
Turkey	3
Macedonia	2
Russia	1
Lithuania	1
France	1
Belarus	1

Table 5: Summary of Polls Compiled by Oikotimes.com at [15]

the only one with a victory since 1975, when the current voting system was implemented and the data set starts.[10] Additionally, Sweden, which placed highly on both simulations, has six polls in its favor, the second highest. (These polls may be biased, as some of them were conducted on Websites with audiences almost entirely from a competitor country.)

In summary, I would predict that Russia is the most likely winner of ESC 2007, but that any of the countries listed in Table 4 are also more likely than not to win.

3.3 Flaws

The flaws of this model are worth noting. One of the most interesting outcomes of the simulations is that the results are not at all stable. In one run of 100 simulations, Serbia may be the most common winner, and in the next, it could be Greece. Even if the number of simulations is increased, the results do not stabilize. This is probably due to a consequence of the extremely high variance in the data. For example, the standard deviation of Malta giving points to songs in English is approximately 21.67.

Additionally, recall the first simplifying assumption: all voting follows past tendencies. In other words, the model is static, since it uses thirty years of data and cannot differentiate between the most recent data and the oldest data. Changes in political climates, cultures, preferences, song types, and other factors are not accounted for in this model. Animocities between Greece and Turkey were at their highest in the first years of the contest, after the Turkish invasion of Cyprus in 1975 [16], but have lessened in recent years. The model still would show a low number of votes for Turkey from Greece and Cyprus.

Not accounted for in this model is the switch to the use of televoting. Since 1997 in some countries and 1998 in others, the people of the countries have voted for their favorite songs by calling telephone numbers specially set up for the contest, instead of juries made up of people from the country. Spierdijk and Vellekoop show in their paper that since the inception of televoting, certain types of cultural voting, such as religious and patriotic voting, have become more pronounced, in

addition to language-preference voting.[6] This is due to the fact that the juries were more objective voters, as they cared more about the quality of the song than their country's relationship with other countries. Obviously, this is a flaw in the data. Since all voting today is done by televoting, the results since 1997/1998 might be better predictors of future success than earlier results. The model described in this paper does not deal with the difference in voting tendencies due to televoting at all, producing a possible flaw in the paper. The pre-televoting juries were still representatives of the voting country, so the tendency may not be particularly strong.

The same simplifying assumption completely removes all subjectivity from the voting model. My model cannot account for personal preference or the quality of an entry. Nor can it account for a recent track record of poor performance. Ireland has won the most contests with 7, but has not finished above 15th since 2003.[10] Ireland's past success far outweighs its recent shortcomings in song selection. The United Kingdom is in similar situation, as its and Ireland's votes are augmented by the fact that they always sing in English, which is the winningest language with 19 wins.[10] Thus, songs that are actually better than than Ireland's or the UK's will place lower because of the assumptions of the model.

4 Conclusion and Future Work

This paper explained the Eurovision Song Contest and developed a model for simulating the results of voting in the contest. The model used a modification of Google's PageRank algorithm, and integrated Monte Carlo simulation by randomizing the numbers within a range calculated from the data. The proper parameters for the simulation were determined by testing. The results of the 2005 and 2006 contests were recreated, and the results of the 2007 contest recreated. I found that the most likely winner of Eurovision Song Contest 2007 Helsinki is Russia.

While the model is a good start, as discussed in Sections 2.3 and 3.3, the model makes a lot of simplifications. Probably the first portion of the model to revise is to take into account more factors in modeling voting. A future model could take into account critics' reviews of the songs and poll results. Since the people determine the voting today, the polls could be accurate predictors of the outcome. Another interesting development would be to actually simulate the scoreboard – how many votes each country gets from each other country. Currently, the model only gives the final leaderboard, but simulating the entire set of results would also be a useful development. Finally, it would be useful to integrate recent performance into the model. Eastern Europe has done increasingly well in recent years (the last four winning countries have been Turkey, Ukraine, Greece, and Finland), and the model would likely be improved by accounting for the fact that they may continue to succeed in the contest.

Overall, the model provided a basis for predicting the results of future Eurovision Song Contests. While the model was not perfectly accurate, the results do often come close to finding the winner. The Internet and betting polls for the upcoming contest support the claim that this year, the simulation will have predicted some of the top Eurovision songs.

References

- [1] Jaquin, Patrick. "Eurovision's Golden Jubilee." *Union Européenne de Radio-Télévision / European Broadcasting Union*. 1 December 2004. Accessed 28 April 2007.

