Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

Comparison of Electoral Systems and Optimization of Voting Advice Questionnaires

Bachelor's Thesis

Nicola Lo Russo

nlorusso@ethz.ch

Distributed Computing Group
Computer Engineering and Networks Laboratory
ETH Zürich

Supervisors:
Judith Beestermöller, Andrei Constantinescu
Prof. Dr. Roger Wattenhofer

July 14, 2023

Acknowledgements

I am grateful for the help and support provided by Judith Beestermöller and Andrei Constantinescu throughout the completion of this thesis. This work would not have been possible without their invaluable contributions, guidance, and encouragement. Working alongside them has been a true pleasure. I also want to thank Prof. Dr. Roger Wattenhofer for granting me the opportunity to work on this topic.

Abstract

Our work is divided into three parts. In the first chapters, we simulate and compare different electoral systems using the 2019 Swiss parliamentary elections smartvote dataset. The results show that the dataset is strongly homogeneous, with little to no cantonal differences in terms of political views and election outcomes. The second part focuses on modeling an election with election trees. In particular, we define and model the Candidate Comparison-Based Election Tree, which allows describing the entire smartvote population with $O(\log_2(n))$ questions, where n is the number of voters. The third and last part of this work considers the problem of voting fatigue among smartvote users and proposes possible solutions. We outline an innovative way of building the questionnaire experience. We design and implement an algorithm that, personalizing the question ordering, converges faster to the final voting advice. The algorithm chooses the next question relying on the previous answers of the voter and all the answers of the candidates. This approach significantly reduces the number of questions needed for 90% of voters to identify their ultimate winner within the top 10 candidates, resulting in a substantial decrease ranging from 37% to 61% when compared to the original order of questions. This optimization allows users affected by voting fatigue to have more reliable voting advice provided, even if they opt to conclude the questionnaire before its completion.

Contents

Chapter 1

Introduction

smartvote is an online voting advice platform founded in 2003 by Politools, a non-profit and non-partisan organization based in Bern. Using a specific questionnaire on various political issues, voters can compare their positions with those of candidates and receive a voting recommendation. All our analyses are based on the smartvote dataset of the Swiss parliamentary elections of 2019, that contains candidates' and voters' data related to that election. We gained access to such dataset thanks to Politools, the association behind smartvote. Our work builds upon this tool, and this chapter has the goal of giving the reader a detailed overview of the dataset.

This work is composed of three main parts. In the first part (Chapters 2-8), we implement, simulate, analyze, and compare different electoral systems on the smartvote dataset. In particular, the voting systems included are: plurality, two-round, instant runoff, plurality veto, sortition. Such systems are evaluated in comparison to the utilitarian rule.

In the second part (Chapter 9), we discuss about election trees. We define and implement Candidate Comparison-Based Election Trees (CCBET). This tool helps us to define an entire election with n voters with only $O(2\log_2(n))$ questions.

In the third part (Chapter 10) we develop a new way of thinking about the questionnaire, to reduce voting fatigue among the users. Previous analyses (see [1]) showed that some smartvote users tend to not complete the questionnaire, asking for the final voting advice after having answered fewer questions than the total. After some theoretical reasoning, we implement an algorithm that personalizes the order of the questions in the questionnaire. This algorithm makes users' profiles converge faster to the final voting advice, allowing voters affected by voting fatigue to obtain a more reliable result after less than 75 questions.

1.1 Description of the smartvote Questionnaire

The smartvote questionnaire involves a set of questions aimed at determining the users' preferences given a list of candidates. It offers two versions: the "Deluxe" questionnaire, featuring 75 questions, and the “Rapid” version, which consists of only 31 questions.

The questionnaire covers a wide range of social, economic, and political themes to gauge the user’s affinity towards each candidate. Its primary objective is to accurately position users on the political spectrum, allowing for effective comparison with the positions of the candidates.

In addition, each question is accompanied by a weight that indicates its importance to the user. The user has also the option to not assign any weight at all.

The grade of similarity to each candidate is measured in relation to the distance of the user and the candidate on the k-dimensional space, where k is the number of answered questions by the user. In smartvote, candidates always answer questions, but voters might not answer all questions. The distance to every candidate is therefore computed with respect to a subset of $k \leq 75$ answered questions. Formally:

Definition 1.1 (Distance and Matching)**.** Let $v, w, c \in \mathbb{R}^k$ be the vectors representing respectively the k voter’s answers, the associated weights, and the corresponding candidate’s answers to the questionnaire. Then the distance between v and c is defined as follows:

$$\mathrm{Dist}(v,c) := \|w^T (v-c)\|_2 = \sqrt{\sum_{i=1}^{k} (w_i \cdot (v_i - c_i))^2} \tag{1.1}$$

The grade of affinity of v and c is finally given by the matching:

$$\mathrm{Matching}(v,c) = 100 \cdot \left(1 - \frac{\mathrm{Dist}(v,c)}{\sqrt{\sum_{i=1}^{k} (100 \cdot w_i)^2}}\right) \tag{1.2}$$

Then, the smartvote questionnaire returns the linear ordering of the candidates in descending order of their affinities to the user. In this work, we will refer to such ordering as the (user’s) *profile*.

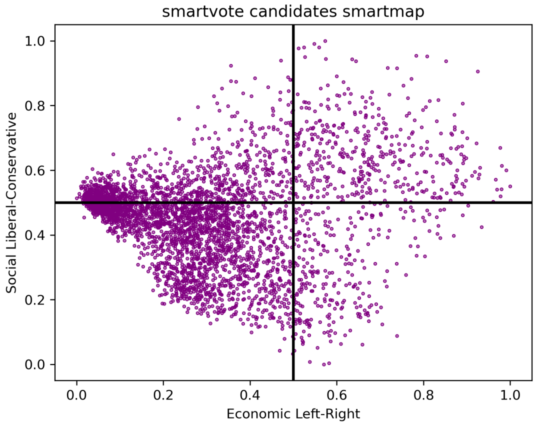
More information on the calculation of the recommendations can be found in the official smartvote documentation.[1]

1.2 Description of the Dataset and Methodology

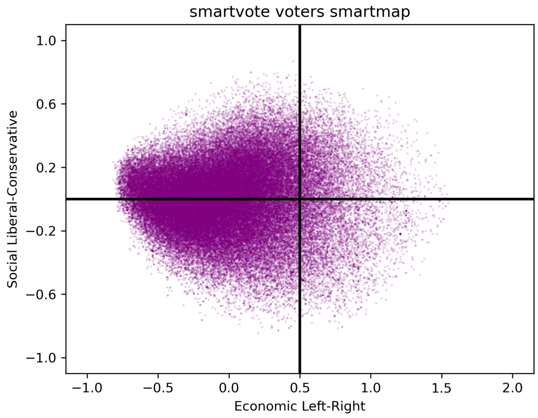
This work focuses on a dataset obtained for the 2019 Swiss federal election, which comprises 427,572 voters and 4,663 candidates.

The dataset comprehends two main files, one for the voters’ answers and one for the candidates’ answers, on which we will focus our analyses. The two documents also include other features, such as the age of the candidates or the district of the user. Most of this

[1] www.smartvote.ch/method-description



(a) Candidates smartmap.



(b) Voters smartmap.

Figure 1.1: Distribution of smartvote candidates and voters on the political compass. Dimensionality reduction is computed using principal component analysis.

additional information goes beyond the goal of this work and is not used in our analysis. However, certain attributes such as the candidates' x and y coordinates on the political compass, referred to as `smartmap_x` and `smartmap_y`, are relevant for interpreting the results.

smartvote also registers data of users and candidates who do not fill out the entire questionnaire. Even if approximately 63% of the voters decided to fill the "Deluxe" version, most of them did not answer *all* of the questions.

For this study, we have excluded candidates who did not complete the entire questionnaire as well as a small number of candidates (12) who lack smartmap coordinates. As a result, the total number of candidates has been reduced to 3,913. Each voter and candidate is identified by a unique identifier, belonging to the respective sets $\{1, \ldots, v^*\}$ and $\{1, \ldots, c^*\}$, where $v^* = 427,572$ and $c^* = 3,913$.

Chapters 2 to 8 include simulations of electoral systems. These simulations have been conducted with the whole voters' pool (427,572 voters) and the reduced candidates' pool (3,913 candidates).

1.2.1 Limitations and Biases of the Dataset

In order to be transparent to the reader, it is dutiful to spend some lines describing some important limitations and biases of the dataset we are working with, and how we decided to tackle them.

First and foremost, the pools of candidates and voters are not representative of the Swiss political spectrum. As we see in Figure 1.1, the pools are heavily leaning towards the left side of the spectrum. Only a small amount of candidates and voters belong to the first and the fourth quadrant of the compass. In reality, the relative majority of the Swiss parliament is composed of members of the Swiss People's Party (SVP), which is historically positioned in the first quadrant of the political compass.

A possible bias in the answers is due to the fact that the order of the questions is not randomized. As a result, users could get tired after answering many questions, and the quality of their last answers could be lower. This bias will be the focus of Chapter 10.

An important assumption made in this work is that the questions used in the questionnaire are representative, neutral, and comprehensive. We consider questions to be representative when they cover Swiss political themes. Additionally, we assume that the questions are neutral in the sense that they are formulated impartially and comprehensive in the sense that they cover *all* the major Swiss political themes. Although this approach has received criticism in the past, as reported in [2], such criticisms go beyond the scope of this study.

In this work, we also assume the candidates' pool, the voters' pool, and the questionnaire are representative of an imaginary single-winner election. Furthermore, we assume that a voter's list of preferences accurately represents the candidate they would ultimately vote for.

1.2.2 Principal Component Analysis

In the next pages, the presentations of the results will frequently involve political compasses, which visually display candidates and voters in a two-dimensional space. As reported in [3], studies have shown that the two axes usually represent economic policies (left-wing versus right-wing) and social policies (libertarian versus authoritarian).

On the horizontal axis, the left-wing end typically signifies policies associated with wealth redistribution, social equality, and a larger role of the government in the economy. On the other hand, the right-wing end signifies policies that prioritize individual freedom, free markets, and limited government intervention in the economy.

On the vertical axis, which represents the spectrum of social policies, the bottom end represents a preference for personal freedom and limited government intervention in personal lives. Conversely, the top end represents a preference for strong governmental control and regulation of personal behavior and societal norms.

The reduction from the original 75-dimensional space to $\mathbb{R}^2$ is computed using principal component analysis (PCA) as a mean of dimensionality reduction. It is worth noting that the reduction of dimensions through PCA may result in a significant loss of information. Therefore, the interpretations of the two-dimensional plots should be approached with care and caution.

While the smartvote dataset provides a set of two coordinates for each candidate through the `smartmap_x` and `smartmap_y` variables, obtained with PCA and reported in Figure 1.1a, there is no such data available for the voters. Therefore, to provide a clearer understanding, as well as to better analyze the results in the following chapters, we computed a PCA also on the subset of voters who completed *all* 75 questions.

It should be noted that we only performed PCA on this limited subset of voters because the varying sizes of the voters' vectors (many voters left some questions blank) would have led to inconsistencies in the process of dimensionality reduction. By focusing on the subset of voters who completed all questions, we were able to perform PCA on a consistent set of data and obtain reliable results. Figure 1.1b clearly shows a notable shift towards the center-left of the political compass. It is important to consider this observation when interpreting the results that will follow.

Chapter 2

Utilitarian Rule

It is important to say that the utilitarian rule is the reference electoral system, since it theoretically maximizes social welfare, but it assumes data that is not available to electoral authorities. Therefore, it can not really be implemented in practice, and one's goal should be to approximate it as well as possible. In the context of smartvote, the winner is defined as the candidate c^* who minimizes the sum of their Euclidean distances to all voters. More formally:

Definition 2.1. Let V be the set of voters, C be the set of candidates and, for each voter $v \in V$ and candidate $c \in C$, let $\text{Dist}(v, c)$ be their Euclidean distance in the sense of Definition 1.1. The winner[1] c^* of the utilitarian rule is defined as:

$$c^* = \text{argmin}_{c \in C} \sum_{v \in V} \text{Dist}(v, c) \tag{2.1}$$

This voting system assumes that the goal of an election is to choose a candidate who provides the greatest benefit to society as a whole, and not just to a particular group or individual. The use of this electoral system implies that the purpose of government is to promote the greatest good for the greatest number of people and that the best way to achieve this is to elect a candidate that maximizes overall happiness or well-being.

When applying this system to the smartvote dataset we get the results shown in Figure 2.1. As expected, this plot clearly shows strong similarities to Figure 1.1b, outlining as winner Candidate 1177, who positions themselves in the center-left of the political compass.

To discuss the overall satisfaction of the smartvote population, we can modify Definition 1.1 in the following way:

Definition 2.2 (Entire population (EP) matching)**.** Let V be the set of voters, C be the set of candidates and, for each voter $v \in V$ and candidate $c \in C$, let $\text{Dist}(v, c)$ be their Euclidean distance in the sense of Definition 1.1. Moreover, let $w_{i,v}$ be the weight assigned from voter v to answer $i \in \{0, \dots, k_v\}$, where $k_v \in \{0, \dots, 75\}$ is the number of questions answered by voter v.

[1] For simplicity we assume it's unique.

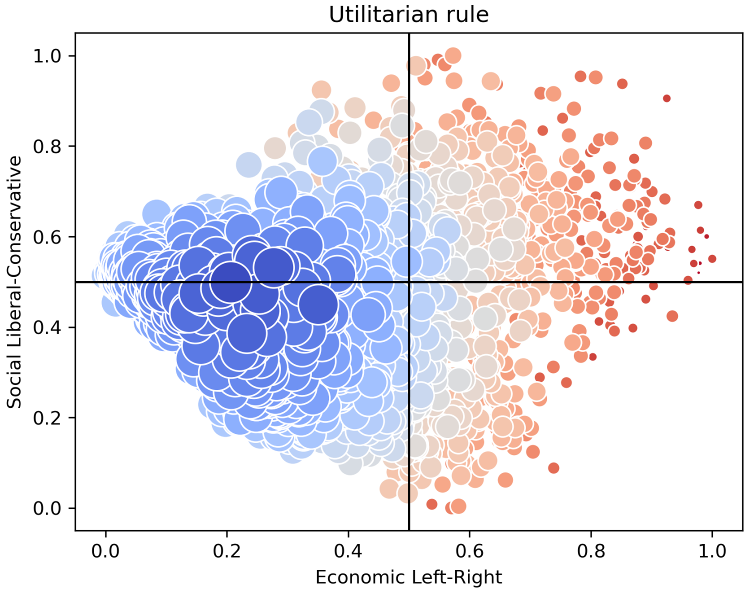


Figure 2.1: Utilitarian rule results. The size of the points is inversely proportional to the total distance to all voters. Color palette ranges between red (greatest distance) and blue (smallest distance). We see a strong similarity to Figure 1.1b.

Then, we can define the normalization factor $\text{MaxDist}_{\text{EP}}$ as the maximum possible attainable distance:

$$\text{MaxDist}_{\text{EP}} = \sum_{v \in V} \sqrt{\sum_{i=1}^{k_v} (100 \cdot w_{i,v})^2} \tag{2.2}$$

In the end, following the same reasoning of Definition 1.1, we can quantify the grade of affinity of a candidate to the set of voters V as:

$$\text{Matching}_{\text{EP}}(c) = 100 \cdot \left(1 - \frac{\sum_{v \in V} \text{Dist}(v, c)}{\text{MaxDist}_{\text{EP}}}\right) \tag{2.3}$$

Using Equation 2.3, we can quantitatively assess the level of social welfare/satisfaction among the entire smartvote population within the framework of a utilitarian rule-oriented election.

Table 2.1 shows that the winner matches at 61.67% the political ideas of the entire population. This outcome is similar to the general level of trust that the actual Swiss population has in their government, which averaged 64.63% over the period 1995-2019, according to data from the Federal Statistical Office[2]. An interesting observation emerges

[2]Trust in the Federal Council - Index from 1 (no trust) to 10 (complete trust), FSO number ind-e-21.02.30.1606.01.01.

candidate	**1177**	**2325**	**77**	**3884**	**2254**	**1258**	**3593**	**2274**	**1315**	**2987**
matching (%)	61.67	60.57	60.36	60.27	59.87	59.85	59.8	59.73	59.73	59.65

Table 2.1: Top-10 winners of the utilitarian rule and their matching to the entire population. The difference in matching between the first and second is smaller than the difference between the second and the tenth.

when we compare the matching scores between the winner and the runner-up, as well as between the runner-up and the ninth candidate. Surprisingly, the difference in matching between the winner and the runner-up is greater (1.1%) than the difference between the runner-up and the tenth candidate (0.92%). This finding shows that among the ten candidates, all of whom belong to the center-left area of the political compass, Candidate 1177 exhibits a significant lead over the others.

Although the overall satisfaction level is slightly above the 60% threshold, we still lack information regarding the social welfare of individual cantons. It is important to note that a candidate whose entire population matching grade is 100% to half of the population and only 20% to the other half would have the same overall satisfaction score as a candidate who satisfies everyone at 60%. Despite their vast differences, they would receive identical overall scores. To further explore this issue, we simulate the utilitarian election at the cantonal level, dividing the voter pool by canton and determining their winners.

Table 2.2 reports that the utilitarian rule yields similar results when applied at the canton level. Only in seven out of the 26 cases does the winner differ from the "global" winner. This outcome strongly suggests that the smartvote population is homogeneous when analyzed based on cantonal divisions. Moreover, the only other two candidates who win in some cantons are Candidate 3884 (with 5 preferences) and Candidate 2325 (with 2 preferences). Not surprisingly, these two candidates are ranked second and fourth in the national election (cfr. Table 2.1). However, it is noteworthy that Candidate 77, who holds the third position in the national election, does not emerge as the winner in any of the cantons.

Furthermore, the level of satisfaction among the population appears to be consistent across all cantons: it appears that a population tends to elect a winner who satisfies it at approximately 60%, regardless of the canton size. This observation is also supported by the linear relationship between the sum of distances of the canton winner and the number of voters per canton, as shown in Figure 2.2. These results suggest that the utilitarian rule provides a fair representation of the cantons' preferences.

canton	voters number	winner	matching (%)
Aargau	40861	**1177**	60.57
Appenzell Ausserrhoden	1186	**1177**	59.43
Appenzell Innerrhoden	493	**1177**	58.54
Basel-Landschaft	12349	**1177**	61.22
Basel-Stadt	11766	**1177**	65.08
Bern	84780	**1177**	62.22
Freiburg	19302	**1177**	61.71
Genf	9633	**1177**	63.59
Glarus	539	**1177**	59.83
Graubünden	10030	**1177**	60.6
Jura	1936	**1177**	63.16
Luzern	26579	**1177**	60.88
Neuenburg	6749	**1177**	62.69
Nidwalden	488	**3884**	59.69
Obwalden	1070	**3884**	60.1
Schaffhausen	2519	**1177**	61.58
Schwyz	7273	**3884**	59.51
Solothurn	13107	**1177**	61.24
St. Gallen	23494	**1177**	60.2
Tessin	3495	**1177**	61.46
Thurgau	11757	**2325**	59.92
Uri	1001	**3884**	59.4
Waadt	22707	**1177**	62.8
Wallis	17860	**2325**	60.57
Zürich	89665	**1177**	62.69
Zug	6933	**3884**	60.64

Table 2.2: Utilitarian rule applied regionally. Seven cantons elect a different winner from the global one. Entire population matching scores are homogeneously spread around 60%, irrespective of the winner or the canton size.

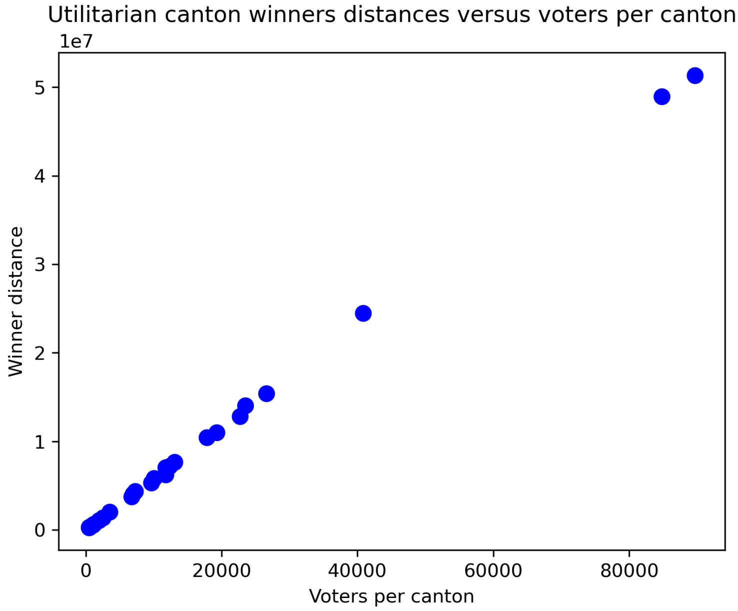


Figure 2.2: Sum of distances of winning candidates in each canton is directly proportional to the number of voters per canton. Satisfaction among cantons is therefore similar and not affected by voters pool size.

2.1 Distortion

In the process of comparing different electoral systems, one needs to define a measure to assess *how different* are the outcomes of the systems. To quantify such outcomes, we need to introduce the concept of metric distortion, which allows us to compare two different electoral systems in terms of global satisfaction.

When defining metric distortion, the utilitarian rule plays a key role. Indeed, distortion relies on the assumption that the utilitarian rule achieves the highest possible level of satisfaction for a given population. The outcome of the utilitarian rule is therefore the baseline to compare the outcomes of other electoral systems. The closer the outcome of a particular voting system aligns with the utilitarian outcome, the more favorable it is considered.

Distortion quantitatively measures the extent to which the outcomes of an election are altered or distorted by a particular voting method. In other words, metric distortion assesses how satisfied the voters are under a particular electoral system. If the metric distortion is high, the chosen system may not accurately reflect the collective preferences, potentially leading to dissatisfaction. Conversely, if the metric distortion is low (close to 1), the satisfaction grade of the population is high.

Mathematically, as Definition 2.3 states, distortion is the ratio between the sum of the distances of the winning candidate to the voters under a chosen voting system and the sum of the distances of the winning candidate to all the voters under the utilitarian rule.

Definition 2.3 (Distortion)**.** Let $\mathrm{SW}(c)$ be the variable that quantifies the lack of social welfare of the total population given by the election of candidate c, defined as:

$$\mathrm{SW}(c) = \sum_{v \in V} \mathrm{Dist}(v, c) \tag{2.4}$$

Moreover, let R be an electoral system, and let c_R be the winning candidate under R.

Then, the metric distortion $d(R)$ of the given electoral system R is defined as follows:

$$d(R) = \frac{\mathrm{SW}(c_R)}{\min_{c \in C} \mathrm{SW}(c)} \tag{2.5}$$

In the next chapters, we will often refer to Equation 2.4 as *satisfaction level/grade* of a population.

Chapter 3

Plurality Voting System

The plurality voting system (also referred to as the top voting system) is an electoral system in which each voter votes for a single candidate, and the candidate with the most votes is declared the winner. Under this system, the candidate who manages to secure the highest number of votes, regardless of whether they achieve an outright majority or not, emerges as the winner. Irrespective of how the votes are distributed among the candidates, the sole criterion for victory is obtaining more votes than any other candidate.

3.1 National Plurality Election

In the context of smartvote, the simulation of the plurality electoral system can be achieved by attributing one vote to a candidate whenever they rank first place in a user's voting advice profile.

Candidate	Votes
3884	6162
2325	3659
794	3065
198	2990
1177	2813
77	2717
2254	2233
3655	2229
3809	1986
3442	1943

Table 3.1: Top-10 winning candidates of plurality voting system applied to the smartvote dataset.

Table 3.1 displays the top-10 winning candidates resulting from the simulation of the plurality electoral system using the smartvote dataset. Notably, Candidate 3884 manages

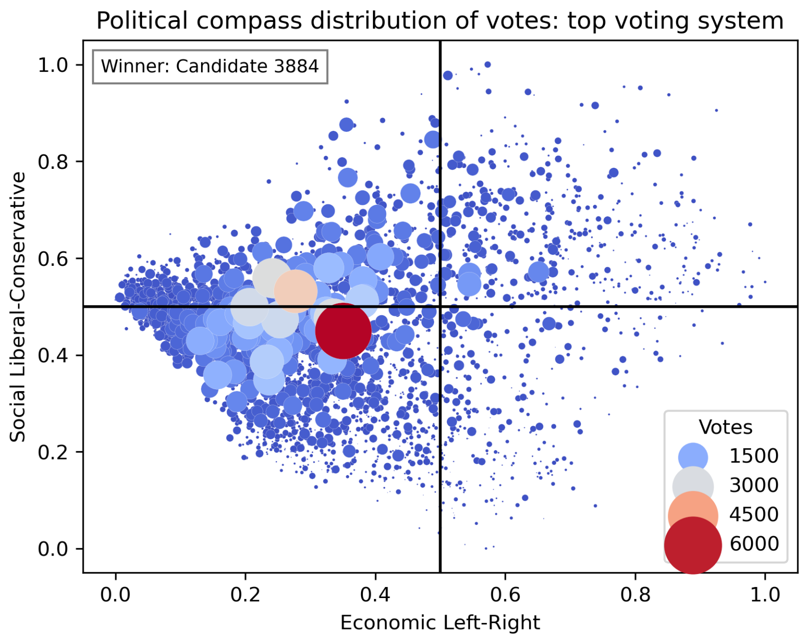


Figure 3.1: Results of the plurality voting system on the national level. The size of the dots is proportional to the number of votes a candidate received.

to secure the election with a significant margin of votes (over 2500) compared to the runner-up. An interesting observation is that the winner only acquires $\frac{6162}{427{,}572} = 1.44\%$ of the total votes.[1] This highlights the influence of several factors, such as a large number of candidates and the concentration of candidates in political areas with a large number of smartvote users (cfr. Figures 1.1a and 1.1b), which contribute to the decentralization of votes. Many voters tend to divide their votes between two candidates who have similar positions.[2] Conversely, Candidate 3884's success can be attributed, at least partially, to a smaller number of candidates in their political area compared to their opponents. Figure 3.1 shows the political compass distribution of votes.

