# Evolved Election Forecasts: Using Genetic Algorithms in Improving Election Forecast Results

Track: Real World Applications

Ronald Hochreiter
Institute for Statistics and Mathematics
WU Vienna University for Economics and Business
Vienna, Austria
ronald.hochreiter@wu.ac.at

Christoph Waldhauser
Institute for Statistics and Mathematics
WU Vienna University for Economics and Business
Vienna, Austria
christoph.waldhauser@wu.ac.at

## ABSTRACT

In this paper, we apply a genetic algorithm to the field of electoral studies. Forecasting election results is one of the most exciting and demanding tasks in the area of market research, especially due to the fact that decisions have to be taken in seconds on live television. We show that the proposed method outperforms currently applied approaches and thereby provide an argument to tighten the intersection between computer science and social science, especially political science, further. Numerical results with real data from a local election in the Austrian province of Styria from 2010 substantiate the applicability of the proposed approach.

## Categories and Subject Descriptors

I.6.3 [**Simulation and Modeling**]: Applications; J.4 [**Social and Behavioral Sciences**]: Sociology—*voting studies*; H.3.3 [**Information Search and Retrieval**]: Clustering

## General Terms

Algorithms, Performance, Experimentation

## Keywords

Election Forecasting, Genetic Algorithms, Computational Political Science

## 1. INTRODUCTION

When the last ballots have been cast and the last polling station closes, the fruits of a stressful afternoon are brought to bear: the first election forecast is being broadcast over the air. Much of the work behind it actually took place long before that, starting weeks before the election and culminating shortly after noon [3].

The foundation of election night forecasting lies in ecological regression models [2, 5, 6, 7, 4]. The performance of a party at a current election is considered to be a linear combination of the performances of all parties at a past election. Since not all polling stations report their results at the same time and voters across polling stations will behave similarily, early results can be used to estimate yet missing results during the course of an election night. As the regression coefficients in the used model are not bound to lay within the 0..1 domain, coefficients larger than one and smaller than zero are impossible to interpret. To circumvent this shortcoming of the model, constituencies are grouped together into homogeneous groups. Within these groups the coefficients are more likely to be within the acceptable domain.

The deriving of these groupings is usually produced by experienced senior researchers following their intuition and statistical clues on clever groupings based on k-means clustering and constant size binnings [8]. As a result, this process is very expensive and error prone. In this paper we present a method of deriving (near) optimal groupings using a genetic algorithm that evaluates the performance of a grouping during a simulated election night forecasting. To the extend of our knowledge, genetic algorithms have never been used for this purpose. All other applications of genetic algorithms in the field seek to explain individual voting behaviour and not optimize an election night forecast.

## 2. OPTIMIZATION

When deriving a grouping solution, it is important to bear a few simple rules in mind: Firstly, the groups need to be roughly of equal size. Depending on the number of parties that are to be forecast and the number of parties that contested in the past election that is used in the model, the groups need to contain a minimum number of cases for the regression models to be computationally stable. Secondly, all groups need to contain polling stations that close early, and hence report their results early and polling stations that close later during the day. Otherwise, it will be impossible to forecast using a mix of present and missing data.

In the terms of genetic algorithms, a grouping solution is a chromosome with one gene per constituency. Each gene expresses the constituency's group membership. The fitness function that is optimized compares the predicted election result with the actual result. Two metrics are used for this. One measures the deviation in absolute votes and the other in percent relative to the number of valid votes cast. Both metrics are used as standard in the industry to evaluate the performance of election forecasts. To aid in the optimization, the standard genetic operators mutation, random re-


*GECCO'11,* July 12–16, 2011, Dublin, Ireland.
ACM 978-1-4503-0690-4/11/07.

**Table 1: Parameters of Genetic Optimization**

| Parameter | Value |
|---|---|
| Initial population size | 100 |
| Generations | 500 |
| Elite proportion | 0.1 |
| Reproduction eligible population proportion | 0.7 |
| Mutation probability | 0.003 |
| Random re-seeding proportion | 0.1 |

**Table 2: Deviations of Optimized Solutions**

| Indicator | Human | OptAbs | OptVald |
|---|---|---|---|
| Mean | 2.742 | 0.810 | 3.992 |
| St.Dev. | 4.277 | 1.257 | 6.655 |
| Mean | 0.014 | 0.029 | 0.000 |
| St.Dev. | 0.537 | 0.879 | 0.058 |

seeding of the population and one- and two-fold crossovers are used. We solve this optimization problem by adapting a standard genetic algorithm[1], adapted to peculiarities of the field as described above.

