Exploring FEC Data

https://www.github.com/khaozavr/SPL_project_WS1617

Janek Willeke Andrii Zakharov Jonas Klein Lukas Mödl

Berlin, March 15, 2017

This seminar paper is written in the course Statistical Programming Languages at the Humboldt-University Berlin. In this paper we engage with the publicly available data of the Federal Election Committee (FEC), which contains information on individual donations to presidential candidates. We add characteristics to the dataset and use these to conduct further analysis. All of our self-written code can be found on Quantnet.

Contents

1 Introduction

The year 2016 was marked by one of the most interesting presidential election campaigns in the history of the USA. Not only did it produce an unprecedented level of polarization in the American society but it culminated in the generally unexpected outcome of Donald Trump winning the presidency. In this project, we decided to look at the openly available Federal Election Commission (FEC) data to try and find signs of this strong political turbulence in financial contributions to different candidates' campaigns. We wanted to ascertain not only how this surprising outcome came about, but also if it could have been predicted.

In 1975, Congress created the FEC to administer and enforce the Federal Election Campaign Act – the statute that governs the financing of federal elections. The duties of the FEC, which is an independent regulatory agency, are to disclose campaign finance information, to enforce the provisions of the law such as the limits and prohibitions on contributions, and to oversee the public funding of Presidential elections. The data on public funding is openly available and can be downloaded at www.fec.gov.

To keep the dataset manageable, we decided to focus on one state. We picked California, as it is the biggest state in the union and more relatable for most people on this side of the Atlantic. The political polarization of Los Angeles, a city quite often depicted in Television and film, might spark bigger interest than political differences across Colorado. Yet one has to keep in mind that California is a so-called "blue" state. It has overwhelmingly voted Democratic in recent federal elections. This entails that some of our result might not be applicable to the nation as a whole. But in a nation so divided it is hard to find a state that is truly representative of it as a whole. Still, for further analysis, the code used in all the Quantlets could also be applied to the datasets of the remaining states.

Unfortunately, the donations dataset does not contain all donations made in support of the candidates, as only donations to the candidate committees are recorded. Donation amounts are capped i.e. only up to $2700 can be donated by an individual to a candidate's political committee (FEC, 2017). Yet if a person or corporation wants to support a candidate with an amount greater than that they may do so by founding or funding a Super-PACs, which may support a candidate by running ads for her amongst other things (Krieg, 2017). In the dataset there is no information about the donations to these Super-PACs. Yet still, we consider the dataset to offer vast opportunities for social research. Few datasets are of comparable size. Also, there is no need to collect the data, as it is publicly available. Additionally, within the limits of the donation amounts, the data show all the donors, as no sampling took place.

Our first step was to prepare the data. Not only did it prove to be messy, such that parts of it could not be properly read in by R. It also holds a lot of hidden information that had to be extracted first. This represents a big part of the work done in this paper.

In working with the prepared data our goal was twofold: First we wanted to characterize the interplay between donation behaviour, party allegiance and sociodemographic factors. This was

achieved by means of basic descriptive statistics, more advanced visualizations using ggplot2 and the variable importance data and partial dependence plots provided by the randomForest algorithm.

In the second part we wanted to ascertain how feasible a prediction of donation behaviour is, given the limited amount of data we have. For this we also trained a random forest, but used only the donor characteristics not related to their donation, i.e. we removed both donation amount and donation date as predictor variables.

Overall, our goal in this course was not only in analysis of political data but in the application and improvement of our programming abilities. We wanted to get our hands dirty on messy real-world data not often used in university courses but quite common in scientific and commercial applications. In doing this, our goal was working with the data while working on our programming skills.

2 Data Preparation

In this section we provide a description of the cleaning process of our dataset. Subsequently we add further characteristics to our data. Each step is explained in one of the following subsections. Please note that the order which we chose here is also important for our Quantlet structure. Each Quantlet produces a new dataset. In order for our Quantlets to work, you always need the final dataset of the previous Quantlet. This applies to Quantlet 1 (FECAdataprep) through Quantlet 5 (FECAaddoccuclass). Quantlet 1 starts with the original FEC dataset, which is open for downloading here.