3.2 Cantonal Plurality Election

To gain a better understanding of the cantonal contributions to the national results, we analyze how the different cantons voted. By examining Table 3.2, we observe that the smartvote dataset appears to be homogeneous also with respect to the plurality voting system. In only six out of the 26 cantons, a candidate other than the national winner wins. Among these outliers, only Jura and Neuenburg do not feature the winner within their top three candidates. Interestingly, only four candidates, who do not rank in the

[1] Note this would likely fail to hold in practice, due to political lobbying and strategic voting.
[2] More about this phenomenon: https://en.wikipedia.org/wiki/Vote_splitting

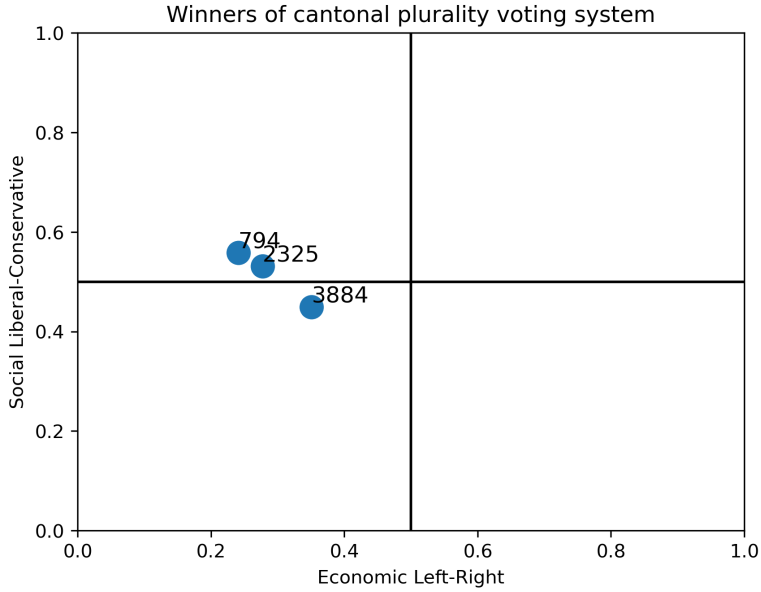


Figure 3.2: Political coordinates of the three cantonal plurality winners.

national top-10, manage to secure a place in the top three positions within any canton, as highlighted in bold in Table 3.2.

Additionally, when we analyze the political compass cantonal distributions of winners, we notice similar patterns. Even in the cantons with results that differ from the national outcome, the winner comes from the center-left political area, as can be seen in Figure 3.2.

Based on the provided results, we can conclude that the smartvote population, when subjected to the plurality voting system, appears to be homogeneous, exhibiting little variation among cantons. This suggests that political preferences within the dataset do not significantly differ based on cantonal boundaries. Regardless of their size, all cantons seem to share similar political views.

However, it is important to note that the plurality voting system has a notable drawback: it may result in winners who only receive a small percentage of the total votes cast. This is a limitation of the plurality electoral method, as the chosen candidate may not necessarily reflect the preferences of the majority of voters.

Canton	Winner (% votes)	Runner-up	Third place
Aargau	3884 (1.69%)	2325	3655
Appenzell Ausserrhoden	3884 (2.11%)	794	2255
Appenzell Innerrhoden	3884 (2.64%)	**1706**	**3050**
Basel-Landschaft	3884 (1.55%)	198	2325
Basel-Stadt	3884 (1.03%)	2254	1177
Bern	3884 (1.36%)	2325	198
Freiburg	794 (1.63%)	77	3884
Genf	3884 (1.11%)	77	2325
Glarus	3884 (2.6%)	**2723**	2325
Graubünden	3884 (1.55%)	2325	1177
Jura	794 (2.17%)	77	3809
Luzern	3884 (1.88%)	2325	198
Neuenburg	794 (1.87%)	77	**1315**
Nidwalden	3884 (3.07%)	3809	198
Obwalden	3884 (1.59%)	3655	2254
Schaffhausen	2325 (1.31%)	3884	3655
Schwyz	3884 (2.09%)	2325	198
Solothurn	3884 (1.67%)	2325	198
St. Gallen	3884 (1.68%)	2325	198
Tessin	3884 (1.72%)	1177	794
Thurgau	3884 (1.77%)	2325	198
Uri	3884 (1.8%)	2325	**3050**
Waadt	794 (1.4%)	77	3884
Wallis	794 (1.63%)	77	3884
Zürich	3884 (1.36%)	1177	2254
Zug	3884 (1.89%)	2325	198

Table 3.2: Winners, runner-ups, and third-placed candidates of plurality voting system for each canton. The majority of cantons align with the national results. Only six cantons elect a different candidate than the national winner. In bold candidates who are not present in the top-10 national results. In brackets percentage of voters received by the winner.

Further tests were conducted, dividing the pool of voters based on their education level, questionnaire type, and political interest. The results of these simulations are presented in Figures C.5 to C.8, which can be found in the appendix. The similarity of these figures highlight the homogeneity of the dataset with respect of any category-based division of the voters' pool.

The winner of the plurality voting system, as we have seen in the previous sections, is Candidate 3884. Candidate 1177, the winner under the utilitarian rule, only manages to get to fifth place. Figure 3.3 summarizes the differences between the two voting rules. We can see that Candidate 1177 is more left-shifted than Candidate 3884. This difference

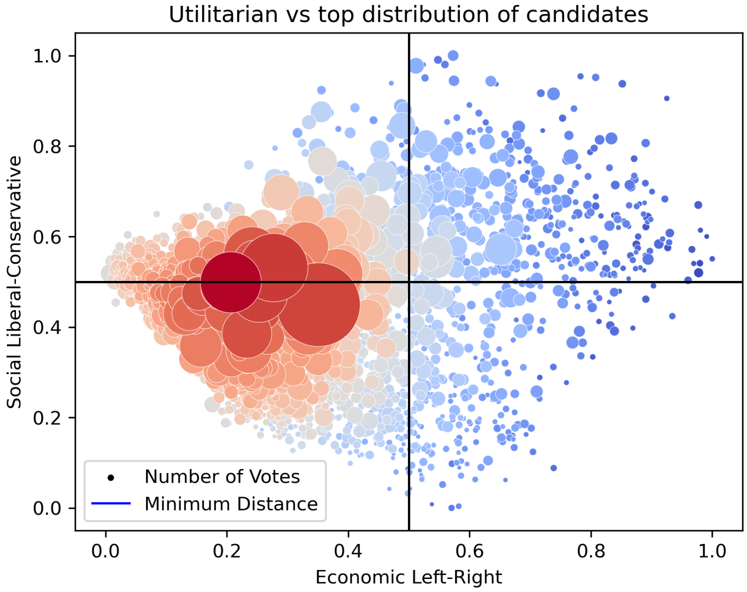


Figure 3.3: Comparison of plurality and utilitarian rule. The size of the dots refers to the number of votes received under plurality, color shade represents the utilitarian distance of the candidates to the voters. The more red the dot is, the smaller the distance of the candidate to all the voters is.

between the outcomes of the two systems leads us to some questions: how satisfied are the voters, when electing a candidate only voted by 1.44% of the total population? How satisfied are the cantons?

To answer this question, make use of the concept of metric distortion, described in Definition 2.3.

Table 3.3 displays the distortion values of the plurality voting system for each canton as well as globally. To establish baselines, we initially consider the distance of the utilitarian winner at the *global* level (Candidate 1177) and compare it to the distances of the global and cantonal plurality winners (second column). Then, we consider the distances of the utilitarian winners at the *cantonal* level and compare them against the distances of the global and cantonal plurality winners (third column). In both cases, the distortion values range between 1 (optimal distortion) and 1.13.

Based on these observations, we can conclude that even if plurality winners are elected by a small percentage of the population, overall population satisfaction remains high. Additionally, the distortion values do not appear to be influenced by canton sizes, indicating that the electoral system's impact is consistent across different cantons. The similarity in satisfaction levels between the two electoral systems can be attributed to the proximity of the candidates' positions in the 75-dimensional space $\mathbb{R}^{75}$.

Canton	Distortion	Cantonal distortion
National	**1.04**	-
Aargau	1.04	1.00
Appenzell Ausserrhoden	1.04	1.01
Appenzell Innerrhoden	1.04	1.00
Basel-Landschaft	1.04	1.02
Basel-Stadt	1.04	1.13
Bern	1.04	1.06
Freiburg	1.06	1.02
Genf	1.04	1.09
Glarus	1.04	1.01
Graubünden	1.04	1.02
Jura	1.06	1.03
Luzern	1.04	1.01
Neuenburg	1.06	1.02
Nidwalden	1.04	1.00
Obwalden	1.04	1.00
Schaffhausen	1.03	1.03
Schwyz	1.04	1.00
Solothurn	1.04	1.01
St. Gallen	1.04	1.00
Tessin	1.04	1.05
Thurgau	1.04	1.01
Uri	1.04	1.00
Waadt	1.06	1.03
Wallis	1.06	1.02
Zürich	1.04	1.05
Zug	1.04	1.00

Table 3.3: Distortion of plurality voting system on the national and cantonal level. The "Distortion" column refers to the utilitarian global winner as the baseline. "Cantonal distortion" column refers to cantonal utilitarian winners as baseline (cfr. Table 2.2). Values are rounded to two decimals.

Chapter 4

Egalitarian Rule

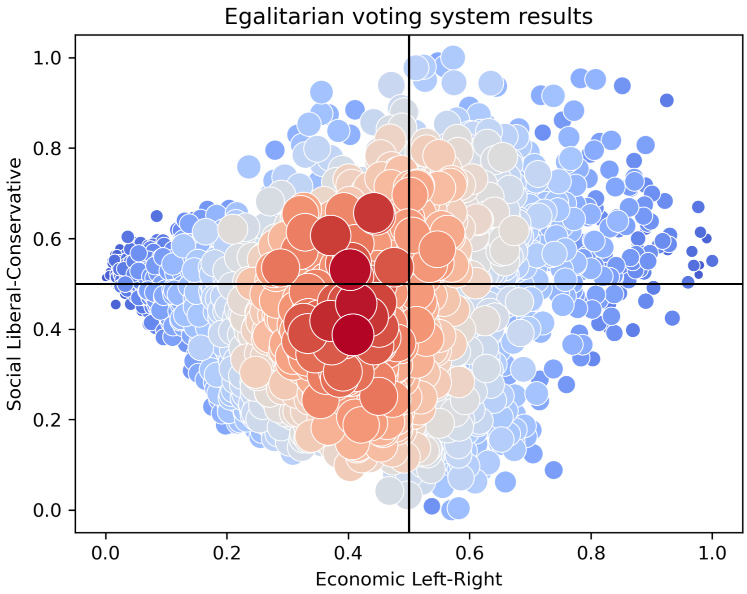


Figure 4.1: Results of the smartvote population under the egalitarian rule. The size of the dots is inversely proportional to the distance of the candidates to the most distant voter.

The egalitarian rule ranks the candidates with respect to their distance to the furthest voter. The winner is defined as the candidate who minimizes the distance to their most distant voter. In other words, under the egalitarian rule, the aim is to make the least happy voter as happy as possible.

Figure 4.1 represents the outcome of an egalitarian election simulated with the entire smartvote dataset. Comparing these results to those of the utilitarian rule, we observe that the candidates are more inclined toward the center. This outcome is intuitive when considering the contrasting objectives of the two systems.

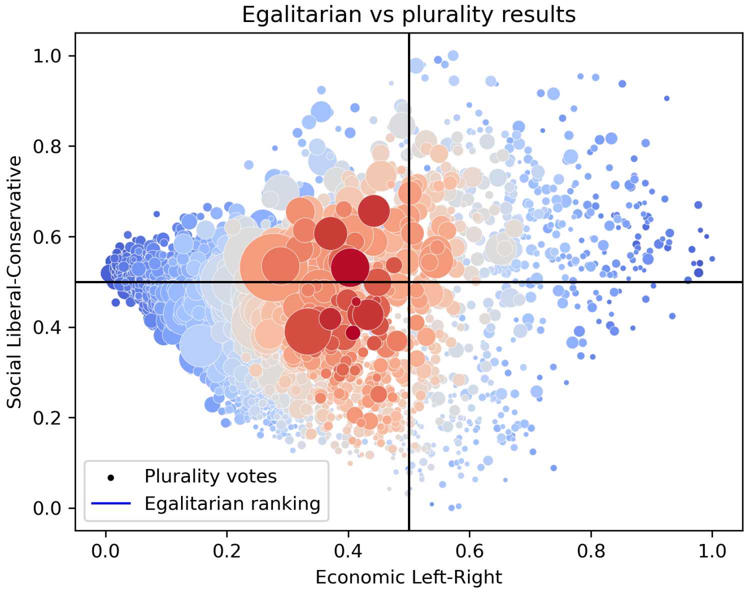


Figure 4.2: Comparison of egalitarian rule and plurality voting system. The size of the dots refers to the number of votes received under plurality, color shade represents the egalitarian ranking of the candidates. The more red the dot is, the better the candidate ranks under the egalitarian rule.

The utilitarian rule strives to maximize social welfare for the entire electorate, taking into account the overall preferences and well-being of the voters. In contrast, the egalitarian rule prioritizes the satisfaction of the least happy voter, aiming to make them as happy as possible.

Considering that the least happy voter is likely to be positioned within the right half of the political compass (as suggested by the data), the outcome of the egalitarian system tends to shift towards the right. This shift occurs because the system seeks to address the concerns and preferences of the least satisfied voter, potentially leading to a candidate selection that aligns more closely with their position.

Figure 4.2 presents a comparison between the egalitarian rule and plurality voting system. The visual representation vividly illustrates the contrasting outcomes of the two systems. The winner under the egalitarian rule, Candidate 1813, is represented by the darkest red dot on the graph. However, its small size indicates that this candidate received only a limited number of votes in the plurality system.

The graph also shows that most of the largest dots, which represent candidates who got the highest number of votes under the plurality system, are not even depicted in dark red, highlighting their poor performance under the egalitarian rule.

The discrepancy between egalitarian and plurality voting systems is also highlighted in Table 4.1. It provides a detailed breakdown of the 10 winning candidates under the

Egalitarian rank	Candidate ID	Plurality votes
1	1813	44
2	1800	31
3	107	74
4	1109	20
5	3066	36
6	3572	10
7	1102	18
8	2415	31
9	694	38
10	1144	13

Table 4.1: Top-10 winning candidates under the egalitarian rule. The third column reports the number of votes gained under the plurality voting system.

egalitarian rule, along with the corresponding number of votes they received under the plurality system.

None of the candidates listed managed to secure a position in the top-10 rankings of both the utilitarian and plurality systems. This stark contrast underscores the significant differences in candidate performance across the two voting methodologies.

Moreover, their number of plurality votes falls within the range of 10 to 80. These numbers are surprisingly small, taking into account that the plurality winner gets over 6000 votes.

Overall, the comparison drawn from Figure 4.2 and Table 4.1 effectively demonstrates the disparities between the egalitarian and plurality voting systems, emphasizing the differences in outcomes and the potential impact on candidate success.

Chapter 5

Two-Round Voting Systems

The two-round system is an electoral system that involves two rounds of voting to determine a single winner. In the first round, the voting process is equal to the plurality voting system depicted in Chapter 3. A second round of voting, also called cutoff, is held between the top-k candidates who received the most votes in the first round, where k is an arbitrary value defining the number of candidates who can participate in the second round. The winner of the second round is then declared the overall winner.

By allowing the top candidates to compete in a second round, the system provides an opportunity for voters to consolidate their support behind one of the remaining contenders. This can lead to a clearer expression of the voters' preferences and a winner who has broader support among the electorate.

When simulating this voting system in this work, we make the theoretical assumption that the number of voters in the first and the second round is the same, which is not always the case in real-world scenarios.

5.1 Two-Candidates Ballot

In this section, we analyze the case of $k = 2$: the second ballot takes place between the two most-voted candidates of the plurality voting rule. When we simulate this electoral system using the complete smartvote dataset, the result is surprising: Candidate 2325, who was the second-place candidate in the first round of voting (according to Table 3.1), manages to secure victory in the election, despite having received, in the first round, 40% fewer votes than the first-place candidate, Candidate 3884. After the ballot, 229,364 voters vote for Candidate 2325 in the second round, while Candidate 3884 only secures 198,208 votes. Examining further Figure 1.1b and Figure 3.2, it becomes evident that some immediate considerations are necessary to fully comprehend the outcome in this scenario. Intuitively, one can deduce that the shift in the results is attributed to the convergence of votes towards Candidate 2325 in the second round, which they had previously split with several neighboring opponents. As Candidate 2325's positioning on the political compass is further to the left compared to its counterpart, it can attract a

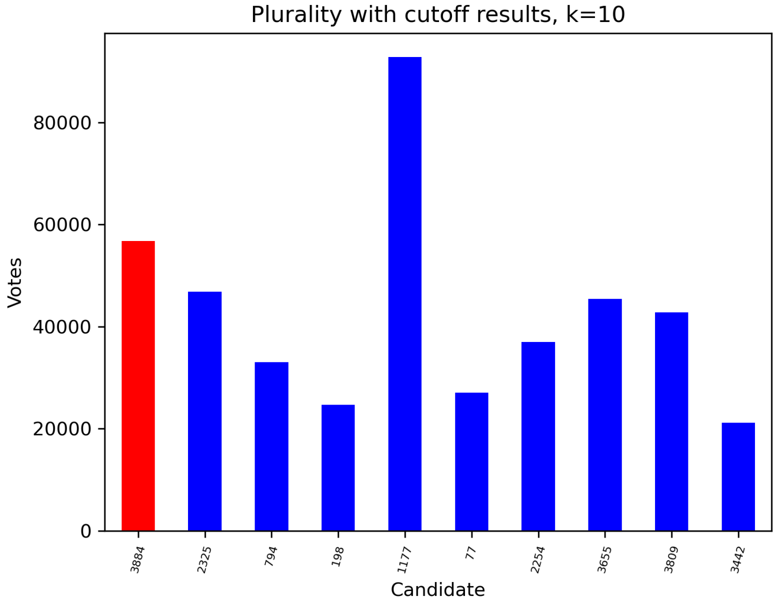


Figure 5.1: Results of the second round with $k = 10$ candidates. In red is the winner of the first round. The first-round candidates' ordering is preserved.

larger number of votes from the smartvote dataset.

It is interesting to notice that the distortion value decreased after the second ballot. The initial distortion value of 1.04, representing the plurality election with Candidate 3884 as the winner, decreases to 1.03 when Candidate 2325 emerges as the winner. This indicates that employing a second round of votes by selecting the top-k candidates can potentially lead to a reduction in the distortion value.

5.2 10 Candidates Ballot

In this section, we analyze the case of $k = 10$: the second ballot takes place between the ten most-voted candidates of the plurality voting rule. Figure 5.1 illustrates that the winner of this ballot is Candidate 1177, who receives the highest number of votes among all candidates. Interestingly, Candidate 1177 achieves this victory despite initially securing only fifth place in the first round. As Candidate 1177 is also the utilitarian winner in the global election, the distortion value in this scenario reaches the optimal value of 1: the highest overall satisfaction of the population is achieved. This result highlights the significance of carefully determining the ballot size to elect a winner who maximizes population satisfaction. If this section shows that an increase in the value of k may be beneficial, in the upcoming section we will examine the potential consequences of an uncontrolled increase in the value of k within the electoral system.

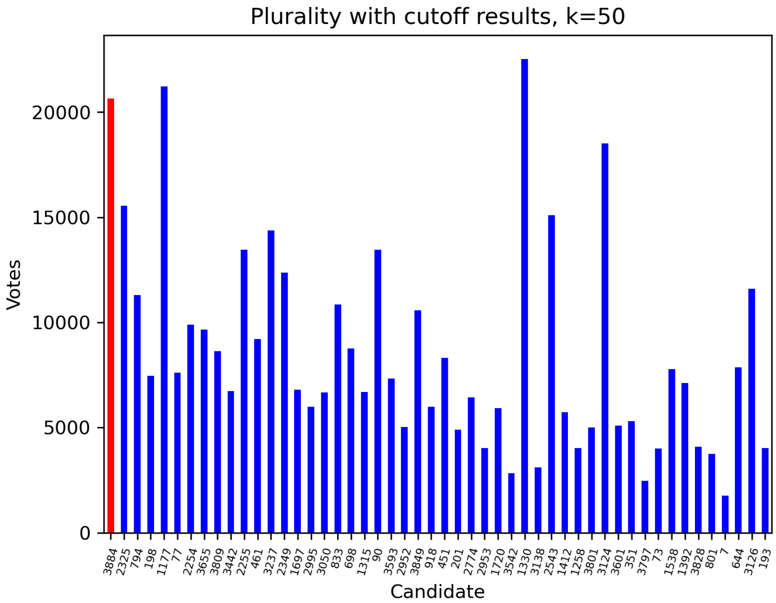


Figure 5.2: Results of the second round with $k = 50$ candidates. In red is the winner of the first round. The first-round candidates' ordering is preserved.

5.3 50 Candidates Ballot

In this section, we analyze the case of $k = 50$: the second ballot takes place between the fifty most-voted candidates of the plurality voting rule. In this case, the value of k includes candidates who in the first round got more than 1,000 votes (Candidate 193, the 50th most voted, got 998). Figure 5.2 reports the candidates in the order in which they entered the ballot. The winner after the second round is, surprisingly, Candidate 1330, ranked 31st before the ballot, and the distortion value is greater than 1.

This result can be due to the fact that Candidate 1330 and Candidate 1177 are close to each other in terms of distance to the voters. Letting Candidate 1330 compete in the ballot, Candidate 1177's votes get divided between the two.

This section also highlights the potential unpredictability of a system when a large number of candidates are allowed to participate in the ballot. It is therefore important to carefully decide how many candidates to allow on the ballot.

Chapter 6

Instant Runoff Voting System

The instant runoff is an electoral system that allows the voters to casts a single vote in the form of a ranked-choice ballot. Under such a ballot, instead of selecting just one candidate, voters can rank multiple candidates according to their personal preferences.

After the first round of the plurality voting system, an arbitrary amount k of candidates, who received a number of votes below a certain threshold, are eliminated from the pool. Their votes are then redistributed according to the voters' rankings.

In a single-winner election, we repeat the process of candidate elimination and vote redistribution until all candidates except the winner are eliminated. After the initial steps, the remaining candidates undergo a series of eliminations. In each iteration of this voting system, the candidate with the fewest preferences is eliminated from the pool, and their votes are subsequently redistributed based on the rankings provided by the voters. In the case of ties, we randomly choose which candidate is eliminated.

This process continues until only one candidate remains, which will be the winner.

In the context of smartvote, given the high computational costs of such simulation, we restrict the voters' pool to 100,000 random voters (approximately 25% of the entire pool). This choice does not influence the reliability of the results, as we show in Chapter 8.

Moreover, we choose to limit the simulation to 900 candidates in the elimination rounds. This choice is given from the fact that most of the candidates who ranked above the 900th position did not reach the minimum amount of votes which would allow them to compete for the win.

We simulate the instant runoff electoral system in the following way:

1. 100,000 voters are selected randomly from the pool.
2. A first round of plurality ranks the candidates per the number of preferences received.
3. We select the top-900 candidates, who will access the elimination steps. All the other candidates are straightforwardly eliminated, and their votes are redistributed

ID	1177	3884	1330	2325	3237	2543	3655	2255	794	2254
Votes-10	12847	12029	**13829**	9119	10145	9842	9808	7336	7717	7328
Votes-9	**15247**	13102	14864	9372	10459	10780	9839	8270	8067	-
Votes-8	**17488**	13587	15255	12364	11302	11429	10271	8304	-	-
Votes-7	**19687**	16357	16131	12824	11499	12849	10653	-	-	-
Votes-6	20137	**22266**	16145	14962	13503	12987	-	-	-	-
Votes-5	23840	22986	**23855**	15739	13580	-	-	-	-	-
Votes-4	25006	**29097**	23900	21997	-	-	-	-	-	-
Votes-3	33349	**41759**	24892	-	-	-	-	-	-	-
Votes-2	**56666**	43334	-	-	-	-	-	-	-	-
Votes-1	**100000**	-	-	-	-	-	-	-	-	-

Table 6.1: Series of snapshots of the evolution of instant runoff with 900 candidates participating in the elimination rounds. In bold is highlighted the candidate who got the most votes at that exact snapshot. Votes-k represents the number of votes received by each candidate when k candidates are remaining. Displayed on the table are the last 10 candidates.

to the remaining candidates according to the voters' preference lists.

4. We eliminate the candidate with the least amount of votes and redistribute their votes to the remaining candidates according to the voters' preference lists.

5. Repeat Step 4 until all candidates except one are eliminated.

6. The last-standing candidate is elected as the winner.

Table 6.1 reports the last 10 snapshots of the instant runoff electoral system with 900 candidates participating in the elimination rounds. When 10 candidates remain, the winner is Candidate 1330, with one thousand votes more than the final winner, Candidate 1177. From 9 to 6 candidates remaining, the winner is Candidate 1177, while after the elimination of Candidate 3655, Candidate 3884 takes the lead.

With only 3 candidates remaining, we notice that Candidate 3884 has the most votes by a strong margin of more than 8000 votes. However, most of the votes of Candidate 1330, who gets eliminated, converge to Candidate 1177, who wins the election.

These analyses allow us to discover similarities within the candidates, such as the one between Candidates 1177 and 1330, already highlighted in the two-round systems (in particular, see Section 5.3).

Four of the ten last-standing candidates, specifically Candidates 1330, 3237, 2543, and 2255, don't show up in the top-10 strictly after the plurality ballot (cfr. Table 3.1) but manage to climb the rankings in the elimination rounds.