- <http://www.ebu.ch/en/union/diffusion_on_line/television/tcm_6-8971.php>.
- [2] “History: The Story.” *Eurovision.tv: The Official Website of the Eurovision Song Contest*. Accessed 28 April 2007. <<http://www.eurovision.tv/content/view/213/187/>>.
- [3] “History: 1956 - Lugano.” *Eurovision.tv: The Official Website of the Eurovision Song Contest*. Accessed 28 April 2007. <<http://www.eurovision.tv/content/view/249/153/>>.
- [4] “Eurovision Song Contest - Helsinki 2007.” *Eurovision.tv: The Official Website of the Eurovision Song Contest*. Accessed 28 April 2007. <<http://www.eurovision.tv/>>.
- [5] Gatherer, Derek. “Comparison of Eurovision Song Contest Simulation with Actual Results Reveals Shifting Patterns of Collusive Voting Alliances.” *Journal of Artificial Societies and Social Simulation*, Volume 9, Issue 2. 31 March 2006. Accessed 28 April 2007. <<http://jasss.soc.surrey.ac.uk/9/2/1.html>>.
- [6] Spierdijk, Laura and Michel Vellekoop. “Geography, Culture, and Religion: Explaining Bias in the Eurovision Song Contest.” Accessed 28 April 2007 <<http://www.rug.nl/economie/faculteit/medewerkers/SpierdijkL/eurovision.pdf?as=pdf>>.
- [7] Multiple pages. *The Diggiloo Thrush*. Accessed 28 April 2007. <<http://www.diggiloo.net>>.
- [8] Fenn, Daniel, et. al. “How does Europe Make Its Mind Up? Connections, cliques, and compatibility between countries in the Eurovision Song Contest.” *arXiv.org* 22 May 2006. Accessed 29 April 2007 <<http://www.arxiv.org/pdf/physics/0505071>>.
- [9] Mikheev, Andy. “History of Eurovision Song Contest 1956-2000.” *Eurovision Song Contest – Kazakhstan*. Accessed 29 April 2007 <<http://esckaz.com/history/1990e.htm>>.
- [10] O’Connor, John Kennedy. *The Eurovision Song Contest: 50 Years: The Official History*. London, United Kingdom: Carlton Books Limited, 2005.
- [11] Govan, Anjela Y. and Carl D. Meyer. “Ranking National Football League Teams Using Google’s PageRank.” *North Carolina State University*. Accessed 25 April 2007 <<http://www.ncsu.edu/crsc/reports/ftp/pdf/crsc-tr06-19.pdf>>.
- [12] Chartier, Timothy. “Google as a Markov Chain.” Computer Science 482 Course Notes, Davidson College.
- [13] Press, William H., et. al. *Numerical Recipes in Pascal: The Art of Scientific Computing*. Cambridge, United Kingdom: Cambridge University Press, 1989.
- [14] “And the winner is...” *Eurovision.tv: The Official Website of the Eurovision Song Contest*. Accessed 1 May 2007. <<http://www.eurovision.tv/content/view/697/262/>>.
- [15] Polls. *Oikotimes.com* 21 April 2007. Accessed 1 May 2007 <<http://www.oikotimes.com/polls.xls>>.
- [16] Housden, Tom. “Cyprus: How the crisis unfolded.” *BBC News*. 1 April 2004. Accessed 1 May 2007 <<http://news.bbc.co.uk/1/hi/world/europe/1760565.stm>>.