2.1 Prepare the Data

FECAdataprep Due to the fact that the data file contains a lot of messy data, a big part of our project was to clean the data set and shape it into a usable format. The data file was not in the correct format (`.csv`) as a missing comma in the first row of the data set caused an error when using R's read.csv function. We solved this problem by using read.table instead with the options header = FALSE and fill = TRUE, deleting the emerging column and moving the first row back into the header. After loading the data set into R, we proceeded as follows:

- Removing all unnecessary and irrelevant variables.
- Rename the remaining variables in order to make further coding more convenient.
- Transforming the reported zip codes, as in the dataset, zip codes were not kept in a coherent format: The variable contbr_zip takes on 5 digit values corresponding to the actual zip codes in California, or 9 digit values, where only the first five digits corresponded to actual zip codes. To be able to analyse zip code data we decided to keep only the first five digits of the values. Moreover, there were zip codes that did not fall into Californian

terrain. We dropped all cases not corresponding to Californian zip codes, which range between 90000 and 96162.

- Remove first names of candidates and add the candidates party (Democrats, Republicans or Third Party if a candidate did not belog neither to Deomcrats nor to Republicans).
- Adjusting variable types and omit invalid cases.

After all those transformations, one issue remained: Some contributions were negative. Due to the structure of the FEC, individual donors are allowed to donate money, but also to request refunds of their donation amounts. Those requested refunds are marked in the donation dataset with negative donation amounts. The best solution to this problem would be to delete both the donating and the respective refunding case.

Unfortunately this was not possible, as it is hard to match corresponding cases. Probably due to the fact that donors self report lots of the data and that some of the donor information changed between time of donation and refund. We decided to omit all cases with negative donation amounts from our data set. This might not be the cleanest solution, as it somehow grows our donation amount in the dataset. However, the negative donation amounts/requested refunds do not make a huge difference. That is why this straightforward approach proved to be the best solution at the time eventually.

2.2 Add the Gender of Contributors

FECAaddgender The FEC data has a lot of useful information on the contributors, but one interesting variable that it is not recorded is their gender. Particularly for the 2016 presidential race, with the first female favourite in history, one might suspect some gender-specific bias in donor behaviour. So we decided to try and infer the gender of donors from their first names. To this extent, we utilised lists with most common American first names for males and females, as seen in the US Census data (the lists were downloaded here). We used R's native string manipulation tools to extract first names from full names, and checked whether they appeared in the lists. The checking operation perfectly showcased R's strengths in vectorised operations and boolean indexing. We first compared the vector of first names from our data with the male and female lists respectively, using the `%in%` operator. This produced two test vectors of booleans, one for males and one for females. We then initiated an empty vector, and filled it with gender markers using logical combinations of both test vectors and R's boolean indexing with which function. The resulting vector contained values `"male"`, `"female"`, `"both"`, and `"neither"`, in the correct order for the donors in the data.

So the gender was assigned according to list appearance. However, a large number of first names either appeared in both lists (e.g. Alex), or in neither (e.g. Vsevolod). These cases were assigned `"both"` and `"neither"` values. Thankfully, the US Census lists were ordered by popularity (i.e. frequency of appearance in the population). This allowed us to get rid of the `"both"`

values in our gender vector, by going through the two lists once again and assigning "male" to those first names that appeared higher on the male list than on the female, and "female" otherwise. Note that this approach strongly resembles that of a Naive Bayes classifier – a simple machine learning technique that classifies data into binary classes according to their probability of occurring. In our case, relative frequency in the population served as a proxy for this probability. If the name had exactly the same probability of being male or female, it was assigned "female" class, to reflect a slightly higher proportion of females in the US population (0.97 males per 1 female, according to the CIA's World Factbook). Of course, such a naive approach was bound to misclassify a number of people, but lacking additional information it was good enough for our purposes. This crude inference operation was executed using a for-loop, which made it blatantly obvious how much faster R's vectorised operations are in comparison. We aimed to only use those whenever possible from that time on.