Although the final winner is the same, six of the ten last-standing candidates are not in the top-10 under the utilitarian rule. In the end, the instant runoff voting system

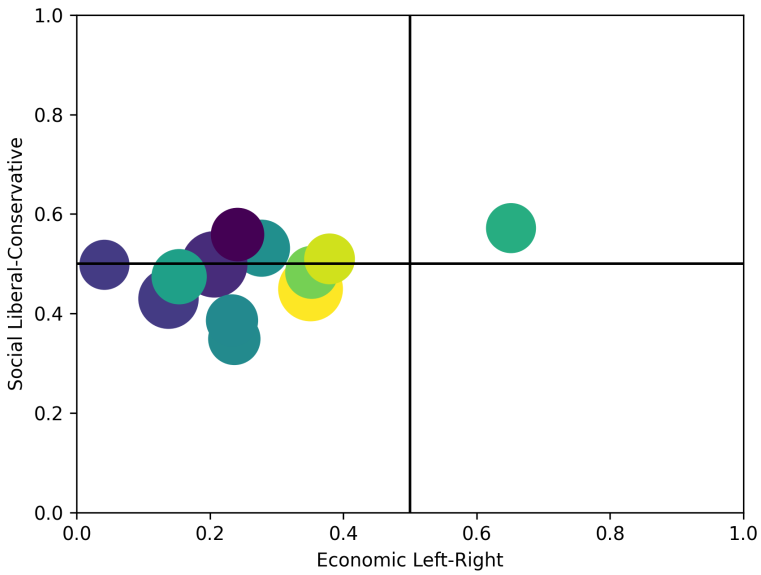


Figure 6.1: Snapshot of instant runoff voting system with 12 candidates remaining. This is the last iteration that includes a candidate coming from the right political area.

converges to the utilitarian result.

As in the case of plurality, all of the ten final candidates come from the center-left political area. To find a candidate from another area, we need to delve into the iteration with 12 candidates left, reported in green in Figure 6.1. In this snapshot, Candidate 2723, whose political compass coordinates lie in the first quadrant, obtains 6660 votes (6.7% of the total) before getting eliminated. This candidate, among the strongest coming from that political area, ranked second in the plurality voting system in Canton Glarus but did not rank in the first three positions in any of the other cantons.

In conclusion, the instant runoff voting system yields similar results to the other electoral systems simulated. The winner and the runner-up outlined by this method are respectively the utilitarian winner and the plurality winner. On the other hand, strong differences between the latter two methods are found in which candidates manage to get to the top-10.

Chapter 7

Plurality Veto Voting System

Plurality veto is an innovative electoral system, which has been proposed by Kizilkaya and Kempe in 2022 [4]. Under this system, each candidate starts with a score equal to the number of first-place votes they receive in a first plurality round.

A veto process then takes place over n rounds, where in each round, voters in a random order decrement the score of their least favored candidate among those still standing. A candidate is eliminated when their score reaches zero. The last standing candidate wins. This rule requires only two queries to each voter.

It has been shown that plurality veto is guaranteed to achieve metric distortion of at most 3. However, its outcome depends on the order in which voters are queried, making it challenging to study its axiomatic properties. As the results will show, in the context of smartvote, actual distortion appears to be smaller.

In this work, we simulate the plurality veto voting system in a couple of ways: first, we allow each candidate to participate in the veto round. Then, we limit the access to the ballot to 10 and 4 candidates. In all of the implementations, the order of the voters is randomized.

7.1 All Candidates Participate in the Veto Rounds

When allowing every candidate to participate in the veto rounds, the winner is Candidate 1177, who doesn't immediately result in the top candidate (after the plurality round Candidate 1177 has fewer votes than Candidates 3884, 2325, and some others, who didn't manage to get to the last 10).

Allowing all the candidates to participate in the veto rounds, we get the same winner as for the utilitarian rule. It is surprising to note that the candidate who initially received the most votes in the plurality round (Candidate 3884) did not even make it to the top 9 candidates in the final round. Some snapshots of the evolution of the veto round are given in Table 7.1.

ID	1177	77	2325	2254	2774	1258	3593	1315	3809	3884
Score-all	2872	2632	3758	2273	1339	1171	1548	1531	2039	**6740**
Score-1000	2861	2555	3758	2272	1339	1171	1548	1531	2037	**6736**
Score-500	2859	2421	3758	2270	1339	1171	1548	1531	2033	**6733**
Score-100	2852	2216	3744	2263	1332	1168	1544	1528	2018	**6548**
Score-50	2852	2182	3627	2261	1308	1154	1532	1469	1968	**5726**
Score-30	2852	2146	3369	2235	1251	1109	1478	1323	1807	**4131**
Score-20	2833	2024	**2950**	2018	1103	881	1256	1017	1140	1982
Score-10	**2803**	1890	2430	1657	925	616	842	472	169	157
Score-9	**2788**	1866	2365	1588	891	578	775	406	22	-
Score-8	**2787**	1864	2352	1578	886	571	766	394	-	-
Score-7	**2702**	1750	1941	1173	656	277	276	-	-	-
Score-6	**2666**	1700	1727	993	521	124	-	-	-	-
Score-5	**2622**	1645	1608	858	401	-	-	-	-	-
Score-4	**2499**	1443	1113	319	-	-	-	-	-	-
Score-3	**2401**	1285	769	-	-	-	-	-	-	-
Score-2	**1953**	696	-	-	-	-	-	-	-	-
Score-1	**1501**	-	-	-	-	-	-	-	-	-

Table 7.1: Series of snapshots of the evolution of plurality veto with all candidates participating to the ballot. In bold is highlighted the candidate who got the most votes at that exact snapshot. Score-k represents the score of each candidate when k candidates are remaining. Displayed on the table are the last 10 candidates who get eliminated.

ID	1177	2325	3884	77	2254	794	3442	3809	198	3655
Score-9	**90603**	45264	56874	24222	25178	22513	14927	13578	8480	-
Score-8	**89921**	44602	55735	23360	22324	19189	10311	6483	-	-
Score-7	**89344**	43725	52819	22509	19889	15898	4526	-	-	-
Score-6	**88342**	43017	50492	21157	16246	12696	-	-	-	-
Score-5	**85006**	38572	37012	16347	3684	-	-	-	-	-
Score-4	**83705**	36040	33211	14731	-	-	-	-	-	-
Score-3	**69192**	22078	10983	-	-	-	-	-	-	-
Score-2	**59202**	15185	-	-	-	-	-	-	-	-
Score-1	**46648**	-	-	-	-	-	-	-	-	-

Table 7.2: Series of snapshots of the evolution of plurality veto with 10 candidates participating to the ballot. In bold is highlighted the candidate who got the most votes at that exact snapshot. Score-k represents the score of each candidate when k candidates are remaining. Displayed on the table are the last 10 candidates who get eliminated.

7.2 Ballot with 10 Candidates

In this case, we deal with a veto ballot containing only the top-10 candidates defined after the plurality round. Candidate 1177 secures the win from the beginning of the ballot, with a strong margin of votes, Candidate 1177 was once again determined to be the winner, as shown in Table 7.2. However, the order in which candidates were eliminated differed significantly from the previous ballot, where all candidates were considered. Notably, candidate 3884, who was eliminated early on in the previous ballot, managed to get to the third position when facing only 9 other candidates. On the other hand, candidate 77 lost two positions in this smaller ballot.

These outcomes emphasize the impact of the number of candidates on the voting process and the importance of carefully considering how many candidates are included in the ballot.

7.3 Ballot with 4 Candidates

When limiting the veto rounds to only the top-4 candidates, an interesting phenomenon occurs: the winner of the Plurality veto system is placed third out of the four candidates in the ballot, as shown in Table 7.3. Despite starting with more than 25,000 fewer votes than the top two candidates, candidate 2325 manages to win these rounds. This is likely due to the fact that the two candidates with more votes are more polarizing, resulting in a larger number of voters who do not favor them.

In contrast, candidate 2325 may not be the favorite of the majority of voters, but they are also not the least favorite of many voters. This moderate level of approval allows them to accumulate enough support from voters who prefer them as an acceptable compromise,

ID	2325	77	3884	794
Score-4	74411	**106302**	100325	48743
Score-3	**59212**	69118	55365	-
Score-2	**20167**	10396	-	-
Score-1	**10247**	-	-	-

Table 7.3: Series of snapshots of the evolution of plurality veto with 4 candidates participating to the ballot. In bold is highlighted the candidate who got the most votes at that exact snapshot. Score-k represents the score of each candidate when k candidates are remaining. Displayed on the table are the last 4 candidates who get eliminated.

rather than a polarizing choice. This demonstrates how a candidate's level of polarizing appeal can affect their success in the voting process.

7.4 Considerations About Distortion

Plurality veto applied to the smartvote dataset seems to yield optimal results in terms of distortion whenever allowing Candidate 1177, the utilitarian winner, to participate in the ballot. In contrast to the two-round election, presented in Section 5, there are no downsides with respect to distortion if we increase the number of candidates allowed to the veto rounds. The metric distortion of plurality veto applied to the smartvote dataset is therefore 1.

Once again, most of the ballots (all except top-4) do converge to Candidate 1177, the utilitarian winner.

Chapter 8

Sortition Voting System

This chapter offers an analysis of the "sortition" voting system applied to the smartvote dataset. This framework works as follows: we select a random sample of voters and assign them the task of voting on behalf of the entire population.

The ultimate aim of this section is to find a solution to the following question: what is, if it exists, the minimum percentage of people we must compel to participate in voting, to ensure that the sample accurately represents the entire population? Moreover, are there any restrictions to how the sample needs to be chosen?

More formally, given a set of voters V and an electoral system, let $p \in [0, 1]$ be the fraction of people who actually vote. Our task is to find the minimum value p^* of p which is required to vote for the sample to be representative. In this context, *representative* means that the winner of the election is the same as the winner of the election if the whole population votes. To do that, for each value of p we randomly select p percent of the given population 100 times and compute the winner of the given electoral system with this subset of voters. Then, we have that a subset is a representative as soon as the following requirements are fulfilled:

1. With p fixed, the winner w^* has to be the same for each sampling.
2. Assuming the above 1, w^* has to be the same of the winner with $p = 1$.

For practical reasons, it is useful to reformulate 1 and 2 as follows.

Definition 8.1 (p^*)**.** Given an electoral system, let V be a set of voters, $p \in [0, 1]$ be the fraction of elements of V selected to go to vote. Run the election t times for each value of p and store the different winners in the set W_p. Moreover, denote as w^* the final winner. Then p^* is defined as follows:

$$p^* = \text{argmin}_{0 \leq p \leq 1} \{W_p = \{w^*\}, \quad \forall p \in [p, 1]\}. \tag{8.1}$$

It should be noted that the model described above is retrospective, in the sense that the exact value of p can only be computed after the winner of the whole given population is known. However, the analyses presented in this section offer a useful approach for

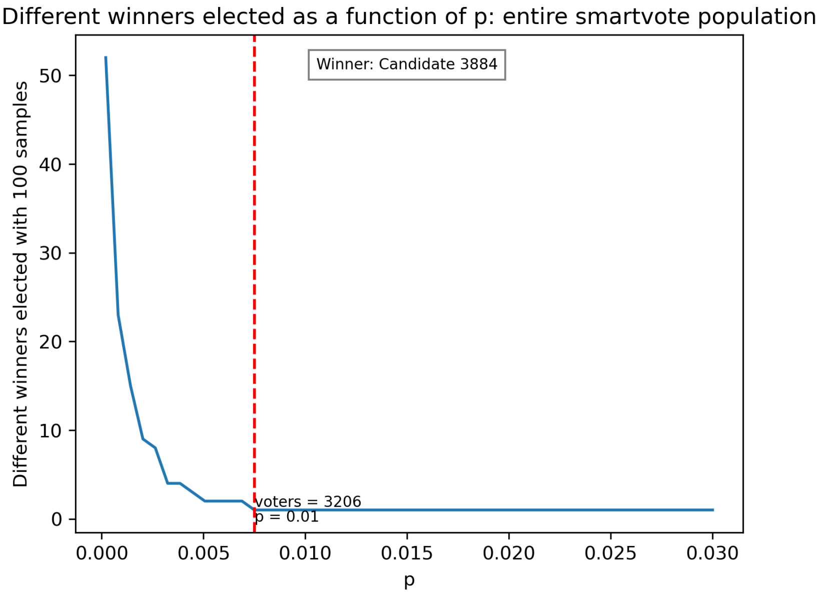


Figure 8.1: Amount of unique winners elected varying p over the smartvote dataset. The red line highlights the minimum number of voters and the percentage of the whole population needed to get a reliable result. Every test was run 100 times.

estimating the minimum fraction of people required to obtain a representative sample for future elections.

8.1 Global Plurality

For the plurality voting system given the smartvote dataset, the results are summarized in Figure 8.1. The red vertical line highlights the p for which Definition 8.1 holds. Less than 1% of the voters are needed in order to make the voting session representative. Specifically, we found an exact value of 3206 voters, which corresponds to 0.75% of the total population.

We can make some observations and comments based on this result. First, the fact that such a small percentage of voters is needed for the sample to be representative suggests that the winner of the election does not heavily depend on the preferences of a specific group of people. This may lead to a state where the voting population is relatively homogeneous in terms of political views. This means that a small fraction of the population can accurately represent the views of the larger group. In a hypothetical scenario in which the sortition voting system is implemented, this result could have important implications for the cost and logistics of holding elections.

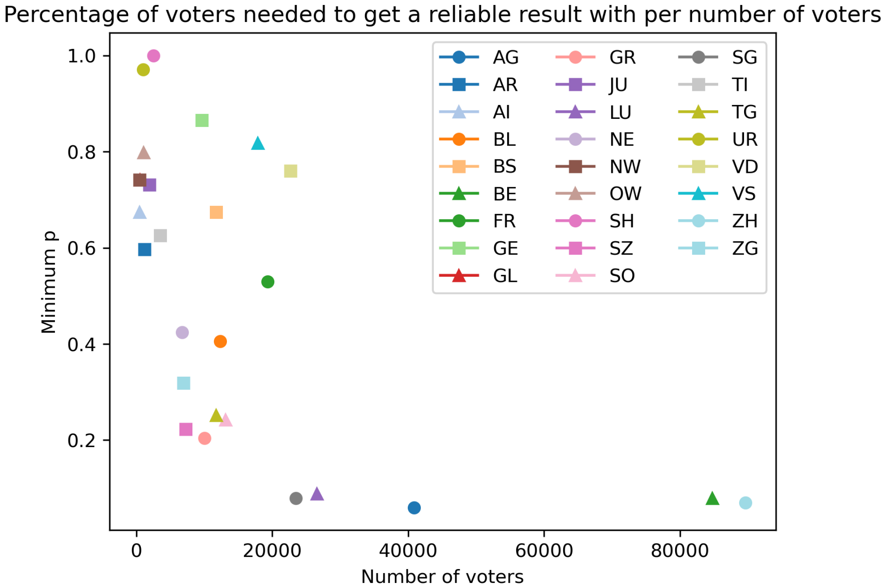


Figure 8.2: Canton plurality p^* values against canton sizes. For big cantons, the value stabilizes at less than 10%. For small cantons, the behavior is undefined.

8.2 Canton Plurality

The same computations can be done for each Swiss canton and the results are interesting to understand how the canton size affects the results and how homogeneous a specific canton is. To summarize the obtained results, which are reported in Figures F.1 to F.4, we refer to Figure 8.2.

The graph shows a clear trend for cantons with populations of greater than 25,000, where a fraction of the population of approximately 0.1 is required to obtain meaningful results. This is intuitively correct since a bigger population makes the random sampling more accurate and reliable. It is interesting to notice that for small cantons (less than 25,000 voters) the results are less straightforward. This could be due to the fact that random samplings of smaller populations are more heterogeneous and people-dependent, which makes it more difficult to obtain a representative sample.

This result shows that the sortition voting system is effective for populations of more than 25,000 voters. In such cases, a small fraction of voters are required to get a reliable result. In the context of smartvote, Aargau, Bern, Luzern, St. Gallen and Zürich are the cantons that would actually benefit from adopting sortition as an electoral system.

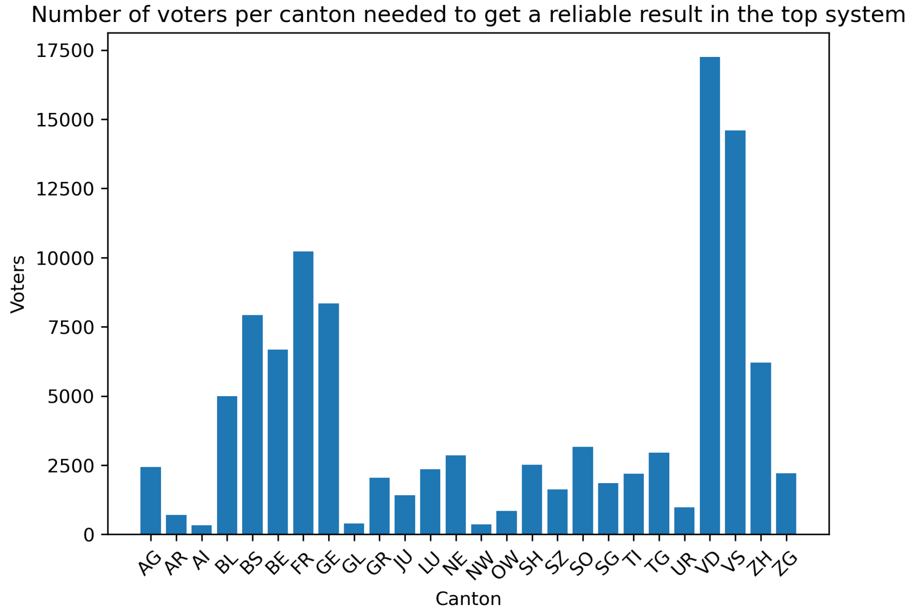


Figure 8.3: Absolute values of voters needed to get a reliable result per canton. Cantons are indexed alphabetically.

8.3 Absolute Values

As described above, it appears that the reliability of results in the sortition voting system is not solely dependent on a specific percentage of voters but rather on the specific circumstances of each canton: small cantons seem to be unpredictable. While there is no precise value p^*, that guarantees a reliable outcome, there are indications that certain cantons require a smaller number of voters to achieve reliable results. The data presented in Figure 8.3 suggests that in 18 out of 26 cantons, reliable results can be obtained with fewer than 2500 voters.

Further examination of the remaining eight cantons reveals that the higher number of voters required (up to 17,000 for the case of Canton Waadt) is often associated with a close tie between two candidates. For example, Figure 8.4 reports the results for Canton Wallis, which necessitates the second highest number of voters (see canton 24 in Figure 8.3). After 20% of the voters are required to vote, there seems to be a close tie between two candidates, which is only decided after 80% of voters cast their preferences. Similar ties occur in the other 7 outliers of Figure 8.3.

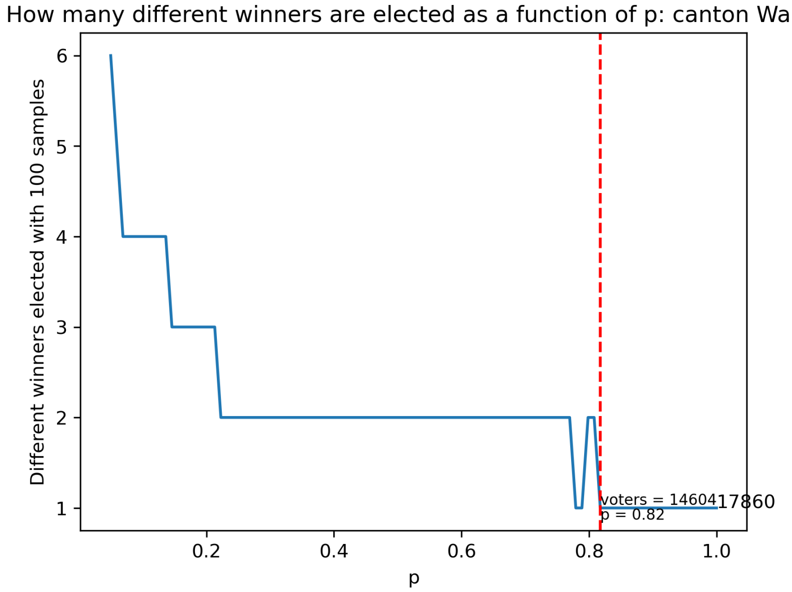


Figure 8.4: Amount of unique winners elected varying p over Canton Wallis smartvote dataset. After $p = 0.2$, there seems to be a close tie between the two candidates. 80% of voters are needed to elect the final winner.

8.4 Conclusions

We opened this chapter with a question: what is, if it exists, the minimum percentage or number of people we must compel to participate in voting, to ensure that the sample accurately represents the entire population?

On a national level, results show that only less than 1% of the population is needed to declare a unique winner. The amount of voters needed to elect Candidate 3884 is 3206. This is a strong result that underlines how homogeneous the smartvote population is. On the cantonal level, we can conclude that generally, with big cantons (with more than 25,000 voters), one requires a percentage of less than 10% to reliably elect a unique winner.

When discussing the actual amount of voters needed, results show that, regardless of the size of the cantons, a number of voters in the order of the *few-thousands* are needed to get a reliable result in the sortition voting system. In cases with two strong candidates who are close to each other, a tie could raise the number of voters needed up to 80% of the total population. These cases outline that in cantons of less than 25,000 people, one needs to carefully consider the candidates' pool when implementing this voting system.

To conclude, it is clear that these results are purely theoretical. The big real-world obstacle to actually implementing the sortition voting system is selecting a true *random* sample of the population.

Chapter 9

Candidate Comparison-Based Election Trees

In the previous sections, we constructed elections based on users' responses to the 75 questions posed in the smartvote questionnaire. Now, our goal is to explore the possibility of building the same elections while minimizing the number of variables necessary to define the profiles. We assume that each voter is aware of their profile, and we want to determine their complete profile by asking the fewest number of questions. We limit the questions to be of the form "Do you prefer candidate A or candidate B?". To effectively model this experiment, we can use decision trees. In particular, we define Candidate Comparison-Based Election Trees (CCBET).

9.1 Definition and Construction

Consider an election E with m candidates and n^* voters. To construct a Candidate Comparison-Based Election Tree (CCBET), we begin by merging voters who have the same voting profiles. This merging process helps us identify the number of *unique* voters, denoted as n. We define the set of candidates as $C = \{c_1, c_2, \ldots, c_m\}$ and the set of unique voters' profiles as $V = \{v_1, v_2, \ldots, v_n\}$. Each voter's profile is a permutation of the candidates in C, where the position of each candidate in the sequence represents its ranking in that voter's profile. We will now describe the construction of a Candidate Comparison-Based Election Tree.

In a CCBET, each node represents a subset of the set V. If a subset contains only one element, it is considered a leaf node representing a unique voter profile. If a subset contains more than one element, it splits into two children nodes based on the comparison of two candidates' positions in the permutations.

The construction of the tree proceeds as follows:

1. Assign to the root node the entire set of unique voters V.

2. If the current node's subset contains only one element, consider it as a leaf node

representing a unique voter profile and stop the algorithm. Otherwise, proceed with the next steps.

3. If the current node's subset of voters contains more than one element, select a pair of candidates (c_i, c_j) that has not been selected in any previous iteration. This selection must satisfy the condition that there exists at least one pair of unique voters who rank c_i and c_j differently (at least one voter prefers c_i over c_j and at least one voter prefers c_j over c_i). For example, if c_1 is ranked above c_2 in some voter profiles and ranked below c_2 in other profiles, we can select (c_1, c_2) as the candidate pair.

4. Split the current node's subset into two subsets:
   - Subset A: Contains the elements from the current node's subset where c_i is positioned before c_j in the permutations.
   - Subset B: Contains the elements from the current node's subset where c_j is positioned before c_i in the permutations.

5. Create two child nodes for the current node, assigning Subset A to one child and Subset B to the other.

6. Recursively repeat steps 2 to 5 for each child node until all leaf nodes representing unique voter profiles are reached.

At the end of this process, we will have constructed a Candidate Comparison-Based Election Tree. Each leaf node of the tree represents a unique voter profile, and the path from the root node to each leaf node represents the comparison of candidate positions in the permutations. This tree provides a useful framework for analyzing and understanding the relationships between different voter profiles based on the positions of candidates in their permutations.

9.2 Example

To illustrate this process more clearly, let's consider the following example.

Consider an election E with $m = 4$ candidates and $n^* = 8$ voters. We define $C = \{\mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{d}\}$ as the set of candidates and $V^* = \{v_1^*, \ldots, v_8^*\}$ as the set of voters. Table 9.1 illustrates the process of merging voters into unique voters. In the left table, one can observe that voters v_3^* and v_4^* have the same profile (a,c,b,d). Therefore, we merge these voters into a single unique voter, denoted as v_3 in the right table. Moreover, voters v_5^*, v_7^* and v_8^* also share the same profile (d,a,c,b), and are merged into the unique voter v_4. The resulting election consists of $m = 4$ candidates and $n = 5$ unique voters.

To construct a CCBET for the given election, we follow the procedure described in the previous section. The root node of the tree represents the entire set of distinct voters

	Profiles			
v_1^*	a	b	c	d
v_2^*	b	a	d	c
v_3^*	a	c	b	d
v_4^*	a	c	b	d
v_5^*	d	a	c	b
v_6^*	c	d	b	a
v_7^*	d	a	c	b
v_8^*	d	a	c	b

$\longrightarrow$

	Unique profiles			
v_1	a	b	c	d
v_2	b	a	d	c
v_3	a	c	b	d
v_4	d	a	c	b
v_5	c	d	b	a

Table 9.1: Example of voters' profiles set and unique voters' profiles set. Voters v_3^* and v_4^*, having the same profiles, have been merged into v_3. Voters v_5^*, v_7^*, v_8^*, having the same profiles, have been merged into v_4.