Appendices

A Representative Matlab Code

```
0001 function esc2005random( numsims, follow, eng, lang )
0002 %ESC2005RANDOM Simulates the 2005 ESC
0003 %   x = esc2005random(numsims, follow, eng, lang)
0004 %
0005 %   numsims - number of simulations to run
0006 %   follow - EuroFollowing Factor (probability that a country will follow
0007 %   its previous averages
0008 %   eng - English Scaling Factor (the votes to English will be scaled
0009 %   by this amount)
0010 %   lang - Language Voting Factor (the amount language-based voting will
0011 %   count)
0012
0013 %% Step 0: Looping Setup
0014 taus = zeros(1,numsims);
0015 tauzs = zeros(1,numsims);
0016 tauprobs = zeros(1,numsims);
0017 topteams = zeros(1,numsims);
0018 besttau = -1;
0019
0020 load allesdata;
0021
0022 %% Step 1: Fix data
0023   %There are some NaNs in there that need to be taken out.
0024
0025   [i,j] = find(isnan(ccvavg) == 1);
0026   for k=1:length(i)
0027       ccvavg(i(k),j(k)) = 0;
0028   end
0029
0030   [i,j] = find(isnan(ccvstdev) == 1);
0031   for k=1:length(i)
0032       ccvstdev(i(k),j(k)) = 0;
0033   end
0034
0035   [i,j] = find(isnan(lpavg) == 1);
0036   for k=1:length(i)
0037       lpavg(i(k),j(k)) = 0;
0038   end
0039
0040   [i,j] = find(isnan(lpstdev) == 1);
0041   for k=1:length(i)
0042       lpstdev(i(k),j(k)) = 0;
0043   end
0044
0045 %% Step 2: Strip non-participating countries
0046   nonparticipants = [3 14 16 22 25 36];
0047   numparticipants = length(ccvavg) - length(nonparticipants);
0048   %Armenia, Italy, Yugoslavia, Luxembourg, Morocco, and Slovakia
```

```

0049     for i = 1:length(nonparticipants)
0050         idx = nonparticipants(i);
0051         nonparticipants = nonparticipants - 1;
0052         ccvavg(idx,:) = [];
0053         ccvavg(:,idx) = [];
0054         ccvstdev(idx,:) = [];
0055         ccvstdev(:,idx) = [];
0056         lpavg(idx,:) = [];
0057         % lpavg(:,i) = [];
0058         lpstdev(idx,:) = [];
0059         % lpstdev(:,i) = [];
0060         countries(idx) = [];
0061     end
0062     %non-participating countries have been stripped
0063
0064     for sim = 1:numsims
0065
0066         %% Step 3: Randomize!
0067         % Step 3a: Randomize Country-Country Voting; Create Voting Matrix V
0068         V = zeros(numparticipants); %for speed
0069         for i = 1:numparticipants
0070             for j = 1:numparticipants
0071                 if ccvstdev(i,j) == 0 %standard deviation is 0; 1 or 0 samples
0072                     V(i,j) = ccvavg(i,j);
0073                 else %standard deviation is non-zero
0074                     V(i,j) = ccvavg(i,j) + ccvstdev(i,j) * randn;
0075                     %generates a random number using the normal distribution
0076                     % with mean ccvavg(i,j) and stdev ccvstdev(i,j)
0077                     %see MATLAB "doc randn" for explanation
0078                 end
0079             end
0080         end
0081         % Step 3b: Randomize Language Preference Matrix
0082         lpnew = zeros(numparticipants,11); %for speed(ish - we actually need this)
0083         for i = 1:numparticipants
0084             for j = 1:11
0085                 if lpstdev(i,j) == 0
0086                     lpnew(i,j) = lpavg(i,j);
0087                 else
0088                     lpnew(i,j) = lpavg(i,j) + lpstdev(i,j) * randn;
0089                 end
0090                 if lpnew(i,j) < 0
0091                     lpnew(i,j) = 0;
0092                 end
0093             end
0094         end
0095
0096
0097         %% Step 4: Scale English by English Scaling Factor (ESF)
0098         lpnew(:,1) = lpnew(:,1) * eng;
0099

```



```

0151 %         q(25) = [];
0152 %     end
0153     actual = [14 20 27 12 17 21 30 33 23 34 15 37 35 4 38 1 19 16 28 36 7 11 26 29 25 31 38 8 9 10];
0154     [tau,tauz,tauprob] = kendalltau(q,actual);
0155 %     disp(sprintf('Kendall Tau Distance PageRank to Actual (Final): %4.2f',tau));
0156
0157     taus(sim) = tau;
0158     tauzs(sim) = tauz;
0159     tauprobs(sim) = tauprob;
0160     topteams(sim) = q(1);
0161
0162     if tau > besttau
0163         besttau = tau;
0164         besttauz = tauz;
0165         besttauprob = tauprob;
0166         bestx = x;
0167         bestq = q;
0168         bestr = r;
0169     end
0170 end
0171
0172
0173 %% Step 9: Output
0174 %%Also adapted from Cleve Moler's pagerank.m
0175 tau = besttau;
0176 tauz = besttauz;
0177 tauprob = besttauprob;
0178 x = bestx;
0179 q = bestq;
0180 r = bestr;
0181 disp('Best Ranking:')
0182 disp('PageRank   PseudoVotes   Country')
0183 for k = 1:n
0184     j = q(k);
0185     disp(sprintf('%8.4f      %4.2f      %s',x(j),r(j),countries{j}));
0186 end
0187 disp(sprintf('Kendall Tau Distance PageRank to Actual (Final): %4.2f with probability %1.6f',tau,t));
0188 disp(sprintf('Average Kendall Tau Distance: %4.2f',mean(taus)));
0189 disp(sprintf('Average Kendall Tau Probability: %1.6f',mean(tauprobs)));
0190 disp(sprintf('Most Common Top Team: %s',countries{mode(topteams)}));
0191 %ret = mean(taus);