In an experimental setting, the regional election results for the Austrian province of Styria were used, as they are typical for the industry. The unknown election was based on data from the 2010 Styrian provincial election. The forecast was based on the Austrian general election from 2008. The parameters for the genetic algorithm were set according to the values in Table 1. They were established by experimentation.

## 3. DISCUSSION & OUTLOOK

After allowing the solution to evolve over 500 generations, the achieved forecast accuracy was compared with the accuracy obtained by using human based groupings. The quality of an election forecast was established by considering the root mean squared error (RMSE) of the forecast with respect to the actually observed election outcome. RMSE was used in spite of [1] arguing against it. His main critique is the poor performance of RMSE as an indicator in forecasting long-run time series data and its sensitivity to outliers. While this is well founded, it does not apply to the election forecasting problem. Here, the shortest possible time series is used. Furthermore sensitivity to outliers is an asset, since clients and the television audience will be sensitive to them as well.

The results are given in Table 2. This table presents a comparision of deviations between human based groupings and optimized solutions in both metrics as deviations from the true result. The optimizations are results of either optimizing with respect to the total electorate or with respect to the valid votes cast. The former has advantages when voter turn-out is of interest. The latter has more applications in the political realm, as non-voters are ignored—just as in real life. In both optimization processes, in one metric or another, human classification is vastly outperformed by our algorithm.

Predicting the outcome of a ballot on election night depends on the usability of the obtained groupings of the constituencies. We have proposed a way of improving and vastly surpassing manually derived groupings by means of a genetic algorithm.

The logical extension of this paper is the improvement of the real world deployability of genetic algorithms in the field of election forecasting. By employing distributed computing environments that are already available for R, genetic optimization can be used during election night presentations to improve results at an early stage. While the overall result in this use case is not yet known, the target function needs to be modified to optimize the forecast for single polling stations as soon as they are being declared. This constant optimization requires a considerable amount of processing power, but is already well within the capabilities of affordable data center solutions.

The introduction of genetic algorithms by computer scientists to the realm of social scientists is akin to crossing into uncharted territory—for both sides. Despite the numerous obstacles of different communication cultures and epistemological propositions, it is an endeavor worthwhile. We hope that this paper is a contribution to building a bridge between what once had been termed incommensurable.

[1]The algorithm was implemented using [9].

## 4. REFERENCES


[1] J. Armstrong. *Principles of forecasting: a handbook for researchers and practitioners*. Kluwer Academic Publishers, 2001.

[2] P. Brown, D. Firth, and C. Payne. Forecasting on British election night 1997. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 162(2):211–226, 1999.

[3] S. Fienberg. Memories of Election Night Predictions Past. *Chance*, 20(4):8, 2007.

[4] J. Greben, C. Elphinstone, and J. Holloway. A model for election night forecasting applied to the 2004 South African elections. *Journal of the Operations Research Society of South Africa*, 22(1):89–103, 2006.

[5] C. Hofinger and G. Ogris. Orakel der Neuzeit: Was leisten Wahlbörsen, Wählerstromanalysen und Wahltagshochrechnungen. *Österreichische Zeitschrift für Politikwissenschaft*, 31(2):143–158, 2002.

[6] G. King. *A solution to the ecological inference problem: Reconstructing individual behavior from aggregate data*. Princeton Univerity Press, 1997.

[7] G. King, M. Tanner, and O. Rosen. *Ecological inference: new methodological strategies*. Cambridge Univerity Press, 2004.

[8] J. MacQueen. Some methods for classification and analysis of multivariate observations. *Proceedings of 5th Berkeley Symposium on Math. Stat. Probab., University of California 1965/66*, 1:281–297, 1967.

[9] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2010. ISBN 3-900051-07-0.