V. During each iteration of the algorithm, we partition the subsets of voters according to the preference rankings between two candidates. In this particular instance, the *splitting candidates* are selected randomly. However, it is worth noting that this approach may not always yield optimal results in terms of minimization of the tree height. We will delve into this topic further in Chapter 9.3, where we explore the concept of tree height optimization.

Let's illustrate the construction of the CCBET for the election, shown in Figure 9.1 (note that voters are represented with numbers from 1 to 5, where voter v_i is represented with number i). The first step involves assigning the set of unique voters to the root node of the CCBET.

We will now proceed with the iterative steps, following steps 2 to 5 (as described in the previous section) for each node until all leaf nodes are reached. Here is a breakdown of the iterations:

- First iteration: Split based on the relative positions of randomly-chosen candidates b and c. The left node will include voters v_1 and v_2 since b ranks higher than c in the respective profiles. The right node will contain the remaining voters, v_3, v_4, and v_5, where candidate c is ranked higher than candidate b.

- Second iteration: Split based on the relative positions of candidates d and c for the left part of the tree and the relative positions of candidates a and b for the right part of the tree. For the left node, we split the voters' set based on the ranking of d and c. For the right node, we split the voters' set based on the ranking of a and b.

- Third iteration: At this level, three out of the four nodes are leaf nodes, representing a unique voter profile. We only need to work on the remaining non-leaf node, which contains voters 4 and 5. We split this node with respect to the candidates d and a, resulting in two new leaf nodes.

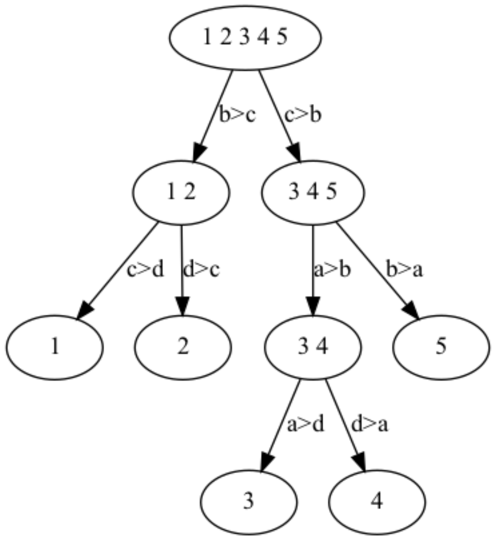


Figure 9.1: Example CCBET with splitting candidates randomly selected at each step. Voters are represented with numbers from 1 to 5, where voter v_i is represented with number i. a > b occurs whenever candidate a ranks before candidate b in a profile.

- Fourth iteration: At this stage, every path in the CCBET reaches a leaf node, representing a unique voter profile. The algorithm terminates.

With these iterations, the construction of the CCBET is complete, and we have a tree that fully represents the election scenario.

9.3 Height Minimization

Asking fewer questions in the process of building the CCBET for an election has several benefits, since it reduces potential frustration or fatigue that voters may experience when faced with numerous questions. By ensuring a low number of required questions, we prioritize efficiency and create a valuable tool for accurately describing the election. Therefore, in the context of a CCBET, our primary goal is to devise an algorithm that constructs a tree with the shortest possible height, where the height of a tree is defined as the length of the longest path from the root of the tree to its farthest node.

9.3.1 In Theory

When considering the set of all permutations of m elements, the optimal height of a comparison-based binary tree is given by $\lceil \log_2 (m!) \rceil$. This height represents the minimum number of comparisons required to sort the permutations using a binary tree structure. While it is true that this binary tree is perfectly balanced, constructing a perfectly

	Profiles					
v_1	a	b	c	d	e	f
v_2	a	b	c	d	f	e
v_3	a	b	c	e	d	f
v_4	a	b	d	c	e	f
v_5	a	c	b	d	e	f
v_6	b	a	c	d	e	f

Table 9.2: With this set of profiles the CCBET height is n. This is the worst case for an election with an equal number of candidates and unique voters.

balanced binary tree that divides permutations into two subsets of the same cardinality at each level is not always possible, as we will see in this section.

In the study of different types of elections related to the construction of a CCBET, two edge cases can be considered: one where the number of unique voters n is equal to the number of candidates m, and another where the number of voters is equal to the factorial of the number of candidates $n = m!$. In the latter case, all of the permutations are present in the voters' profiles.

In the case where the profiles include *all* possible permutations of m elements ($n = m!$), comparison-based sorting algorithms can be employed. It is well known that any deterministic comparison-based sorting algorithm requires $\log_2 m!$ comparisons to sort n elements in the worst case. To simplify the expression, we can apply Stirling's approximation, which approximates the factorial function. We thus obtain $\mathcal{O}(\log_2 m!) \approx \mathcal{O}(m \log_2 m)$.

On the other hand, we don't observe the same behavior in the case where $n = m$. With this setting, the worst-case scenario occurs if you let V be the set of permutations that switch any two adjacent elements. Table 9.2 illustrates an example of the case with $n = m = 6$, where $C = \{\mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{d}, \mathrm{e}, \mathrm{f}\}$ and $V = \{v_1, \ldots, v_6\}$.

This leads to a situation where the height of the CCBET tree, representing the comparison-based sorting process, is equal to n. The number of possible permutations that switch any two adjacent elements is $n = m - 1$. However, for any pair of candidates, at least $m - 2$ permutations preserve their order. As a result, the binary tree constructed for this election needs to have a height of $n - 1$, as each comparison can eliminate at most one permutation in the worst case. Therefore, the complexity of this case is $\mathcal{O}(n)$. For all elections with $n < m$, the same result is trivially true. We can therefore conclude that the height of a CCBET is not in $\mathcal{O}(\log_2 n)$, for $n \leq m$. Moreover, for this specific example, we can conclude that it is not possible to create a balanced binary tree that divides permutations into two subsets of the same cardinality at each level.

In conclusion, we can state that not every election can be represented with a Candidate Comparison-Based Election Tree of height $\lceil \log_2 (m!) \rceil$. But what about smartvote?

9.3.2 In Practice

The smartvote dataset, with $m < n \ll m!$ unique voters, falls between the above two cases, and it presents a unique problem as no specific theorem exists to address this scenario. The complexity of the sorting process for such a dataset is not immediately evident, and further research is required to develop efficient algorithms or determine the lower and upper bounds for this specific case. The crucial challenge is the selection of the two candidates that will lead to the most effective division of voters at every iteration of the creation of the CCBET (point 3 of subsection 9.1).

To investigate this issue, we compare four different methods:

- **Random:** The two splitting candidates are selected randomly.
- **Best Two:** The two splitting candidates is the winner and runner-up of an imaginary presidential election run with the subset of profiles present in a node at each iteration.
- **Random Top:** The two splitting candidates are randomly selected from the set of winners of each profile present in a node.
- **C1 fixed:** One candidate is randomly selected and fixed throughout the entire algorithm, while a random opponent is chosen at each iteration.

Figure 9.2 illustrates the tree heights of these four methods, and reports in black the theoretical lower bound of $\lceil \log_2(n) \rceil$. None of the methods yield optimal results. However, the Random Top method seems promising, as it achieves results close to an expected complexity of $\mathcal{O}(\log_2(n))$. It would be interesting to consider more tests, to deepen our understanding of the relation between CCBET and smartvote.

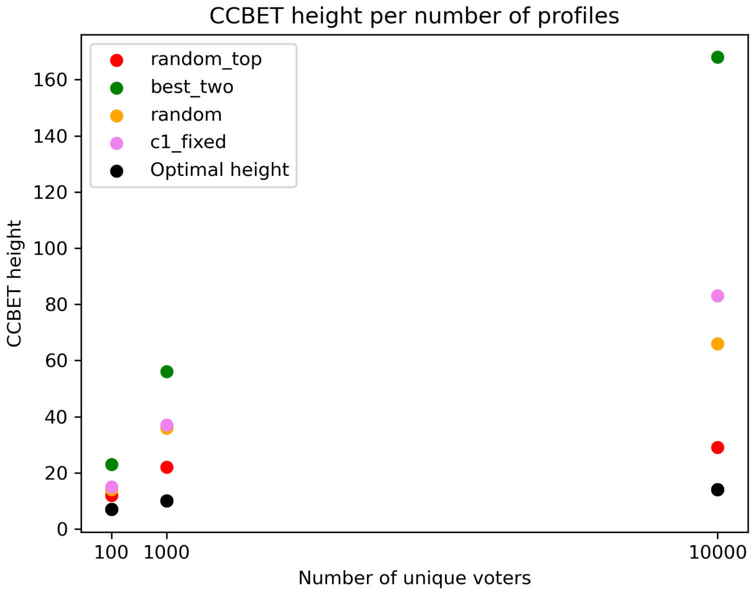


Figure 9.2: Comparison of candidates-selection methods with $n \in \{100, 1000, 10000\}$ unique voters profiles. Optimal sizes are set at $\lceil \log_2 n \rceil$.

Chapter 10

Personalization of the Questionnaire

Survey fatigue refers to the phenomenon wherein individuals experience laziness or a lack of interest when responding to questionnaire questions. It is a well-recognized issue that can impact the quality and reliability of survey data and, in the context of smartvote, can lead to inaccurate voting advice. Prior research (see [1], 22–24, and [5], 8–9) have demonstrated that smartvote is not immune to the effects of survey fatigue: especially in the last 15 questions of the "Deluxe" questionnaire the attention of the users drops, and the number of answered questions drops consequently. This raises concerns about the accuracy of the voting advice provided to users experiencing survey fatigue.

To address this problem and mitigate the potential for incorrect voting advice due to survey fatigue, one immediate solution is to reduce the number of questions. This approach has been implemented by introducing a "Rapid" version of the questionnaire, which consists of only 31 questions. However, as previously stated, survey fatigue primarily affects users who opt for the longer version of the questionnaire.

10.1 The Approach

Although the idea of reducing the number of questions in the long version of the questionnaire is the easiest to tackle survey fatigue, it is important to consider the trade-off involved in reducing the number of questions. While it may alleviate survey fatigue, it also leads to a loss of information, that inevitably leads to less precise voting advice. Consequently, users who do not experience fatigue may be disadvantaged by receiving less accurate recommendations.

To lose the least amount of users' information, our idea is not based on proactively reducing the number of questions, but rather on improving the so-called *convergence of the final profile*. The idea is to personalize the order of questions for each user, ensuring an earlier appearance in the profiles of the candidates who will eventually be featured in the final profile. This way, if a user experiences fatigue and decides to conclude the questionnaire prematurely, the resulting advice is more accurate compared to the standard ordering of questions. This approach implies that if a user decides to stop answering questions earlier than anticipated, the advice generated is still reliable and reflective of their preferences.

The solution we propose is based on the principle of portability: we do want to make it work not only for the 2019 smartvote dataset, which we are working on and which is useful to benchmark our algorithms, but we want to propose an innovative way of building the questionnaire experience for the user only given the answers of the candidates. In this way, future questionnaires can be optimized with our algorithm. For this reason, the algorithm only relies on the answers of the candidates.

10.2 The Highest-Variance Question Selection Algorithm

The Highest-Variance Question Selection algorithm is a new method designed to enhance the voting advice provided to smartvote users. By personalizing the order of questions for each user, this algorithm aims to deliver more reliable recommendations earlier in the questionnaire process. The algorithm operates in iterations, selecting each subsequent question based on the user's previous answers and the answers of the candidates.

In the initial phase, a random subset of n questions (where $n < 75$) is chosen and asked every user. Once these initial questions have been answered, the algorithm computes a voting profile for each user. The voting profile strongly depends on the user's answers and serves as a basis for subsequent question selection.

Next, from the remaining pool of $75-n$ questions, the algorithm identifies the question that maximizes the variance of answers among the top-10 candidates in each user's profile. This selection process is described in Algorithm 1 and is repeated iteratively until either a user decides to conclude the questionnaire or all questions have been asked.

The choice to focus on the top-10 candidates in each user's profile was made using a greedy approach. The goal was to find a balance between having a sufficiently large number of candidates k to allow for potential rearrangements, but small enough to give weight to the previous answers of the users, which are embedded within the top-k profile. To determine the optimal value for k, several tests were conducted with different values ($k \in \{5, 10, 20, 50\}$), and it was found that $k = 10$ yields the most promising results.

Algorithm 1 Select question with maximum variance

```
function getNextQuestion(topProfile, questionnaire)
    remainingQuestions ← store not already asked questions
    nextQuestion ← -1                                  ▷ Initialize variable nextQuestion
    maxVariance ← -1                                   ▷ Initialize variable maxVariance
    for all question in remainingQuestions do
        variance ← compute variance of answers of question among the candidates
present in topProfile elements
        if variance > maxVariance then
            maxVariance ← variance
            nextQuestion ← question
        end if
    end for
    return nextQuestion
end function
```

Canton	**Number of voters**	**Number of candidates**
Aargau	26173	383
Appenzell Ausserrhoden	653	1
Appenzell Innerrhoden	296	4
Basel-Landschaft	7736	121
Basel-Stadt	7481	116
Bern	54328	554
Freiburg	11730	145
Genf	6257	139
Glarus	305	2
Graubünden	5991	82
Jura	1159	32
Luzern	15777	218
Neuenburg	4143	43
Nidwalden	244	1
Obwalden	526	4
Schaffhausen	1366	26
Schwyz	4290	81
Solothurn	8285	157
St. Gallen	14551	223
Tessin	2311	95
Thurgau	7207	124
Uri	543	3
Waadt	14955	297
Wallis	10943	200
Zürich	58372	799
Zug	4225	63

Table 10.1: Number of voters and candidates per canton. In bold cantons that have been tested.

10.3 Testing

To test the algorithm, it is important to make some considerations. In the previous sections, we simulated hypothetical smartvote presidential elections, which is why we utilized the entire pool of voters and candidates to simulate different voting systems.

However, in this section, our focus is on optimizing the questionnaire itself and enhancing the users' experience. Therefore, we will test our algorithm using the original smartvote cantonal divisions. This means that we will split both the candidates and voters pools according to the respective cantons.

Furthermore, to ensure accurate benchmarking, the voters' pool has been limited to the 269,847 users who completed the "Deluxe" version of the questionnaire: all users who completed the "Rapid" version have been excluded from the benchmarks.

Table 10.1 reports the features of each canton. To ensure accurate testing within our infrastructure limitations, certain cantons have been excluded. Specifically, Appenzell Ausserrhoden, Appenzell Innerrhoden, Glarus, Nidwalden, Obwalden, and Uri have been excluded due to their limited number of candidates (less than 10), which makes it impossible for our algorithm to run correctly. Additionally, Aargau, Bern, Luzern, St. Gallen, and Zürich have been excluded due to their substantial number of voters (more than 15000), which would require an impractical amount of time to process given our current resources.

The remaining cantons, which are of medium size in terms of both the number of voters and candidates, have been subjected to benchmarking. Each test was repeated 100 times for accuracy and reliability.

We run tests to compare three different ordering methods: the original smartvote order, a randomly-ordered questionnaire, and our HVQS algorithm ordering. For the HVQS algorithm, we consider different scenarios where n initial random questions are used, specifically $n \in \{1, 2, 5, 10\}$. These scenarios allow us to analyze the impact of the number of initial random questions on the performance of the algorithm.

We measure two indicators to evaluate the performance of the different questionnaire orderings. The first indicator is the presence of the final winner within the top-10 profiles of a user. This measures how likely it is for the winning candidate to appear in the top 10 choices of a user's profile after a number of questions answered. The second indicator is the evolution of the presence of the final top-5 candidates within the top-10 profiles.

These two indicators do not consider the specific ordering of candidates or the associated similarity scores.

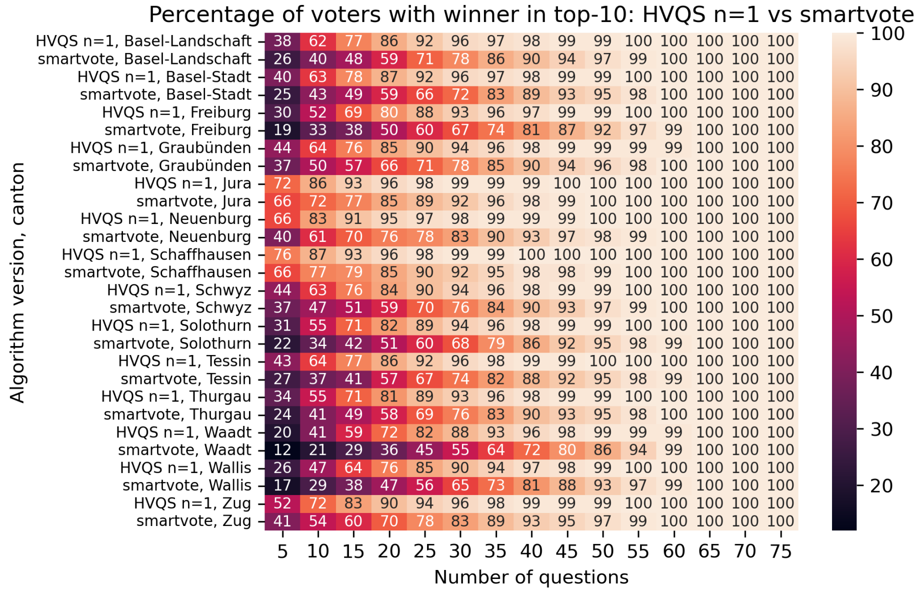


Figure 10.1: Cantonal comparison of HVQS algorithm with $n = 1$ and original smartvote ordering. Percentage of voters with final winner present in top-10 per number of questions.

10.4 Results and Discussion

Consistent results were observed across all cantons, indicating that the Highest Variance Question Selection algorithm with 1 initial random question has the best convergence in terms of both the presence of the winner and the presence of the top-5 candidates within the top-10 profiles throughout the questionnaire.

Figures 10.1 and 10.2 summarize the findings we will discuss in the next subsections and report the results for the HVQS algorithm with $n = 1$ initial random question and for the original smartvote order. The heatmaps show the percentage of voters with respectively the winner and the top-5 candidates present in their top-10 profile with respect to the number of questions answered. Both figures show a clear trend: for all cantons the personalized order performs better and converges faster to the final voting advice.

10.4.1 Final Winner in Top-10

Figure 10.3 shows the relationship between the percentage of voters who have the final winner included in their top-10 profile and the number of questions they have answered. For every canton, the first four rows of the respective heatmap report the HVQS performances with $n \in \{1, 2, 5, 10\}$ initial random question. The fifth row outlines the

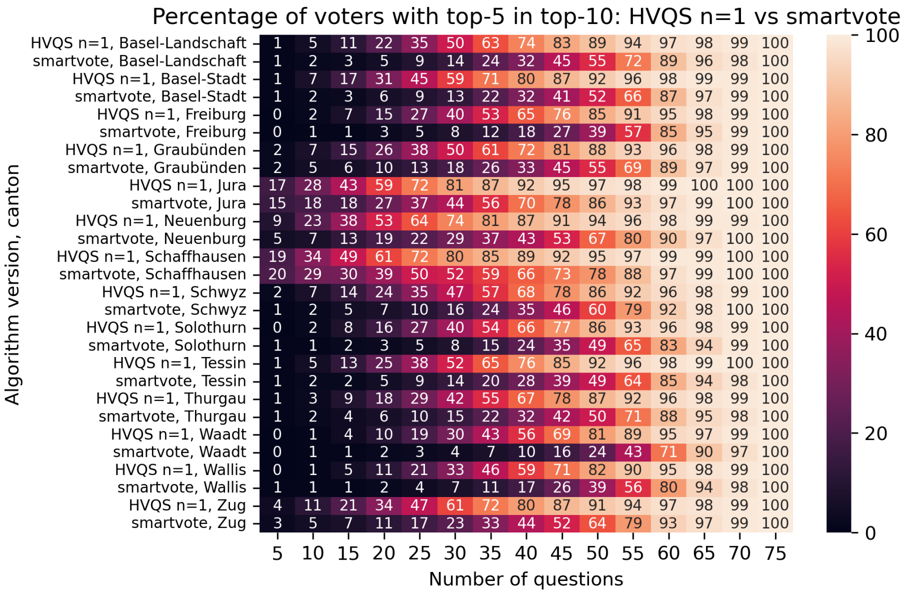


Figure 10.2: Cantonal comparison of HVQS algorithm with $n = 1$ and original smartvote ordering. Percentage of voters with final top-5 candidates present in top-10 per number of questions.

performance of a questionnaire where questions are ordered randomly and the last line reports the original smartvote ordering performance.

In all of the cantons, using a random order for the questions already improves the convergence of the voting profile with respect to the presence of the winner in the top-10. This trend can be attributed to the fact that the original smartvote order organizes questions into groups based on themes (refer to Appendix A). Consequently, a random ordering introduces different aspects of the voters' political views earlier in the questionnaire, rather than sequentially addressing a single aspect at a time.

The HVQS algorithm with $n \in \{1, 2, 5, 10\}$ consistently outperforms both the original smartvote ordering and the random ordering in every canton, with peaks of double performance (double the percentage of voters with the winner in the top-10) for Canton Waadt after 20 questions. Moreover, the lower the number of initial random questions n, the better the performance: the best performance obtained by the algorithm is with only 1 initial random question, necessary for outlining a first profile of the user.

The algorithm seems to work better for larger cantons. Indeed, cantons with a larger amount of voters and candidates tend to gain more relative performance compared to smaller cantons (cfr. Canton Schaffhausen, with 1366 voters and 26 candidates, and Canton Waadt, with 14955 voters and 297 candidates, in Figure 10.3).

We can define a specific metric to quantify the effectiveness of the new algorithm, which measures the percentage of gained questions required to have the winner among the top-10 profiles of 90% of the voters. This measure provides valuable insight into the efficiency gained by the algorithm, with values ranging from 37% for Canton Thurgau to 61% for Canton Neuenburg.

In the case of Canton Thurgau, the original order takes 41 questions to reach the 90% threshold, while the improved HVQS version achieves the same level of consensus in just 26 questions.

On the other hand, Canton Neuenburg witnesses a more significant improvement. The original order requires 36 questions to reach the desired 90% threshold, while the optimized HVQS version achieves the same consensus with only 14 questions.

10.4.2 Final Top-5 Candidates in Top-10

We can measure the performance of the algorithm also considering the presence of the top-5 candidates in the top-10 profiles.

Figure 10.4 shows the relationship between the percentage of voters who have the final top-5 candidates included in their top-10 profile and the number of questions they have answered. For every canton, the first four rows of the respective heatmap report the HVQS performances with $n \in \{1, 2, 5, 10\}$ initial random question. The fifth row outlines the performance of a questionnaire where questions are ordered randomly and the last line reports the original smartvote ordering performance.

First of all, it is worth noting that this time, in most of the cantons, the randomly-ordered questionnaire does not outperform the original smartvote order. They appear to be equally reliable in terms of performance. In certain cases, the random ordering allows for better performance during the first half of the questionnaire, but it is eventually surpassed by the original ordering in the second half.

The HVQS results confirm the findings discussed in the previous subsection, where the version with only 1 initial random question demonstrates the most effective performance.

Quantifying the performance with the same measure we used in the previous section, we can conclude that Canton Jura gains a maximum improvement of 30% of fewer questions needed to get top-5 candidates in the top-10 profile of 90% of the voters. The original order requires 53 questions, while the optimized HVQS version achieves the same consensus with only 37 questions.

On the other hand, Canton Thurgau shows the minimum improvement, dropping from 60 questions with the original order to 52 with the personalized one.

10.5 Conclusions

The use of the Highest Variance Question Selection algorithm in smartvote significantly enhances the convergence of the final voting advice. This improvement is evident in the

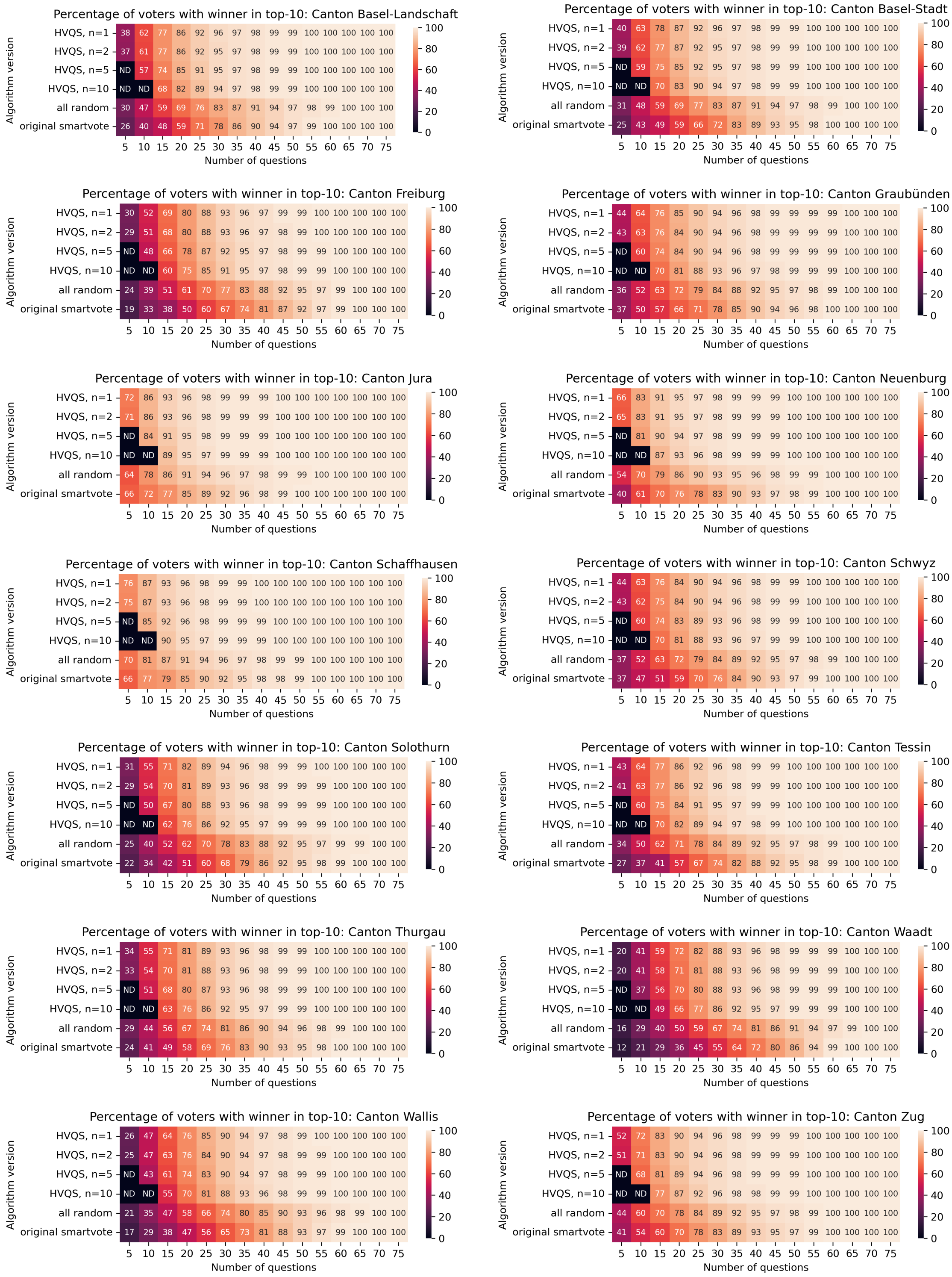


Figure 10.3: Percentage of voters with the final winner in the top-10 against the number of questions answered. Benchmarks of original smartvote order, random order, and HVQS algorithm with varying initial random questions (1, 2, 5, and 10).