```

B Representative Data: Language-Preference Voting Averages

	English	French	Spanish	Italian	Portuguese	German	Scandinavian	Slavic	Turkish	Hebrew	Other
Albania	41.67	2.33	2.33	2.00	NaN	0.00	0.00	11.33	5.67	1.50	0.00
Andorra	39.33	4.00	14.33	3.00	NaN	0.00	0.00	4.00	0.00	4.00	6.00
Armenia	50.00	0.00	2.00	4.00	NaN	NaN	1.00	12.00	0.00	0.00	NaN
Belgium	37.22	2.56	4.67	1.00	1.75	4.67	7.33	4.67	5.00	3.00	4.67
Bosnia	36.44	3.44	1.22	4.00	0.20	1.67	5.00	12.78	6.00	1.33	5.83
Bulgaria	47.50	0.00	0.00	2.00	NaN	NaN	0.00	12.50	3.50	0.00	5.00
Estonia	41.91	3.09	0.27	1.33	0.00	1.50	10.25	5.91	1.40	1.50	6.75
Spain	34.82	2.00	0.36	7.67	2.33	5.75	2.00	6.09	4.30	3.13	12.13
Netherlands	34.60	3.20	2.20	4.33	1.00	3.50	6.50	6.60	7.11	2.43	6.50
Ireland	39.70	2.60	0.90	2.33	0.33	2.38	9.00	3.40	0.67	1.71	9.38
Iceland	39.33	2.22	1.00	1.00	2.00	1.57	11.67	4.89	1.50	0.17	9.29
UK	40.64	2.45	0.45	3.00	0.33	2.00	5.50	2.27	2.40	1.50	13.00
Israel	43.11	2.11	4.56	5.00	0.50	2.40	2.00	6.44	1.63	0.00	7.00
Italy	18.00	0.00	3.00	2.00	0.00	5.00	0.00	7.00	7.00	NaN	16.00
Austria	32.25	2.13	0.00	0.50	0.25	7.50	9.00	16.25	5.57	0.40	6.50
Yugoslavia	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Greece	33.44	1.67	3.56	6.00	0.20	7.50	1.00	9.89	2.13	2.50	10.17
Croatia	36.64	1.45	1.73	6.00	1.67	1.25	1.25	15.55	2.70	1.00	8.38
Cyprus	35.50	1.00	4.60	5.33	2.80	0.00	1.50	7.90	0.00	2.86	10.57
Latvia	47.14	1.43	0.57	1.00	0.00	4.25	6.00	8.29	0.00	0.83	7.00
Lithuania	52.67	2.33	0.00	1.00	0.00	0.00	0.00	7.83	0.67	0.60	1.67
Luxembourg	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Macedonia	40.33	1.67	0.17	3.00	4.00	0.00	0.50	12.67	4.33	1.80	2.67
Malta	41.18	0.27	1.82	3.67	0.33	2.63	2.25	7.45	3.90	1.00	6.38
Morocco	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Moldova	53.50	0.00	0.00	12.00	NaN	NaN	3.00	7.00	0.00	3.00	0.00
Monaco	43.00	5.00	1.00	0.00	NaN	0.00	1.00	10.67	2.33	6.00	0.00
Norway	38.30	4.50	0.40	1.33	2.00	1.63	8.50	5.70	1.00	1.14	6.75
Portugal	34.33	3.11	5.22	11.00	0.00	4.57	2.25	7.22	2.13	4.17	8.00
Poland	41.00	5.25	0.86	5.50	0.00	2.83	4.75	7.44	1.50	3.33	13.00
France	30.64	1.45	3.18	2.33	5.50	4.13	4.50	7.91	7.30	8.75	8.38
Romania	36.71	1.43	2.14	0.00	0.00	2.50	0.00	13.29	3.67	3.17	7.00
Sweden	40.00	2.18	1.45	3.00	0.67	0.63	10.25	7.55	0.90	1.25	8.50
Germany	38.50	2.60	1.10	3.00	0.00	1.71	1.00	10.20	8.33	3.00	5.29
Serbia	48.67	0.00	0.00	4.00	NaN	0.00	0.00	15.00	0.00	0.00	6.00
Slovakia	15.00	3.50	0.00	NaN	1.00	0.00	2.00	16.00	0.00	0.00	20.50
Slovenia	31.44	2.80	0.70	5.00	0.00	2.00	8.25	14.18	1.00	1.14	7.29
Finland	38.00	1.71	0.00	3.00	1.50	0.75	9.33	5.43	1.29	4.00	10.75
Switzerland	29.38	2.13	4.63	3.67	3.33	5.80	4.25	9.00	5.38	2.40	4.20
Denmark	46.38	3.00	1.75	3.33	0.00	1.80	3.50	5.25	1.88	1.33	6.60
Turkey	39.00	1.18	1.27	2.67	0.83	1.50	1.00	15.09	1.70	2.25	5.63
Ukraine	41.00	0.00	0.00	1.00	3.00	4.00	3.00	19.25	0.00	2.33	6.00
Hungary	30.67	1.67	3.33	3.00	0.00	3.00	2.50	7.33	2.00	4.00	7.00
Belarus	51.67	0.00	0.67	3.00	NaN	0.00	1.00	10.33	0.00	4.00	2.00
Russia	43.13	3.13	2.50	4.00	0.00	3.40	3.50	9.00	0.57	2.17	6.40