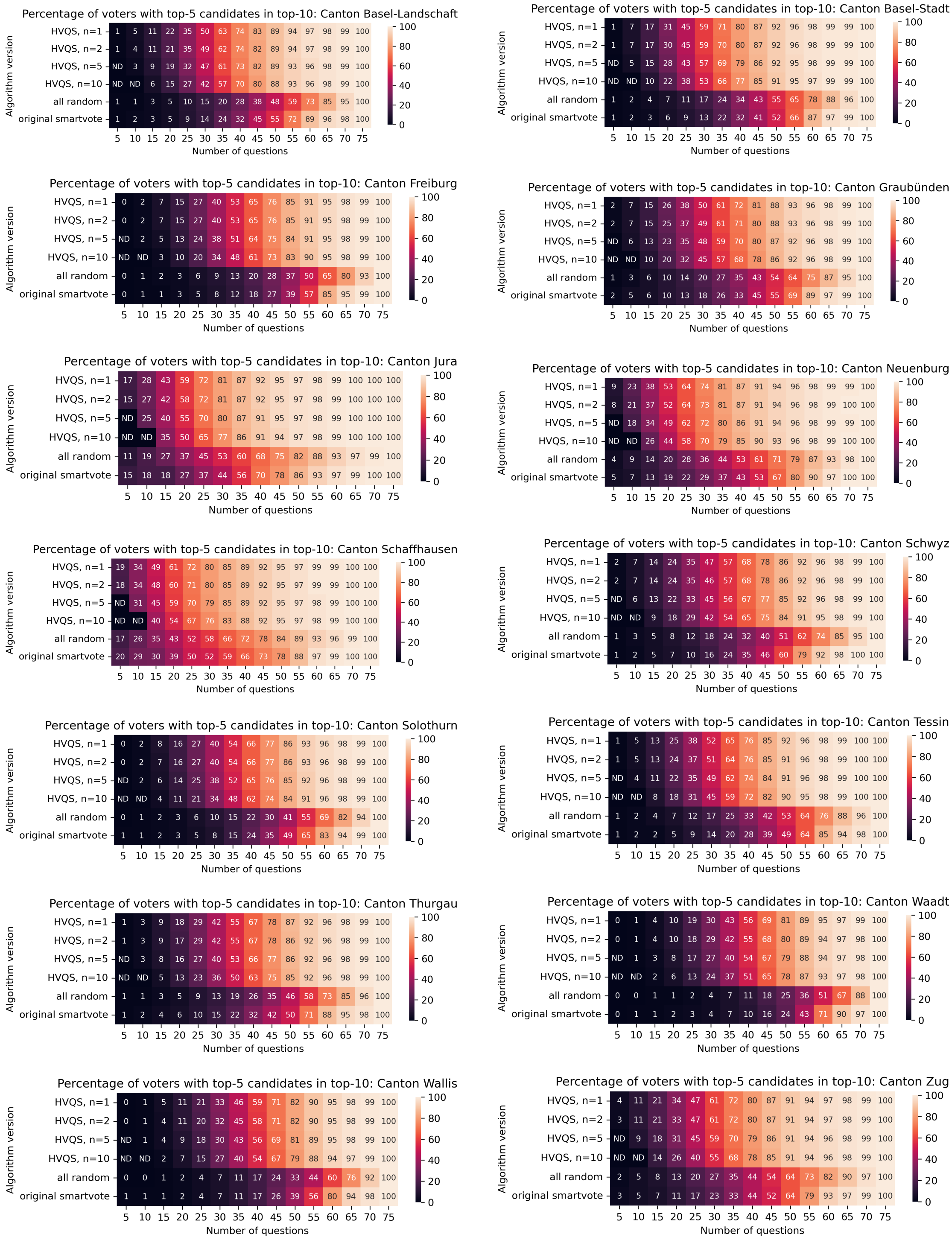


Figure 10.4: Percentage of voters with the final top-5 candidates in the top-10 against the number of questions answered. Benchmarks of original smartvote order, random order, and HVQS algorithm with varying initial random questions (1, 2, 5, and 10).

earlier appearance of both the winner and the top-5 candidates in users' profiles.

This outcome is quite promising as it suggests that employing a personalized question order can effectively mitigate voting fatigue and reduce the number of questions needed to get a reliable voting advice.

The benchmark results also indicate that there is no necessity to implement an initial round of n random questions, as using only $n = 1$ yields the best performance. Opting for a higher value of n would only offer advantages in terms of computational efficiency. Therefore, there exists a trade-off between performance and computational costs when considering the choice of n.

In summary, the implementation of the HVQS algorithm and personalized question ordering significantly improves the convergence of the smartvote voting advice. Additionally, using a single initial random question proves to be the most effective strategy, while higher values of n primarily impact computational efficiency rather than overall performance.

Bibliography

[1] D. Orsini-Rosenberg, "Machine learning vs. swiss politics," Jul. 2022.

[2] U. G. James Thurman, "Three case studies from switzerland: Smartvote," in *Berkman Center Research*, Mar. 2009.

[3] J.-F. Laslier, "Spatial approval voting," in *Cambridge University Press*, Jan. 2017.

[4] D. K. Fatih Erdem Kizilkaya, "Pluralityveto: A simple voting rule achieving optimal metric distortion," Jun. 2022.

[5] K. S. Sebastian Bensland and P. Ronchetti, "A machine learning analysis of the swiss political spectrum and candidate recommendation process," Jul. 2020.

Appendix A

Questionnaire

Page 1: Welfare state & Family		
Order	ID	Question
1	3412	Do you support an increase in the retirement age (e.g. to 67) ?
2	3413	Should the federal government provide more financial support for the creation of childcare facilities outside the family?
3	3414	An initiative calls for the introduction of paid paternity leave for four weeks. Do you support this proposal?
4	3415	Should the conversion rate of the occupational pension fund be reduced in order to adjust for increases in life expectancy?
5	3416	Do you support cantonal efforts to reduce social welfare benefits?
6	3417	Should the federal government provide more support for the construction of non-profit housing?

Page 2: Healthcare		
Order	ID	Question
7	3418	Should insured persons contribute more to healthcare costs (e.g. by increasing the minimal deductible)?
8	3419	Would you support the introduction of an opt-out solution of for organ donation?
9	3420	Should compulsory vaccination of children be introduced based on the Swiss vaccination plan?
10	3421	An initiative calls for health insurance subsidies to be designed so that no one needs to spend more than ten percent of their disposable income on health insurance premiums. Do you support this proposal?
11	3422	An initiative wants to give the federal government more powers to introduce measures to reduce healthcare costs (Introduction of a cost barrier). Do you support this proposal?

Page 3: Education		
Order	ID	Question
12	3423	Should the government increase its efforts to support equal education opportunities (e.g. through vouchers for private tutoring for students from low-income families)?
13	3424	Are you in favour of schools granting/allowing exemptions from individual subjects or events for religious reasons (e.g. PE/swimming, sex education, etc.)?
14	3425	Should the federal government expand its financial support for continued education and retraining?
15	3426	According to the Swiss integrated schooling concept, children with learning difficulties or disabilities should be taught in regular classes. Do you approve of this concept?

Page 4: Immigration & integration		
Order	ID	Question
16	3427	Should foreigners who have lived in Switzerland for at least ten years be given the right to vote and be elected at the municipal level?
17	3428	Should foreigners who have lived in Switzerland for at least ten years be given the right to vote and be elected at the municipal level?
18	3429	Should sans-papiers be able to obtain a regularized residence status more easily?
19	3430	Are you in favor of further tightening the asylum law?
20	3431	Should the requirements for naturalization be increased?
21	3491	Should the federal government provide more support for the integration of foreigners?

Page 5: Society & Ethics		
Order	ID	Question
22	3392	Should cannabis use be legalized?
23	3332	Should same-sex couples have the same rights as heterosexual couples in all areas?
24	3433	Should the rules for reproductive medicine be further relaxed?
25	3434	Are you in favour of stricter monitoring of pay equity for women and men?
26	3435	Would you be in favour of a doctor being allowed to administer direct active euthanasia in Switzerland?

Page 6: Finances & Taxes		
Order	ID	Question
27	3436	In your opinion, is lowering taxes at the federal level a priority for the next four years?
28	3437	Do you support a further reduction in contributions paid by financially strong cantons to financially weak cantons within the framework of financial equalisation (NFA)?
29	3438	Should married couples be taxed separately (individual taxation)?
30	3439	Are you in favour of restricting competition between the cantons with regard to corporate tax rates?

Page 7: Economy & Labour		
Order	ID	Question
31	3440	Should private households be free to choose their electricity supplier (complete liberalisation of the electricity market)?
32	3441	Are you in favour of introducing a general minimum wage of CHF 4'000 for all employees for full-time employment?
33	3442	Should investment controls be introduced in order to better protect Swiss companies from takeovers by foreign investors?
34	3443	Are you in favour of a complete liberalisation of business hours for shops?
35	3444	Should the protection against dismissal for older employees be extended?
36	3445	Should the federal government provide more support for public services (e.g. public transport, post offices) in rural regions?

Page 8: Digitisation		
Order	ID	Question
37	3446	Should the expansion of the mobile network according to the 5G standard continue?
38	3447	Should online brokerage services (e.g. "Airbnb" accommodations, "Uber" taxi services) be regulated more strongly?

Page 9: Energy & Transport		
Order	ID	Question
39	3448	An initiative calls for Switzerland to stop using fossil fuels by 2050. Do you support this proposal?
40	3449	Currently, a CO2 charge is levied on fossil combustibles (e.g. heating oil, natural gas). Should this charge be extended to motor fuels (e.g. petrol, diesel)?
41	3450	Should the federal government provide more support for renewable energies?
42	3451	Should high traffic motorways be expanded to six lanes?
43	3452	Are you in favour of introducing "Road Pricing" for motorised individual transport on busy roads?

Page 10: Nature Conservation		
Order	ID	Question
44	3453	Do you support the relaxation of the current measures to protect large predators (lynx, wolves, bears)?
45	3454	Should the current moratorium on genetically modified plants and animals in Swiss agriculture be extended beyond 2021?
46	3455	Should direct payments only be granted to farmers that provide an extended ecological performance record (e.g. no synthetic pesticides and limited use of antibiotics)?
47	3456	Are you in favour of extending landscape protection (e.g. stricter rules for building outside existing building zones)?
48	3457	Are you in favour of stricter animal welfare regulations for livestock (e.g. permanent access to outdoor areas)?

Page 11: Political System		
Order	ID	Question
49	3458	Should campaign finance for political parties and referendums be openly declared?
50	3459	Should the introduction of electronic voting in elections and referendums (e-voting) be further pursued?
51	3460	Are you in favour of lowering the voting age to 16?

Page 12: Security & Military		
Order	ID	Question
52	3398	Should Switzerland terminate the Schengen Agreement with the EU, in order to reintroduce more security checks directly on the border?
53	3461	Should the Federal Council's proposal to tighten the conditions for admission to the civil service be abandoned?
54	3462	Should the export of war materials from Switzerland be banned?
55	3463	Are you in favour of Switzerland acquiring new fighter jets for the armed forces?
56	3464	Do you support an expansion of the legal possibilities for using DNA analysis in investigations?

Page 13: Foreign Trade & Foreign Policy		
Order	ID	Question
57	3468	Should Switzerland start membership negotiations with the EU?
58	3469	Should Switzerland strive for a free trade agreement with the USA?
59	3470	An initiative calls for liability rules for Swiss companies with regard to compliance with human rights and environmental standards abroad to be tightened. Do you support this proposal?
60	3471	Are you in favour of Switzerland's candidacy for a seat on the UN Security Council?

Page 14: Values		
Order	ID	Question
61	3387	What is your position the following statement: "Someone who is not guilty, has nothing to fear from state security measures."
62	3465	What is your position the following statement: "In the long term, everyone benefits from a free market economy in the long term."
63	3399	What is your position the following statement: "Wealthy individuals should contribute more to the funding of the state."
64	3389	What is your position the following statement: "It is best for a child, when one parent stays home full-time for childcare."
65	3466	What is your position the following statement: "The ongoing digitalization offers significantly more opportunities than risks."
66	3388	What is your position the following statement: "Punishing criminals is more important than reintegrating them into society."
67	3467	What is your position the following statement: "Stronger environmental protection is necessary, even if its application limits economic growth."

Page 15: Federal Budget		
Order	ID	Question
68	3472	Should the federal government spend more or less in the area of “Development assistance”?
69	3473	Should the federal government spend more or less in the area of “National defence”?
70	3474	Should the federal government spend more or less in the area of “Public security”?
71	3475	Should the federal government spend more or less in the area of “Education and research”?
72	3476	Should the federal government spend more or less in the area of “Social services”?
73	3477	Should the federal government spend more or less in the area of “Road traffic (motorised individual transport)”?
74	3478	Should the federal government spend more or less in the area of “Public transport”?
75	3479	Should the federal government spend more or less in the area of “Agriculture”?

Appendix B

Utilitarian Rule

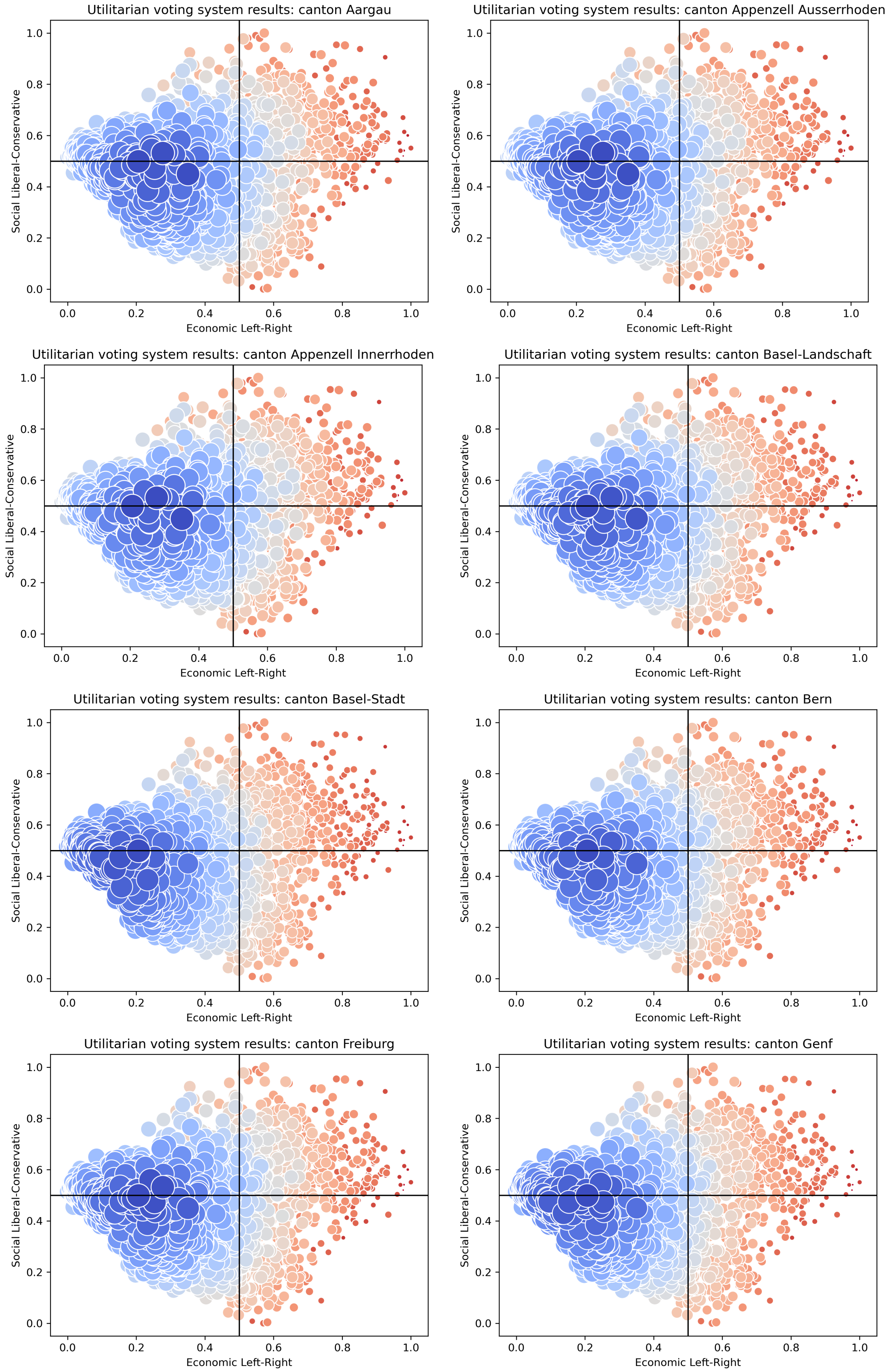


Figure B.1: Utilitarian rule results for each canton: Cantons A to Ge.

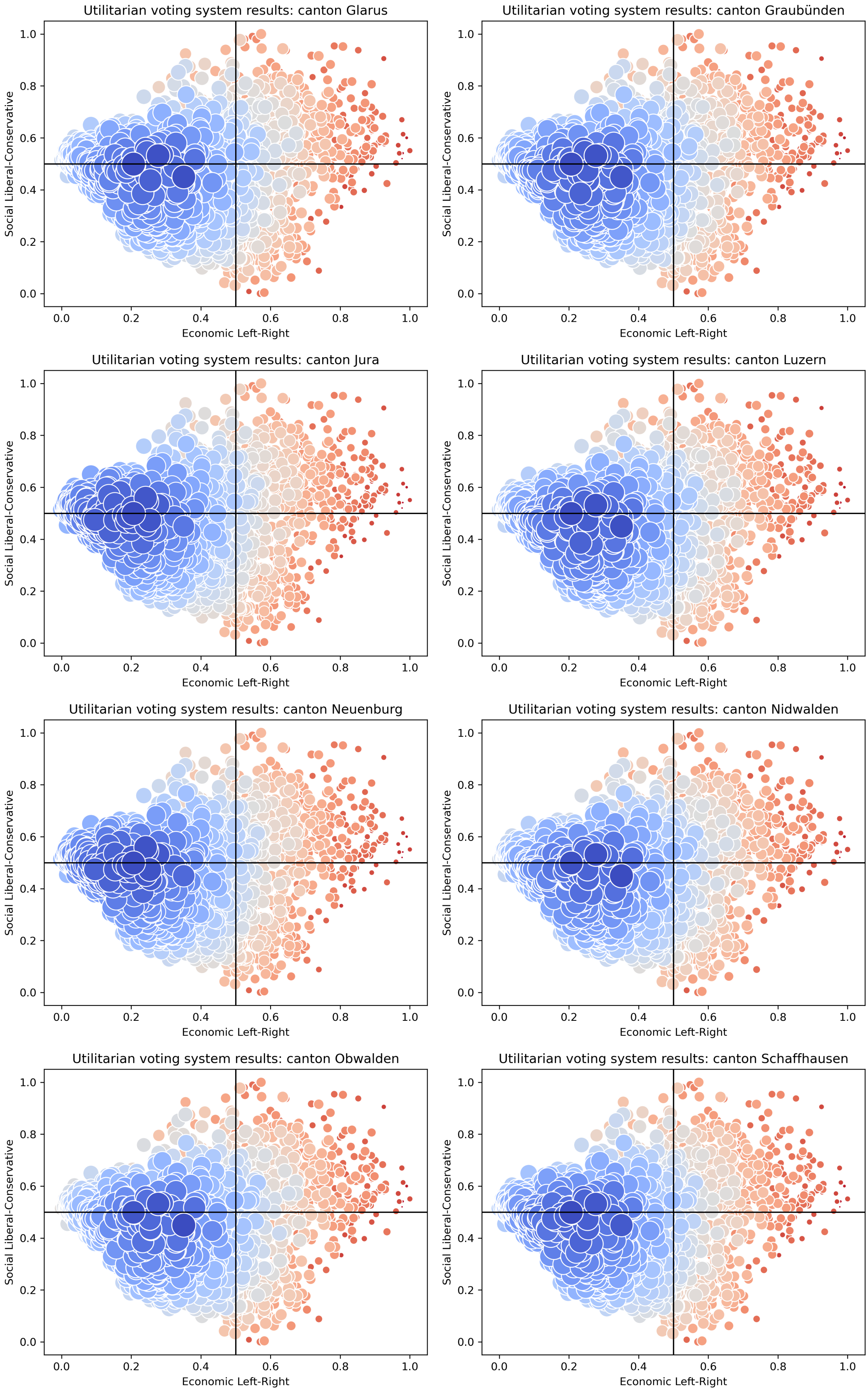


Figure B.2: Utilitarian rule results for each canton: Cantons Gl to Scha.

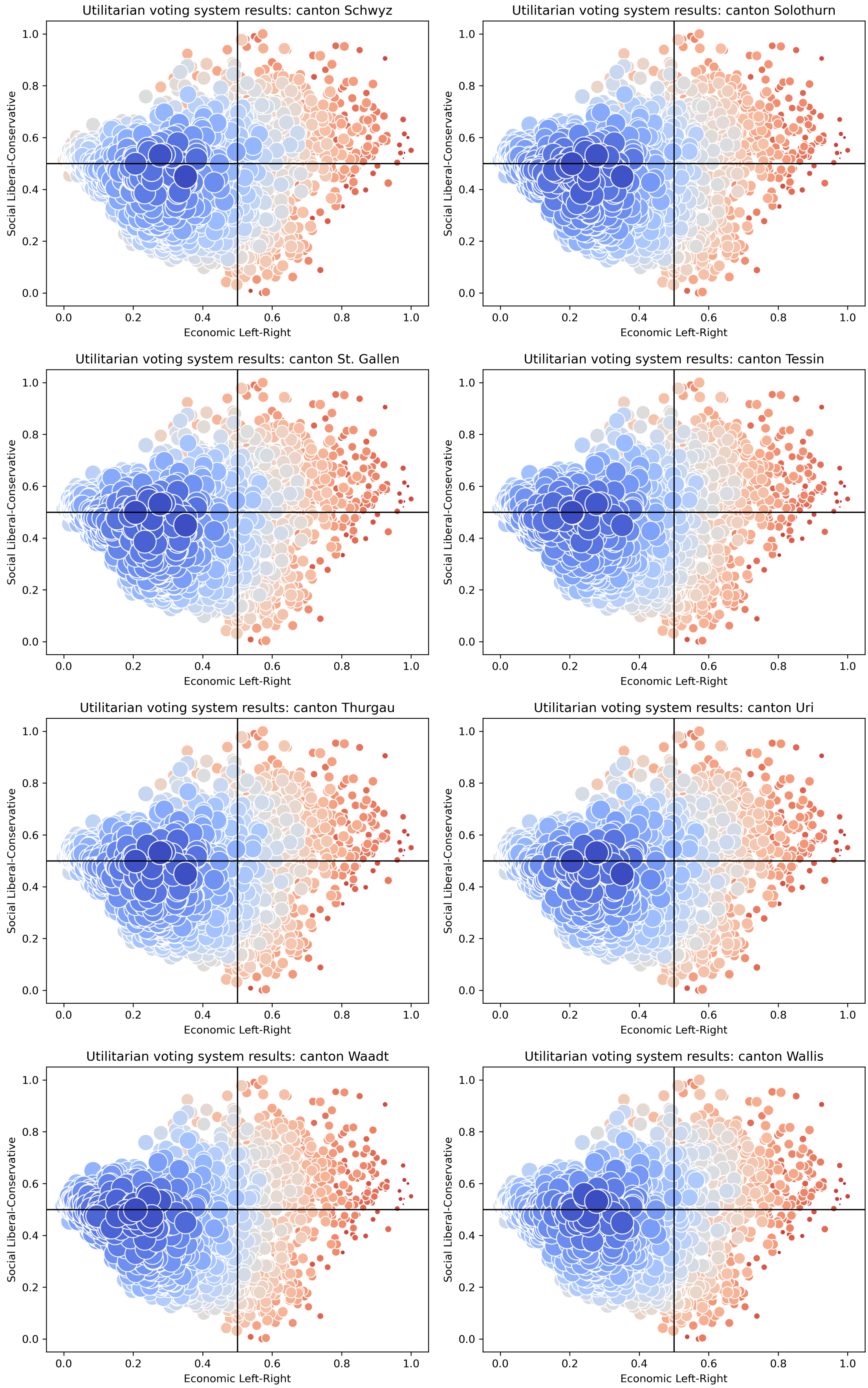


Figure B.3: Utilitarian rule results for each canton: Cantons Schw to W.

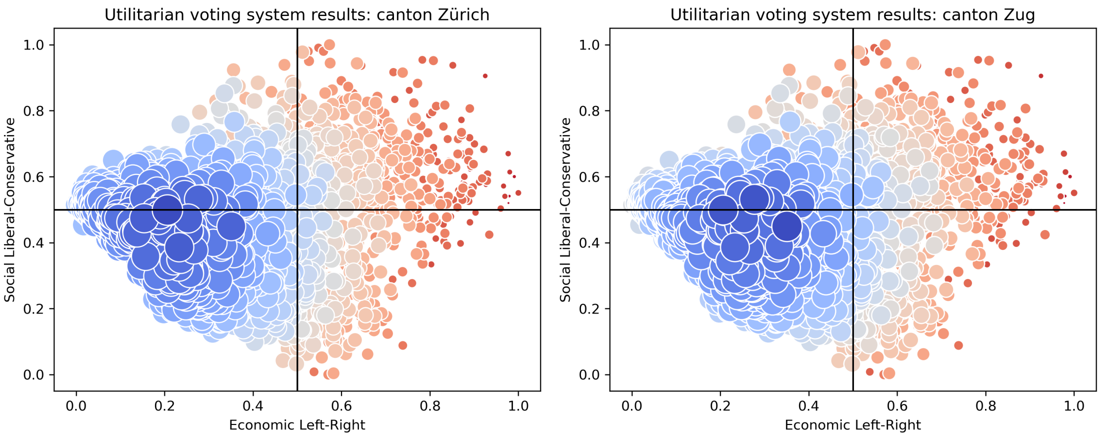


Figure B.4: Utilitarian rule results for each canton: Cantons Zue to Z.

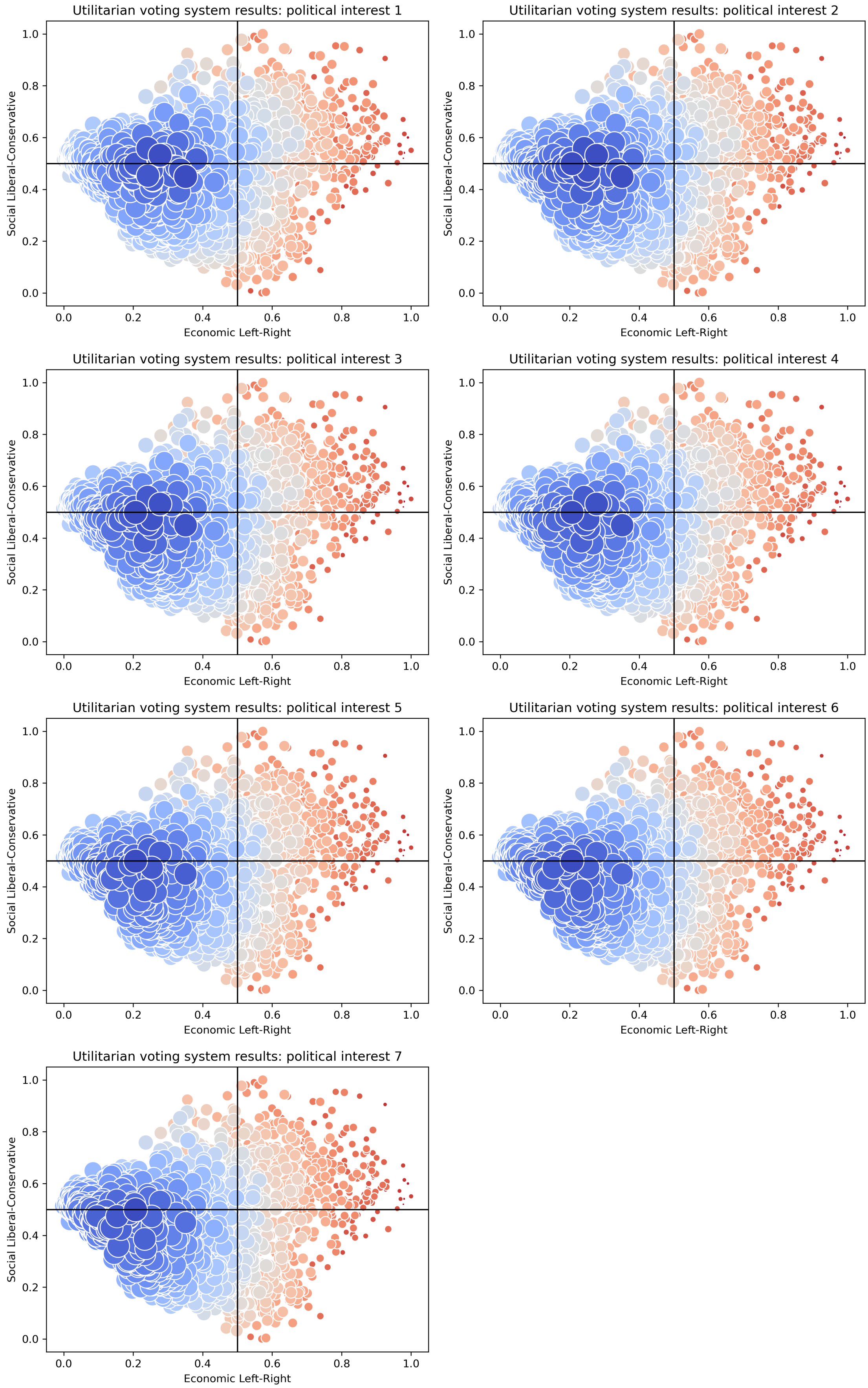


Figure B.5: Outcomes of utilitarian rule of different political interest levels. Users can select a political interest level between 1 and 7.

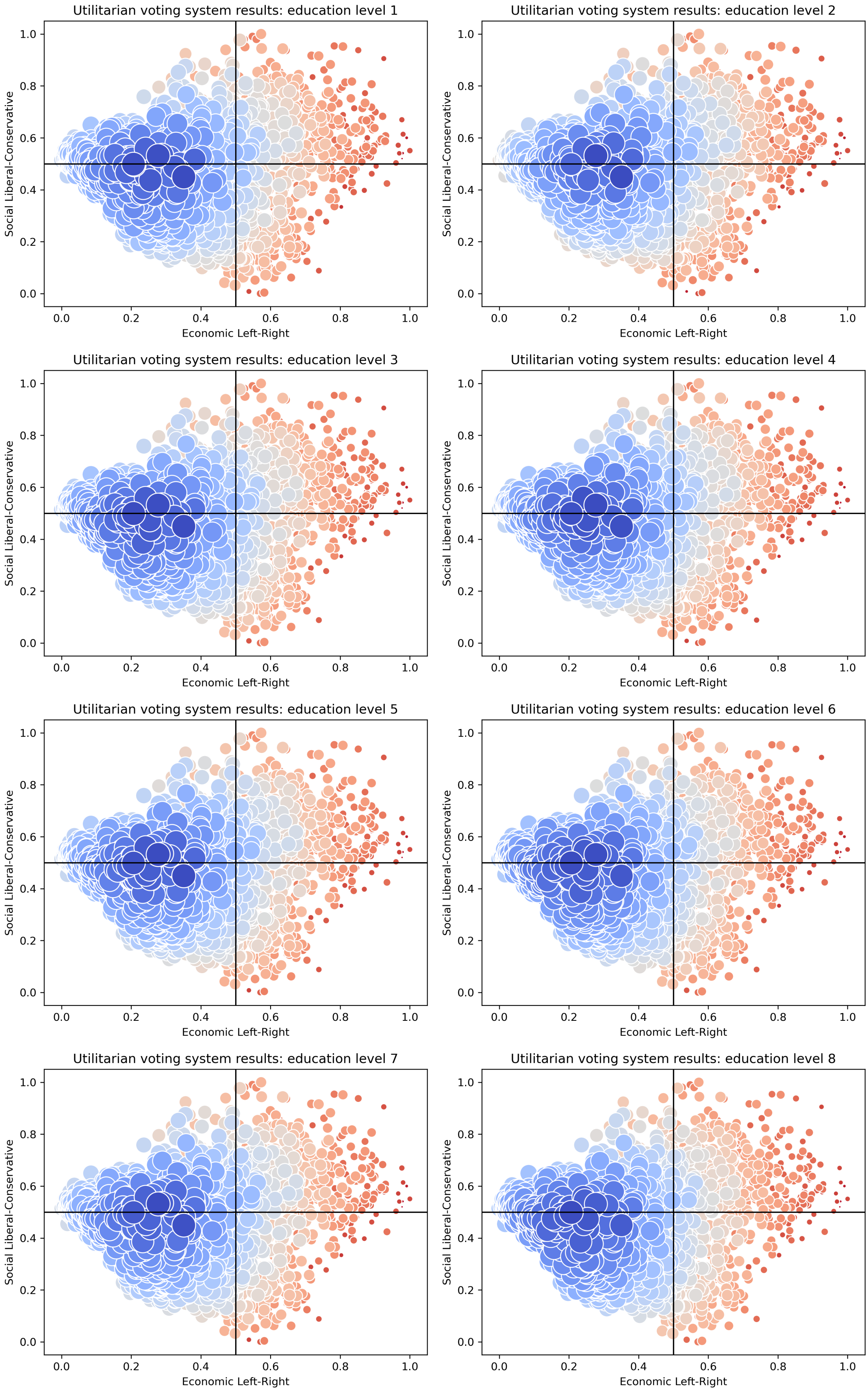


Figure B.6: Outcomes of utilitarian rule of different education levels. Users can select an education level between 1 (low education) and 14 (high education). Levels 1 to 8.

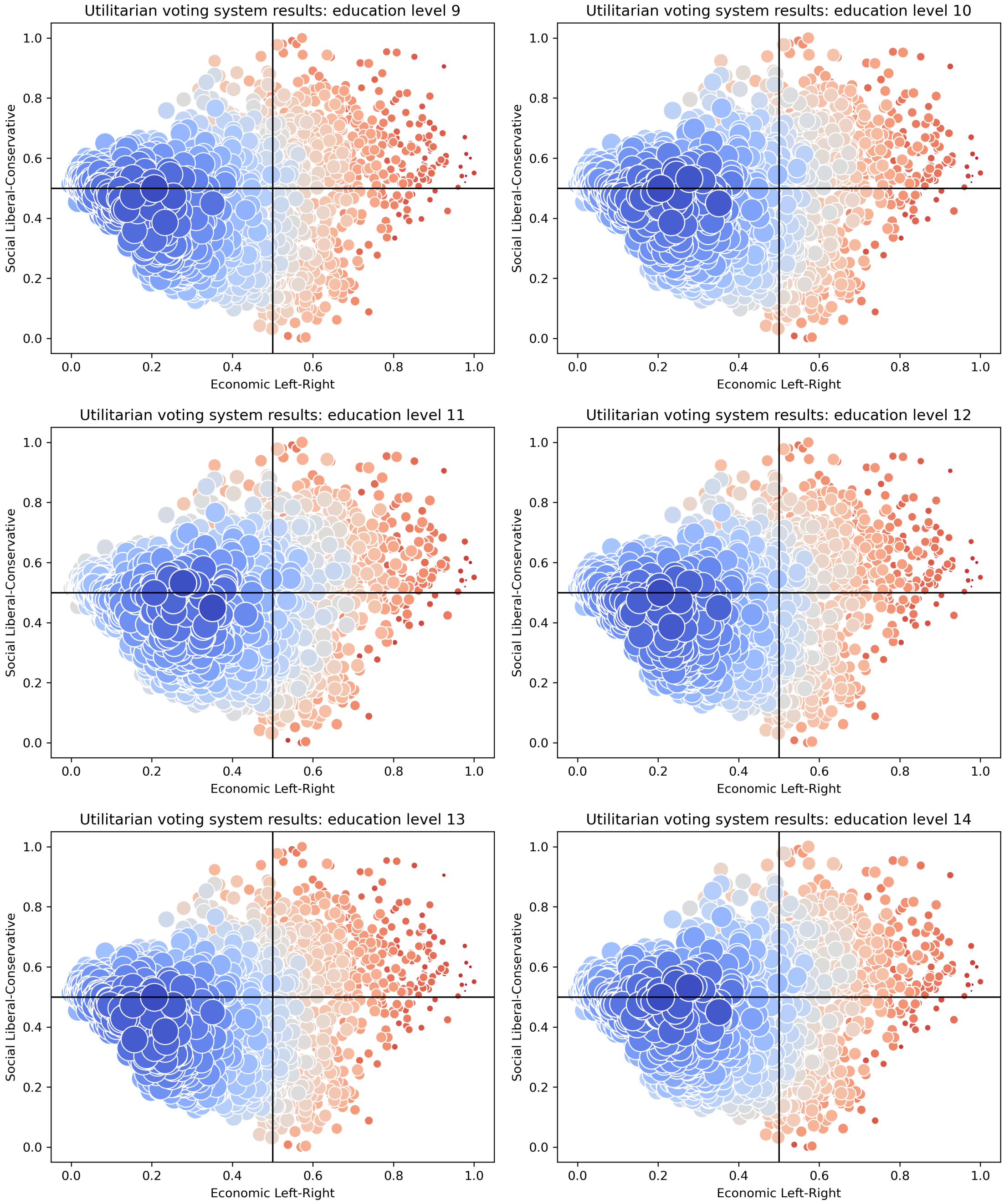


Figure B.7: Outcomes of utilitarian rule of different education levels. Users can select an education level between 1 (low education) and 14 (high education). Levels 9 to 14.

Appendix C

Plurality Voting System

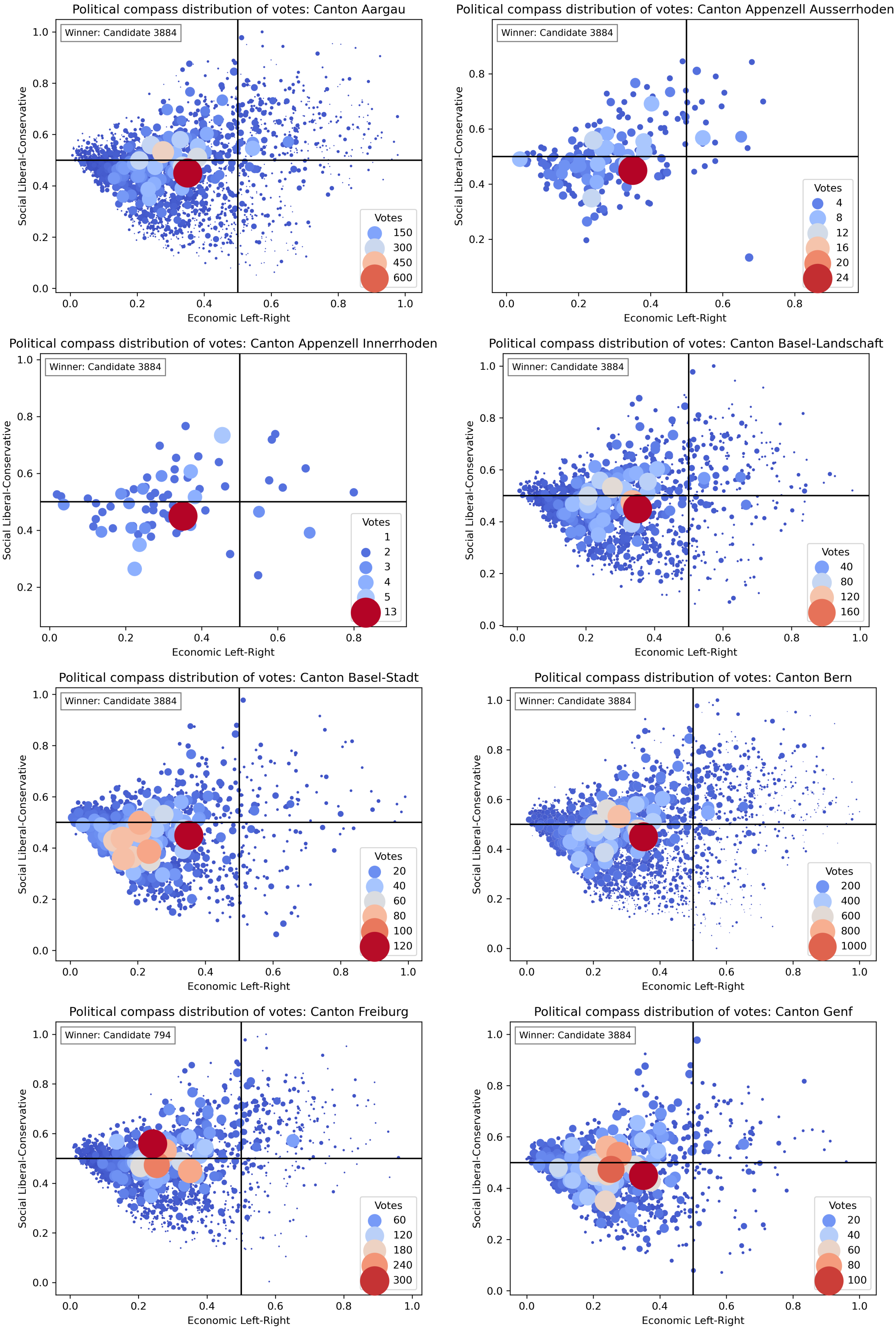


Figure C.1: Plurality voting system results for each canton: Cantons A to Ge.

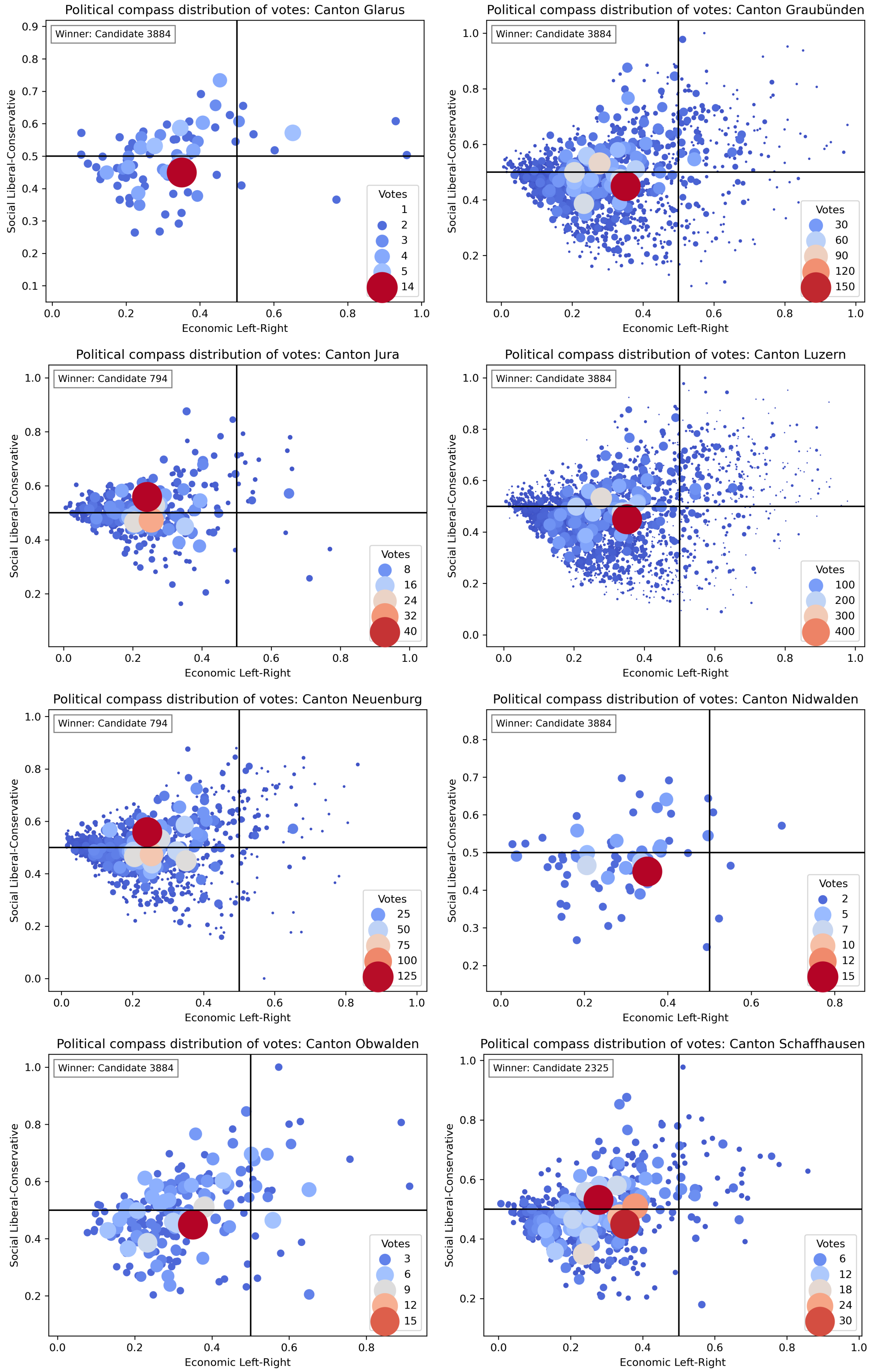


Figure C.2: Plurality voting system results for each canton: Cantons Gl to Scha.

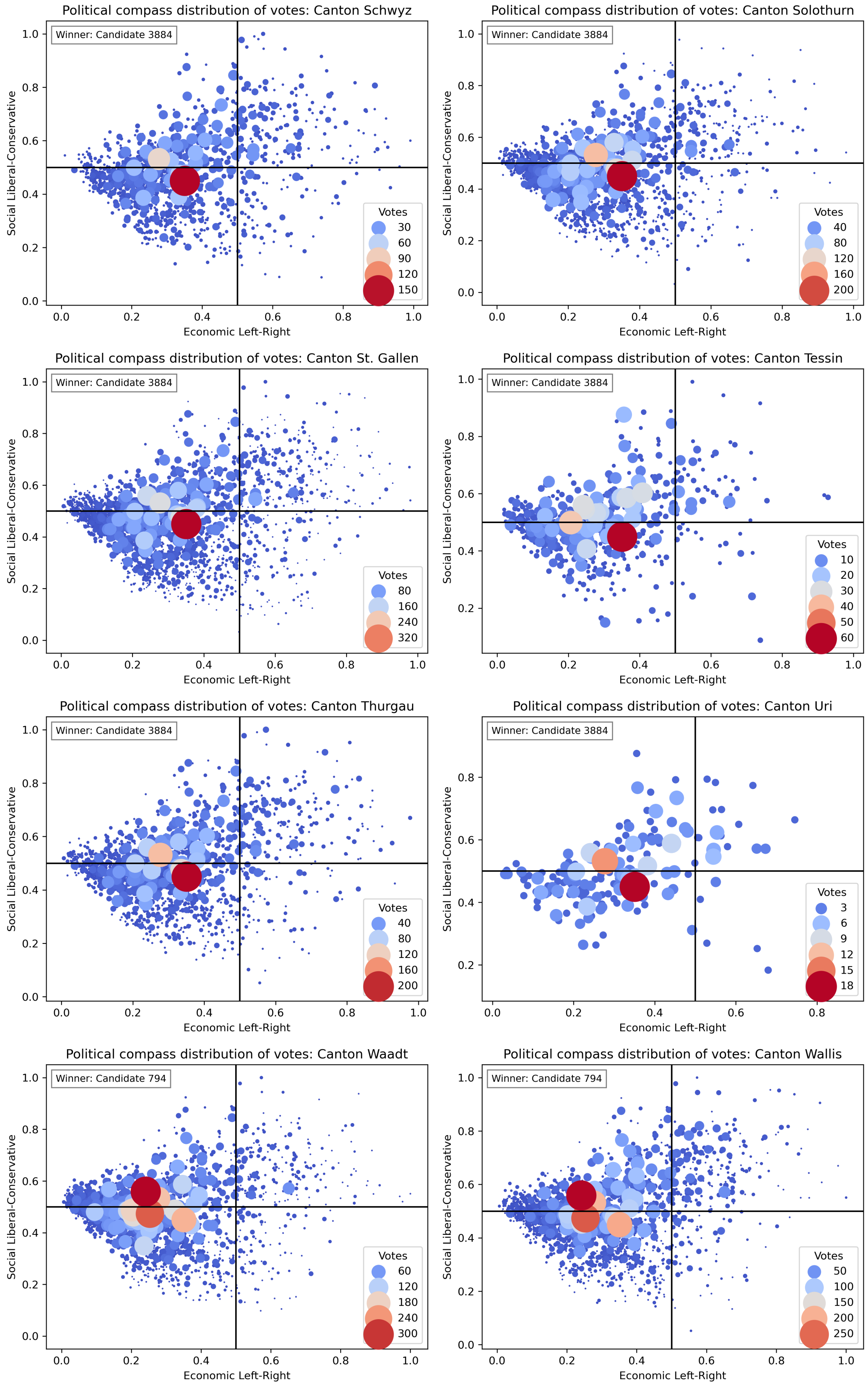


Figure C.3: Plurality voting system results for each canton: Cantons Schw to W.

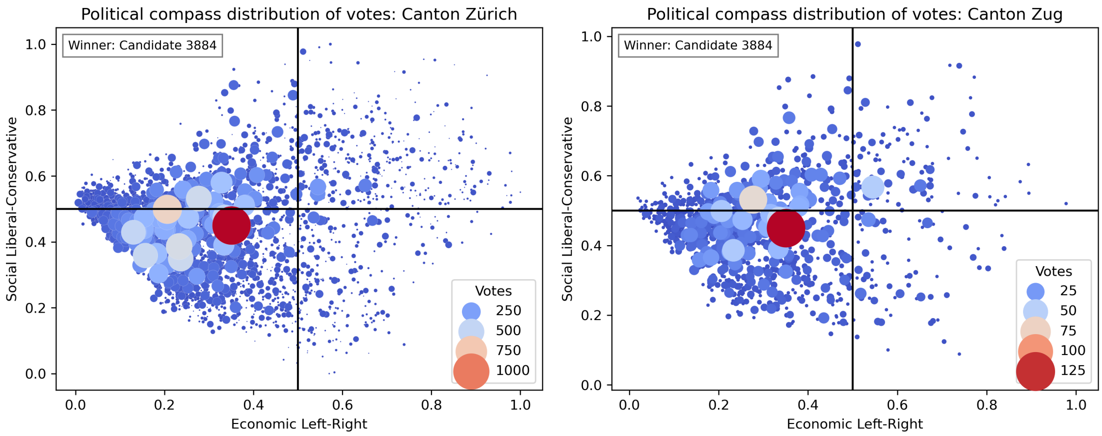


Figure C.4: Plurality voting system results for each canton: Cantons Zue to Z.

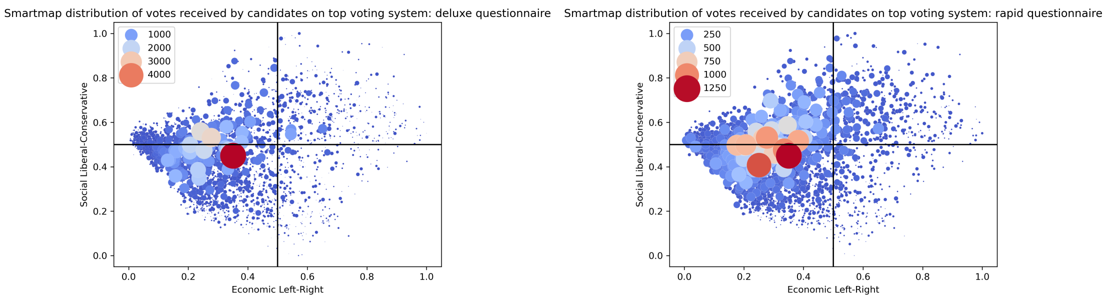


Figure C.5: Outcomes of plurality voting system of different smartvote questionnaire types. Deluxe is composed of 75 questions. Rapid is composed of 31 questions.

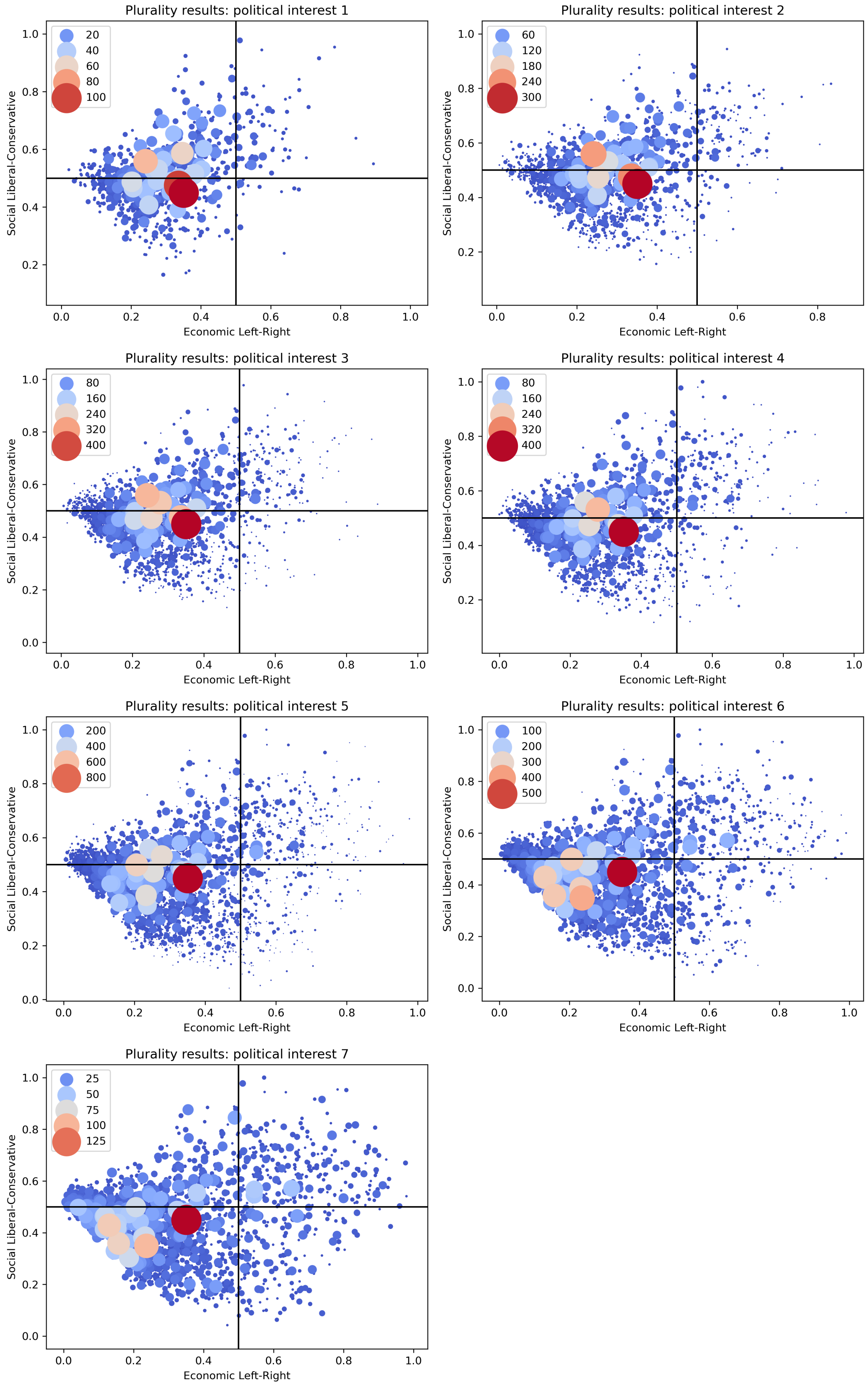


Figure C.6: Outcomes of plurality voting system of different political interest levels. Users can select a political interest level between 1 and 7.

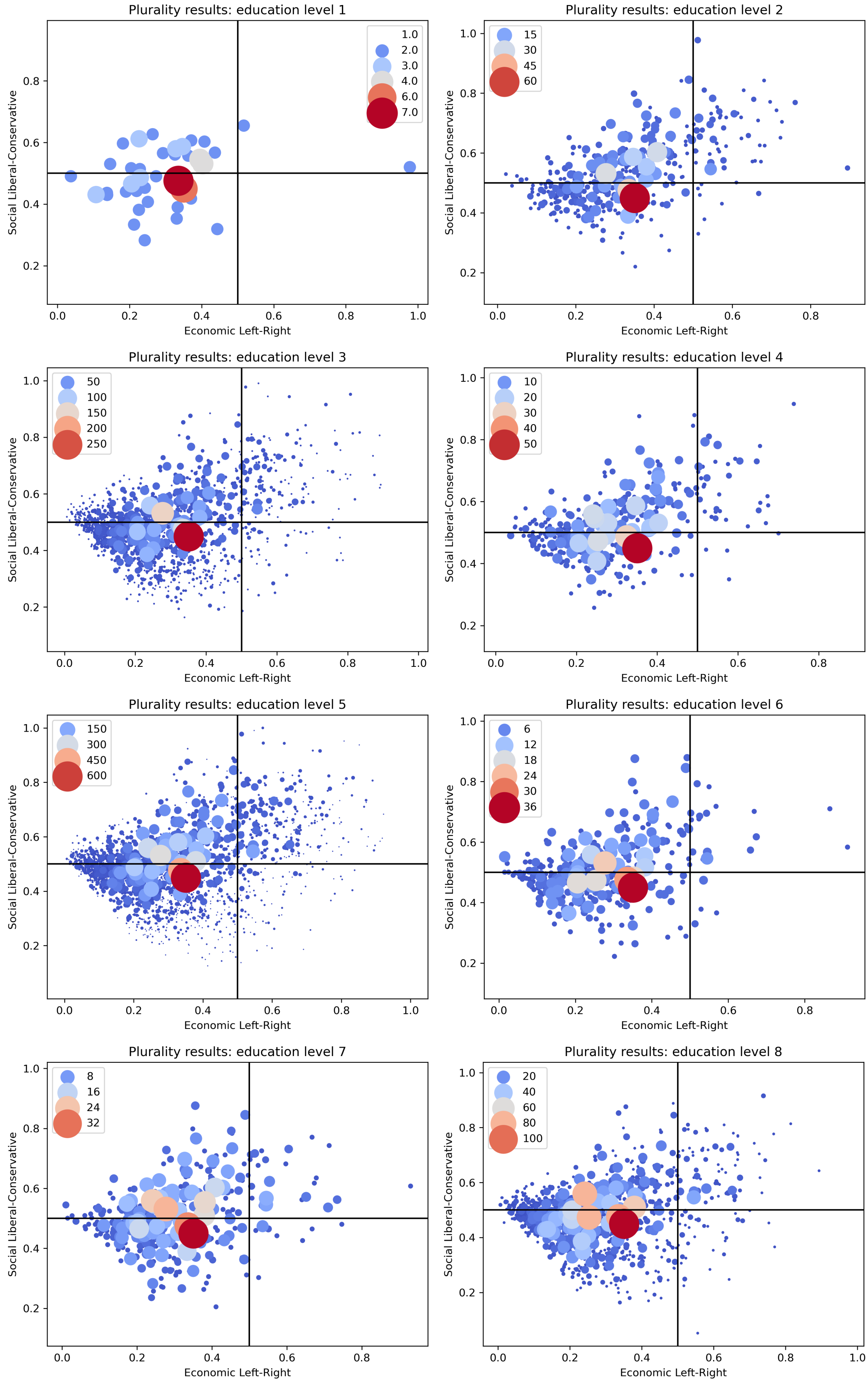


Figure C.7: Outcomes of plurality voting system of different education levels. Users can select an education level between 1 (low education) and 14 (high education). Levels 1 to 8.

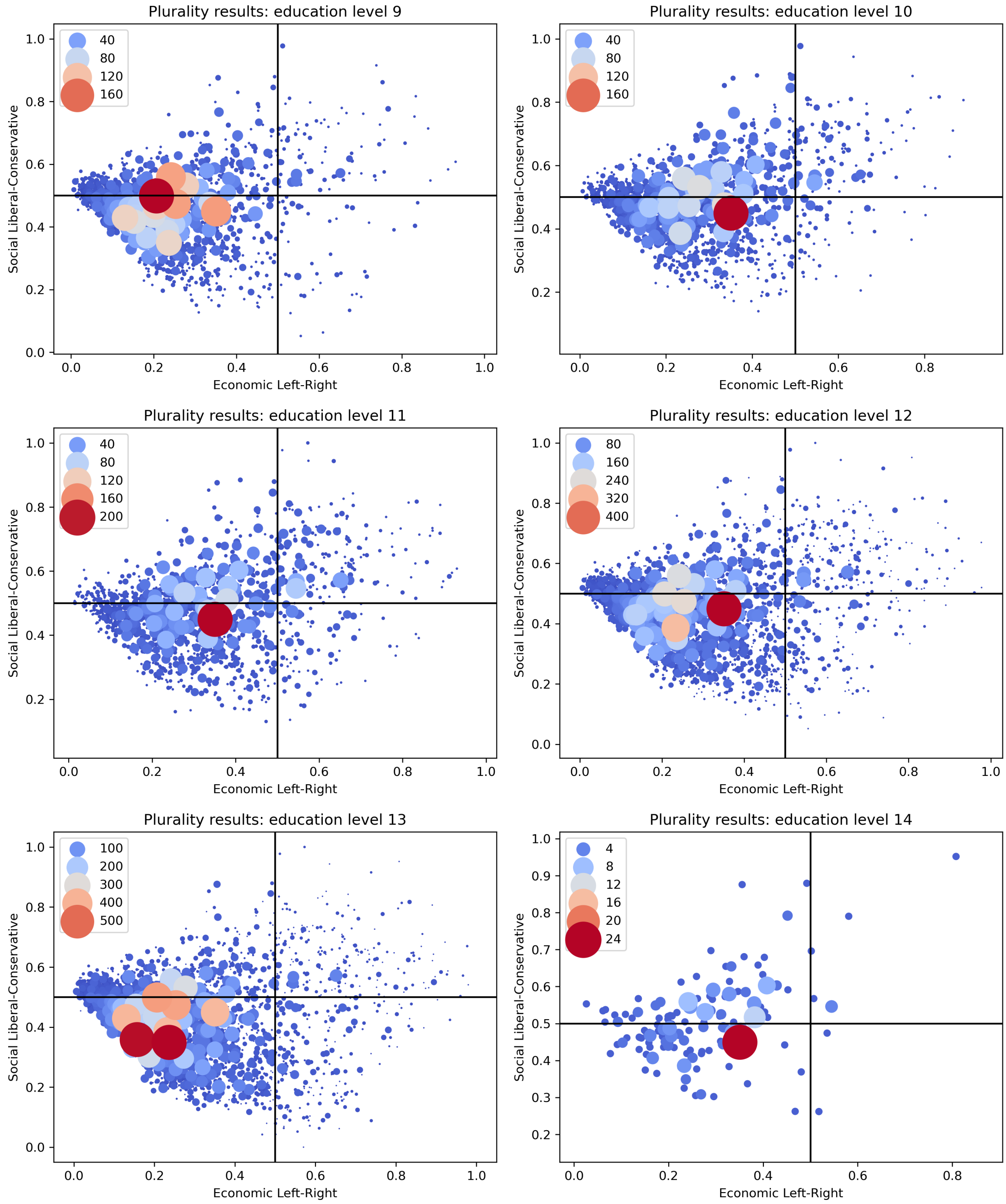


Figure C.8: Outcomes of plurality voting system of different education levels. Users can select an education level between 1 (low education) and 14 (high education). Levels 9 to 14.

Appendix D

Two-Round Voting Systems

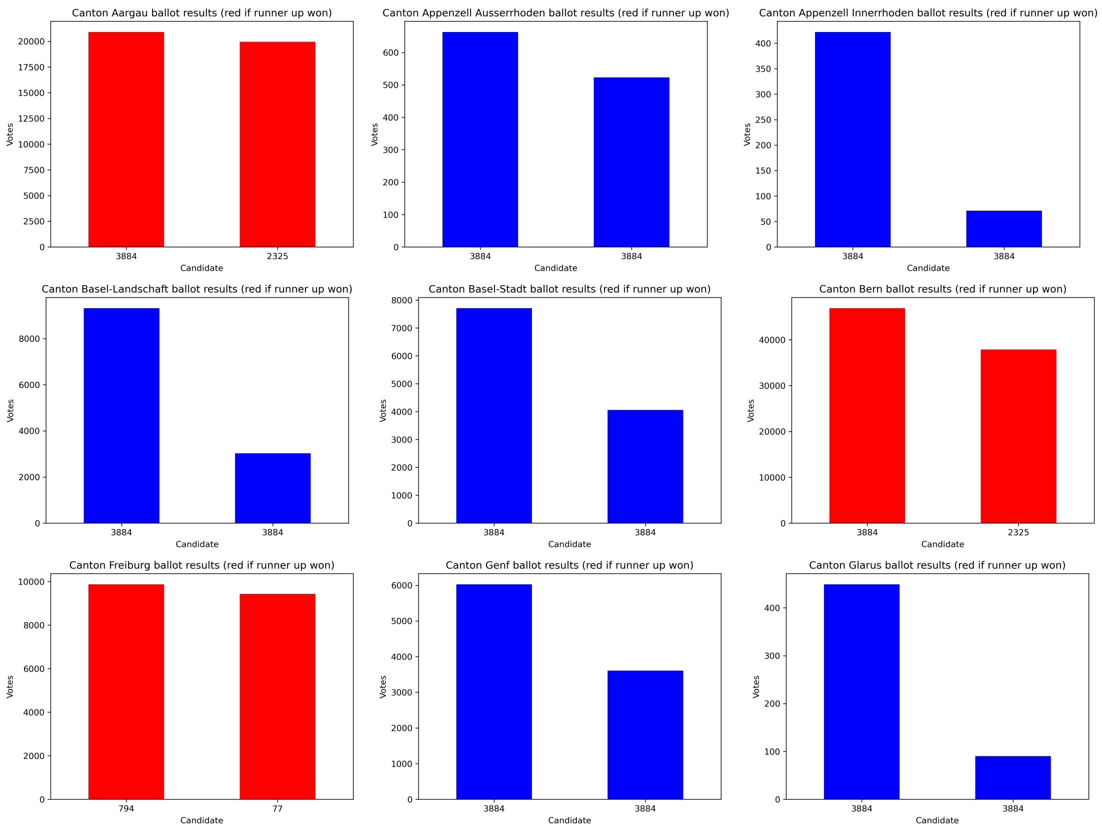


Figure D.1: Top-2 candidates ballot result in the second round of the election. The plot is red if the runner-up won. Blue otherwise. Cantons A to Gl.

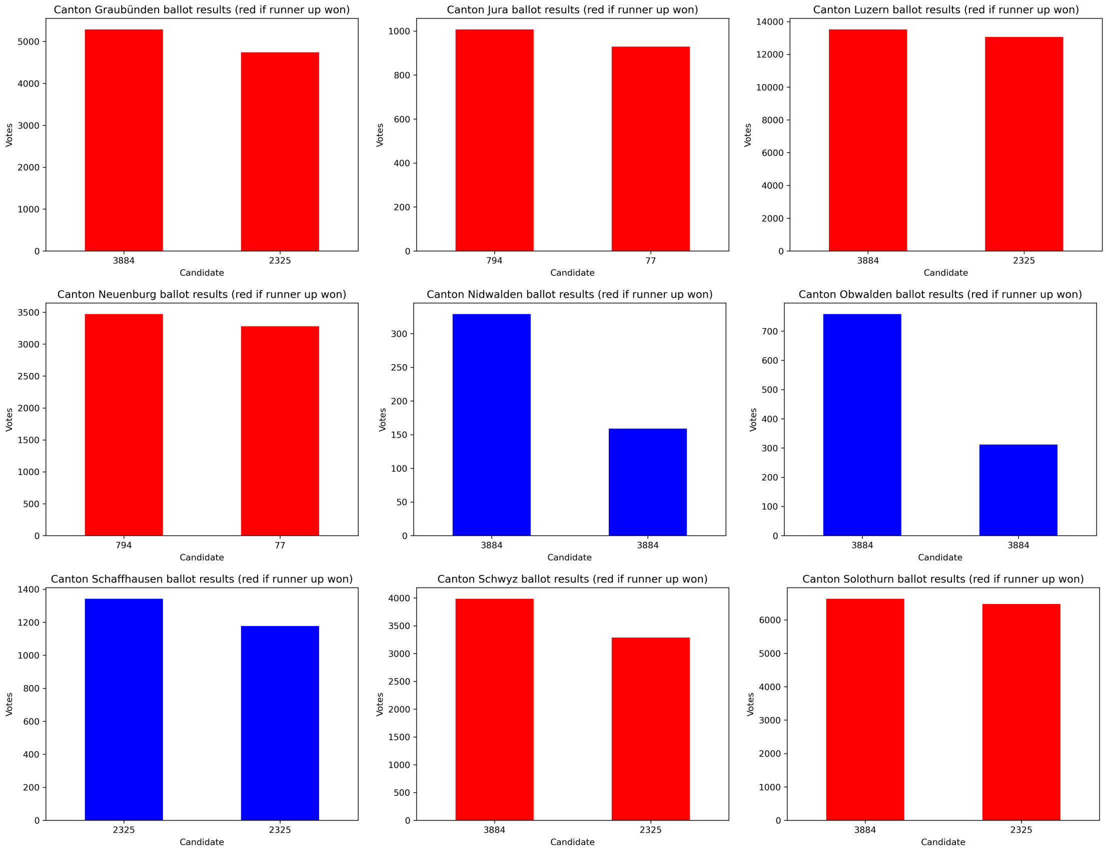


Figure D.2: Top-2 candidates ballot result in the second round of the election. The plot is red if the runner-up won. Blue otherwise. Cantons Gr to So.

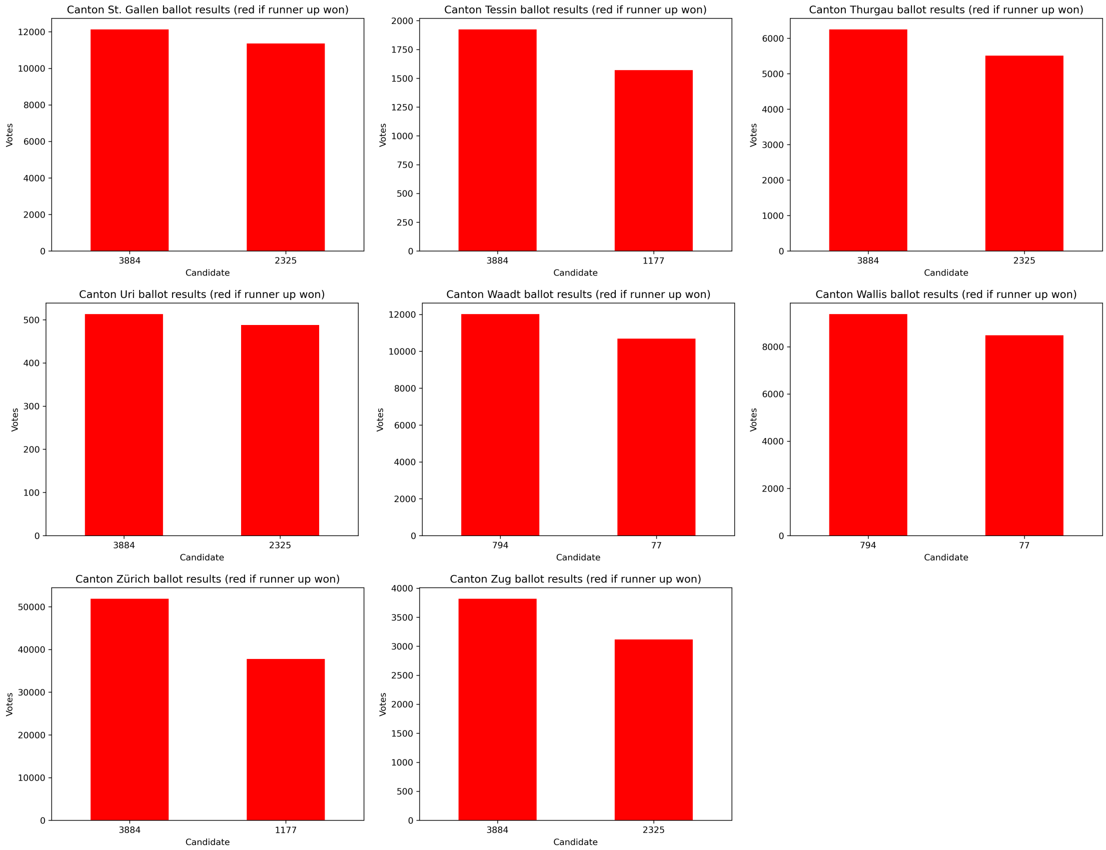


Figure D.3: Top-2 candidates ballot result in the second round of the election. The plot is red if the runner-up won. Blue otherwise. Cantons St to Z.

Appendix E

Instant Runoff Voting System

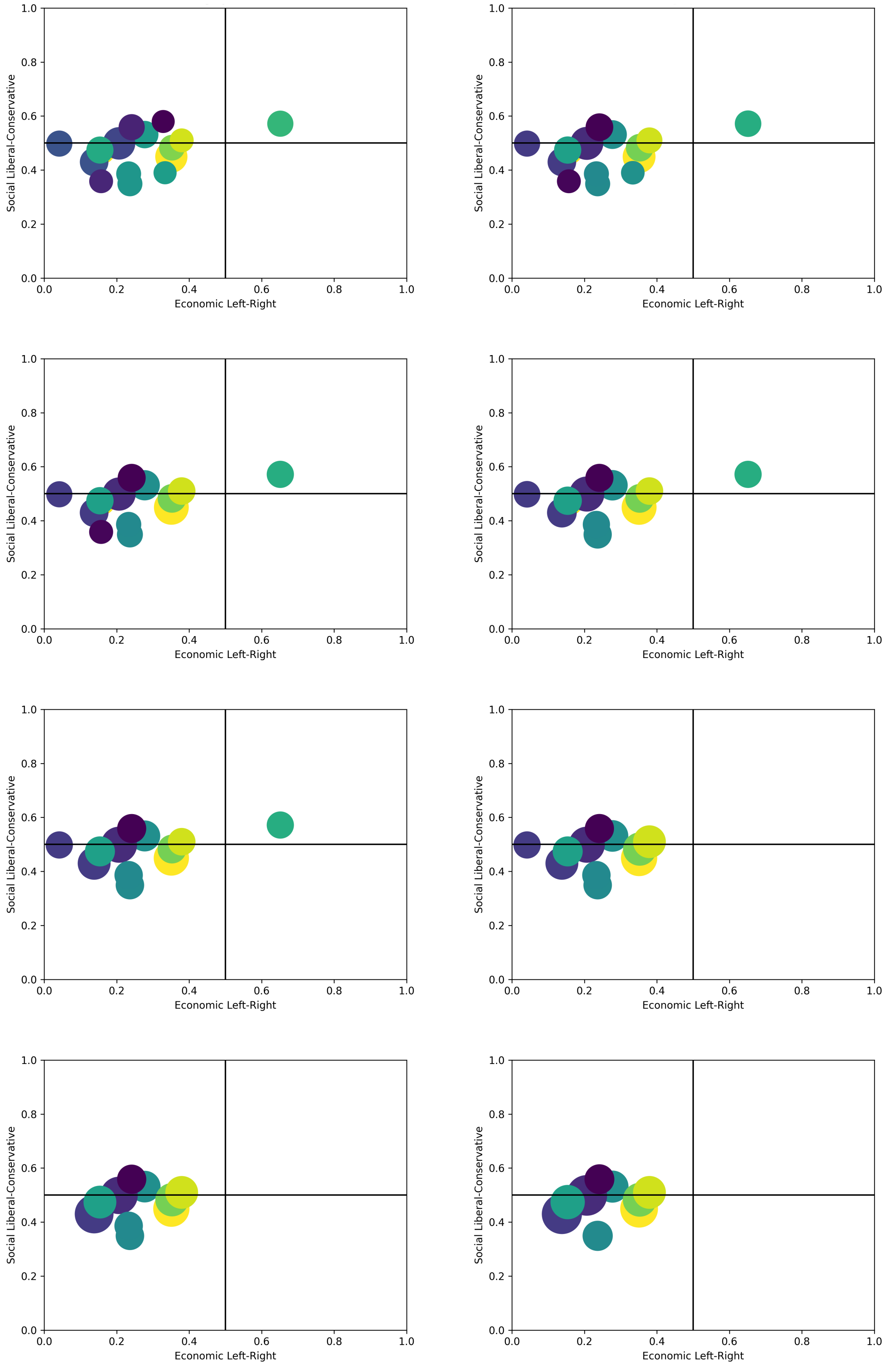


Figure E.1: Part 1 of last 16 snapshots of instant runoff voting system evolution. Last 16 candidates to last 9 candidates. Snapshots 16 to 9.

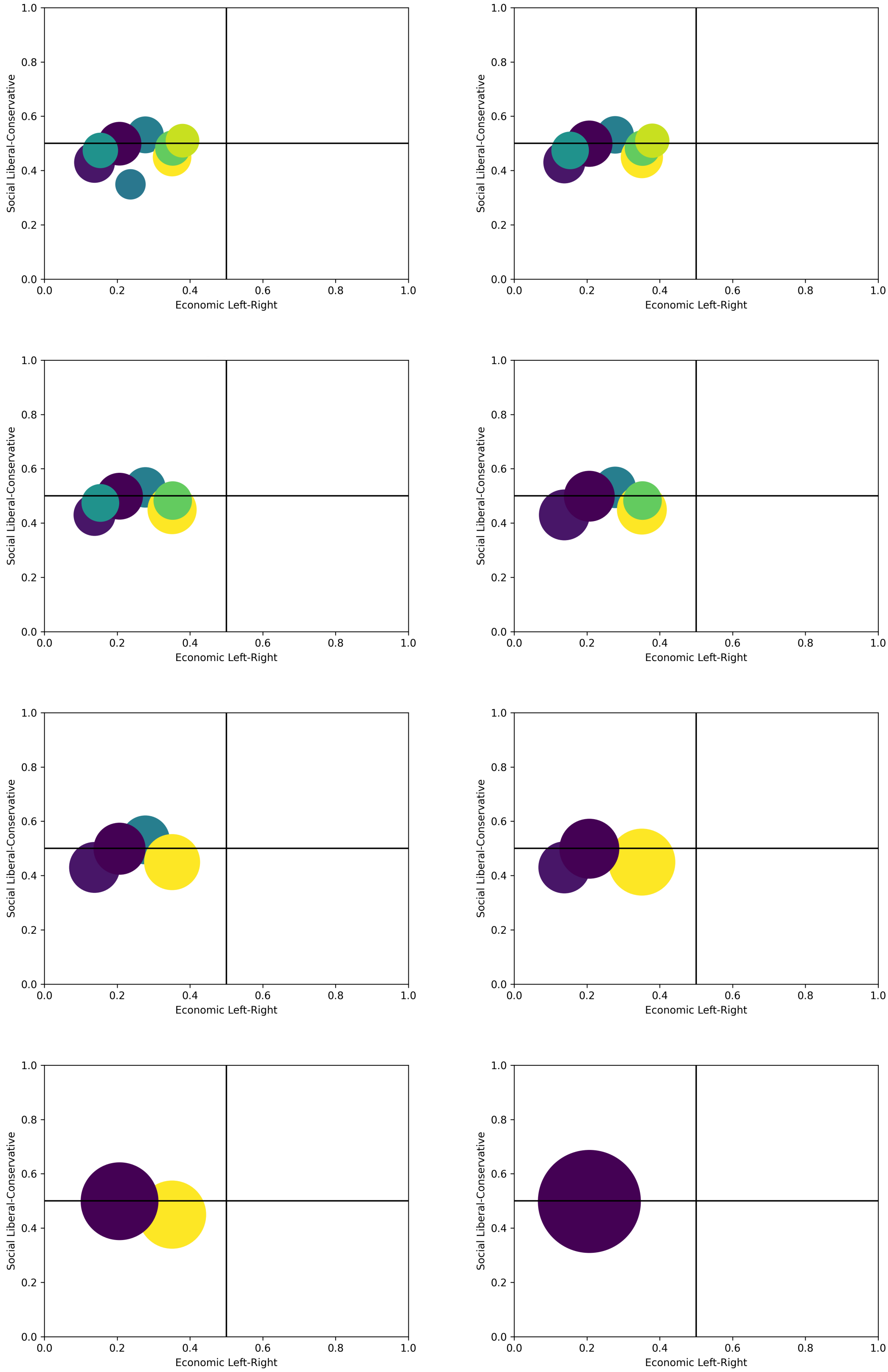


Figure E.2: Part 2 of last 16 snapshots of instant runoff voting system evolution. Last 8 candidates to last 1 candidate. Snapshots 8 to 1.

Appendix F

Sortition Voting System

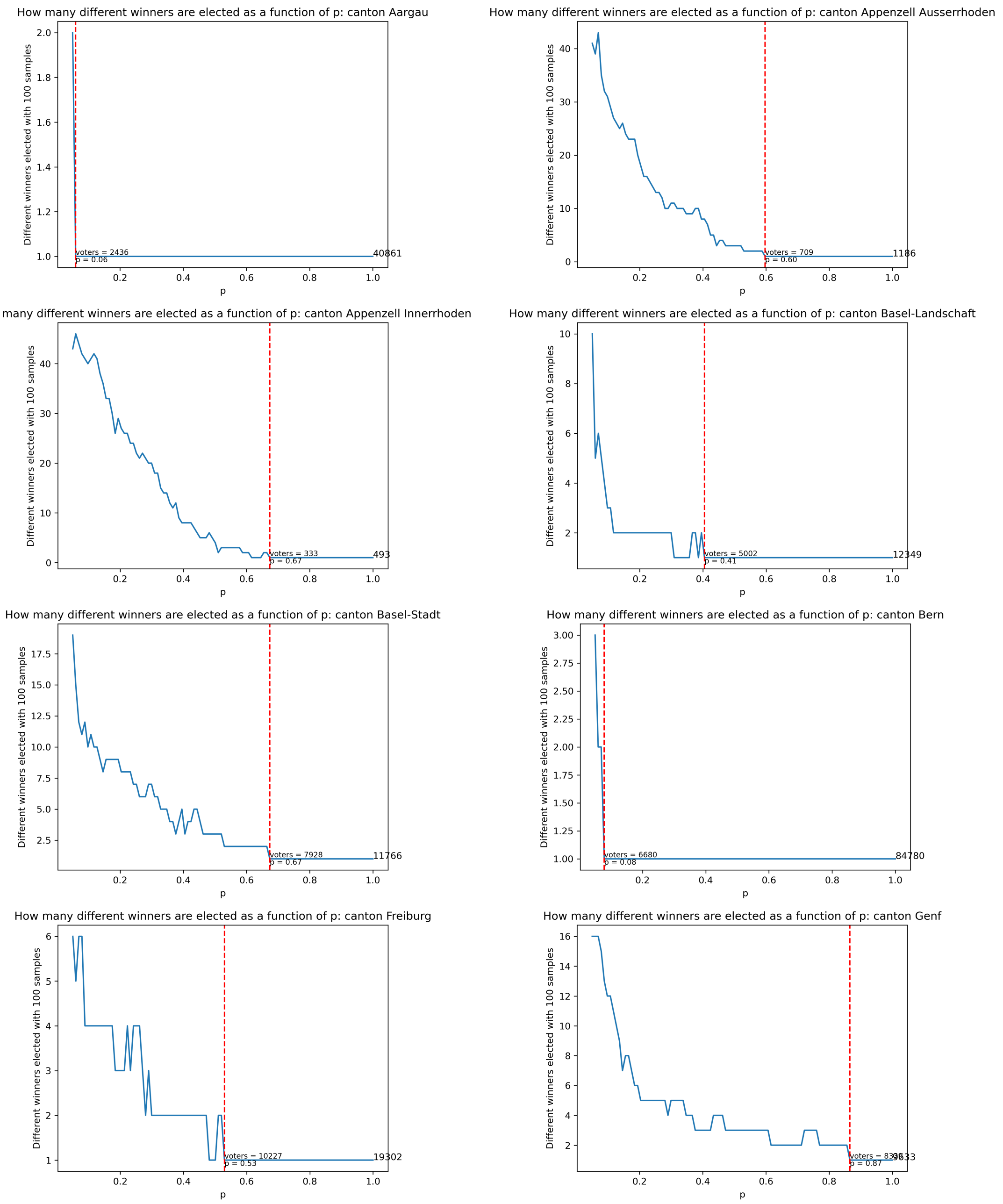


Figure F.1: Amount of unique winners sampling 100 times p percent of the cantonal population. Cantons A to Ge.

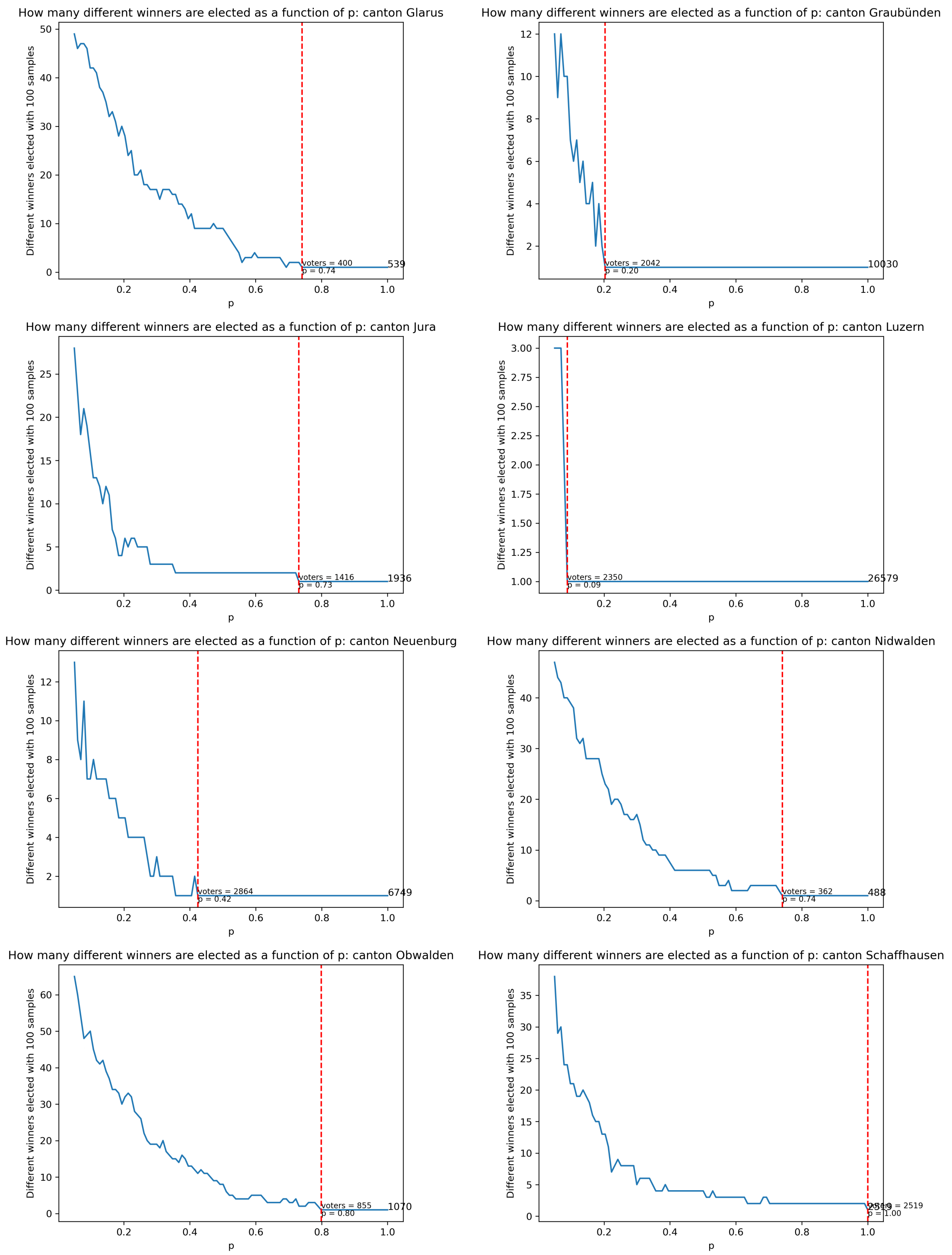


Figure F.2: Amount of unique winners sampling 100 times p percent of the cantonal population. Cantons Gl to Scha.

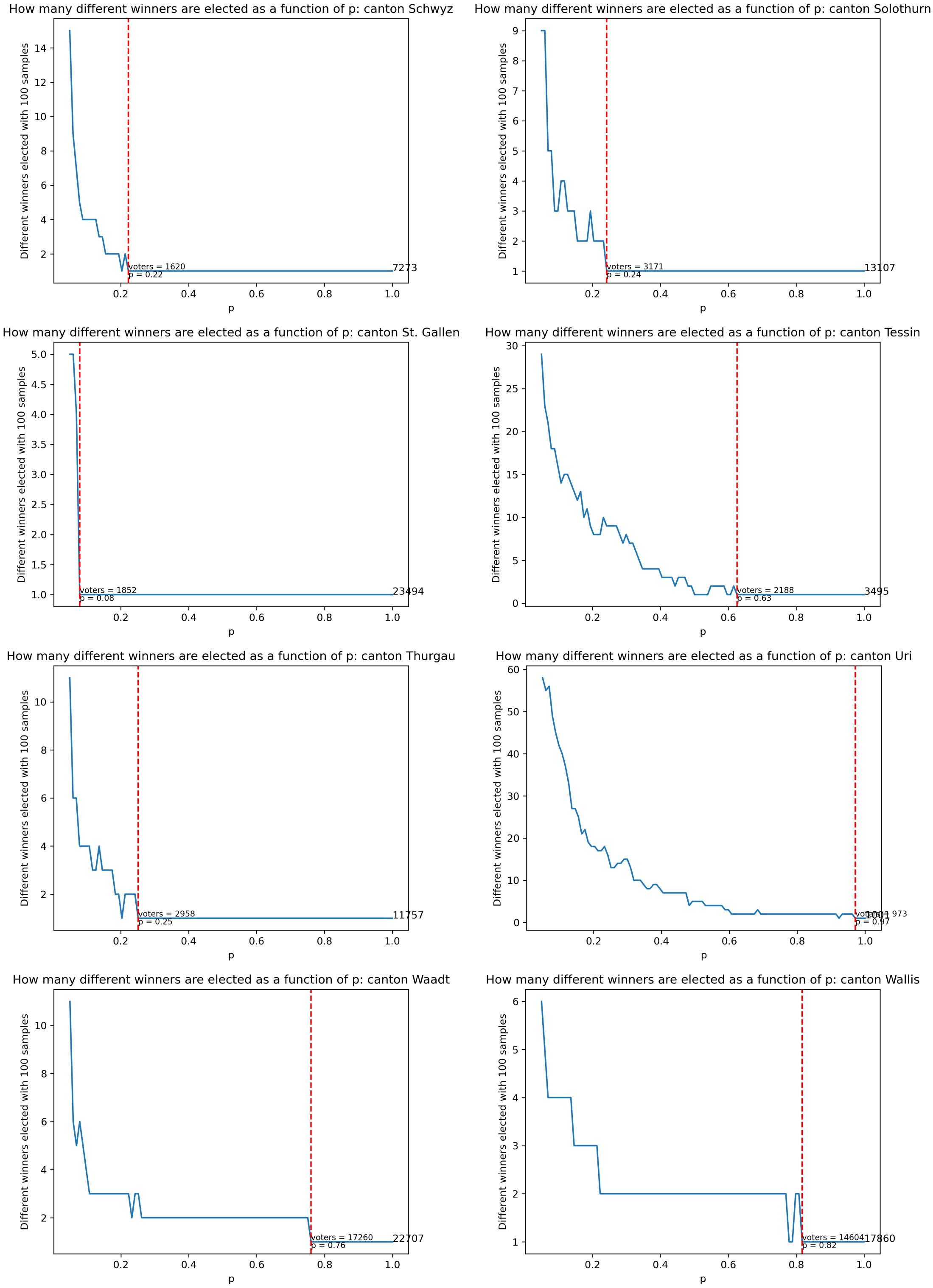
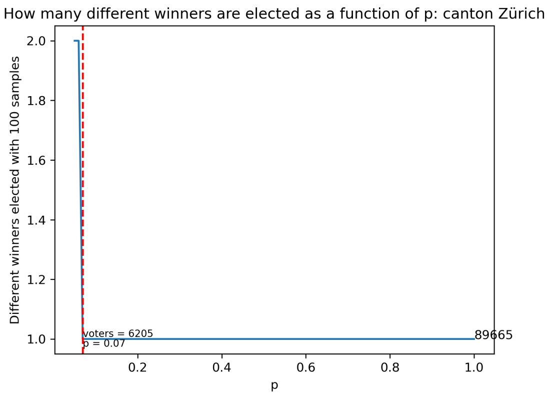


Figure F.3: Amount of unique winners sampling 100 times p percent of the cantonal population. Cantons Schw to W.



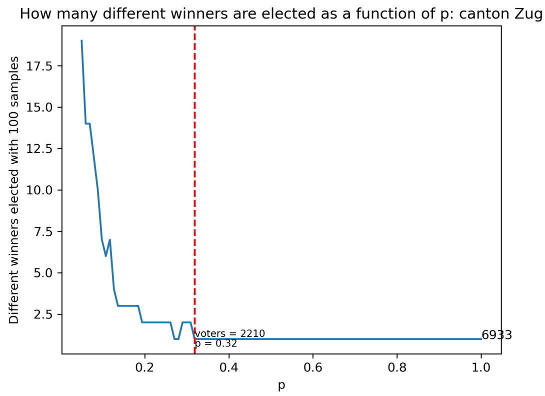


Figure F.4: Amount of unique winners sampling 100 times p percent of the cantonal population. Cantons Zue to Z.

Appendix G

Modified Questionnaires

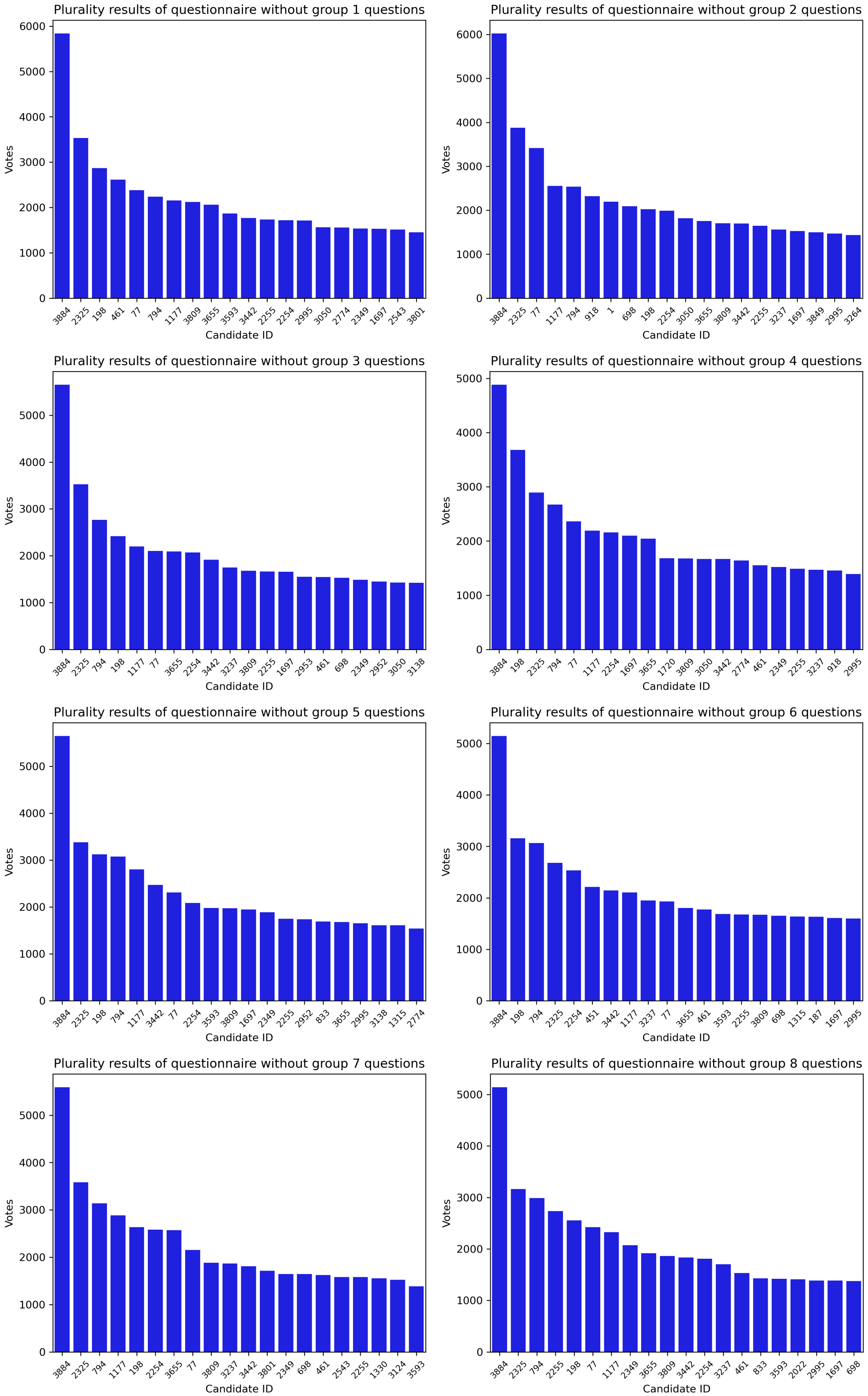


Figure G.1: Plurality winners of election based on questionnaire without questions belonging to groups 1 to 8.

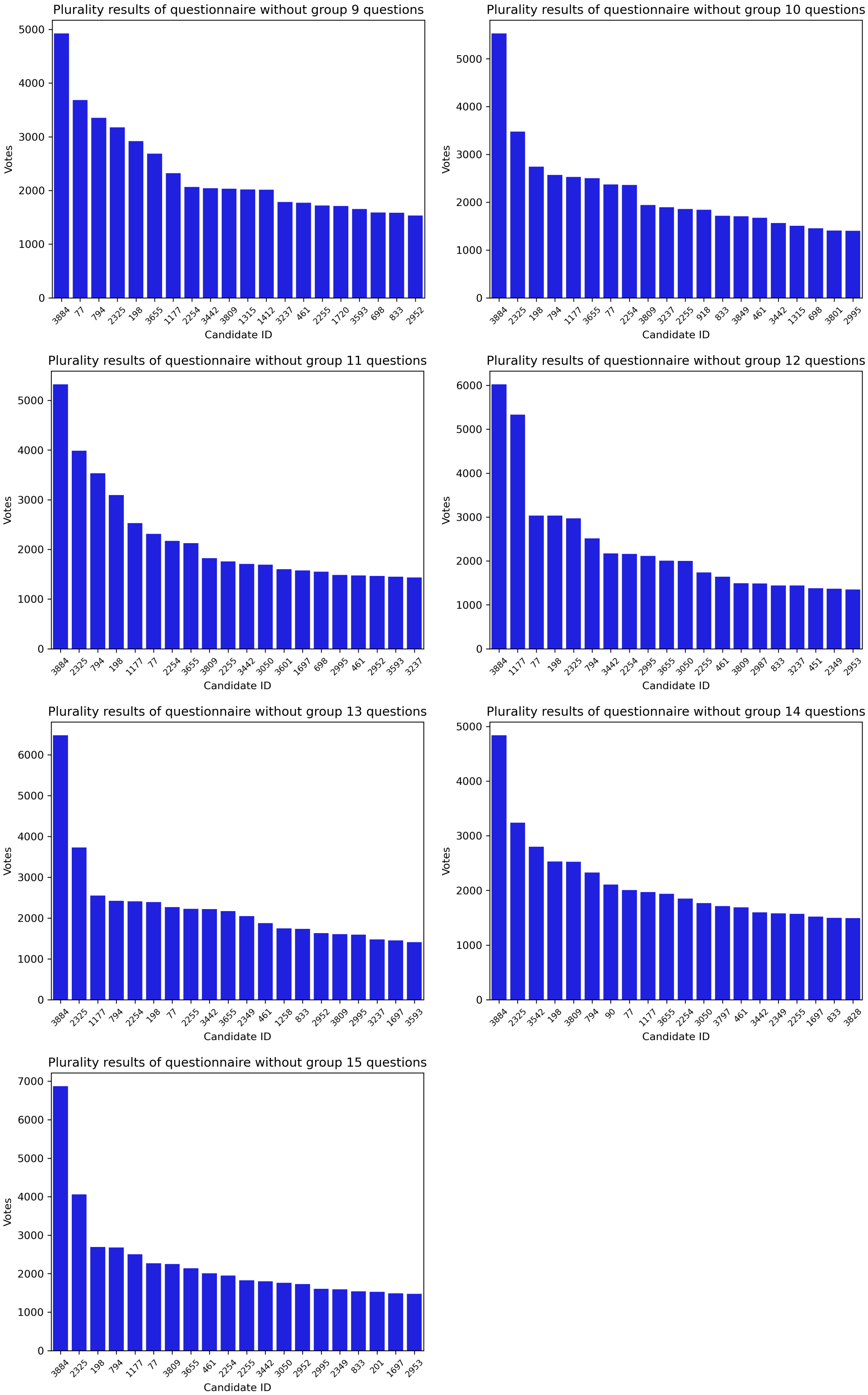


Figure G.2: Plurality winners of election based on questionnaire without questions belonging to groups 9 to 15.