Unfortunately, getting rid of the "neither" values in the gender vector could not be done with the same degree of plausibility. Most of those were foreign or exotically spelled names, which made them substantially harder to binarize into genders. We had no other choice but to leave them as is.

As the last step of this quantlet, and to facilitate further analysis, we assigned genders to the presidential candidates themselves. This was a simple manual operation, as there were, sadly, only three women running.

2.3 Neighbourhood Wealth Effects

FECAaddincome Neighbourhood effects have been studied extensively in economics and the social sciences, see for example Galster, Andersson, and Musterd (2010) for a literature review. In the analysis of voting behaviour different aspects have been focused on. Prominent topics were social interaction, see for example Miller (1977) but also the relative economic and social standing of the neighbourhood, see Johnson, Shively, and Stein (2002). We would expect the economic constitution of the neighbourhood to influence voting behaviour and therefore donation behaviour as well. For the United States, there is a good amount of data supporting the claim that regional wealth distribution predicts voting behaviour. Surprisingly, people from rich congressional districts tend to vote democratic (Edsal, 2017). Neighborhood wealth in our case can also function as a proxy for individual wealth, as we do not have any knowledge about the donor's wealth, but do know their zip code.

Our estimates of neighbourhood wealth are based on tax return data provided by the Internal Revenue Service (IRS), the taxation agency of the United States. It shows tax return data, classified by state and zip code. Here, we assume the zip code area to be the neighbourhood, which may not be completely accurate in more rural areas because of the varying zip code size (Grubesic, 2008). The data is available for each state as an individual file, we only used the Californian file. The data is based on the individual income tax returns collected by the IRS

(IRS, 2017). A number of studies thus far have used IRS individual tax return data, mostly while studying income inequality, see for example Burkhauser et al. (2012). For our paper it seems sensible to base our estimates of the income distribution in the zip codes on this data, as such data is not and cannot be feasibly collected in social surveys, and IRS return data is very reliable (Burkhauser et al., 2012). Income in our case will be measured by the adjusted gross income, which is the gross income minus specific tax deductions. As this factors in job related expenses, such as income deductions for business expenses or e.g. expenditures for further training, it can be thought of as the actual income pre tax of the individual. Adjusted Gross Income (AGI) in the IRS dataset is grouped into six groups, for each of these groups the AGI is reported alongside the total amount of AGI over all groups.

The IRS dataset proves to be very messy. In its original state it cannot be properly processed with R. First of all, not all variables are clearly designated, so that names have to be inserted first. This can only be done since the missingness of the names is systematic. For the purpose of transforming and reading in the data, it is conceptualized as being in long format, i.e. in the format most commonly used by panel studies. It is then transformed to wide format, thus allowing R to identify the income groups as variables. This is achieved by using the reshape command:

```
irsgai_wide = reshape(irs.gai, idvar = "CALIFORNIA", timevar = "X", direction = "wide")
```

Afterwards, variables are renamed for the purpose of convenience. Redundant columns are removed, making it easier and more efficient to work with the dataset. A few more minor adjustments are made. For example, thousands separator in one column are removed and placeholders for 0 are replaced with an actual 0. Then, the amount of AGI in each group in the zip code is divided by the total amount in the given zip code, thus transforming the agi data into proportions. Afterwards, income distribution in the zip code is computed as a score, by weighing the income categories and multiplying the percentage in the categories with the weights, ranging from 1 to 6. Afterwards, the thus computed score is normalized by dividing it by 6. It then ranges from 0 to 1. At last, the income distribution is matched to our data set.

2.4 Area Party Affiliation

FECAaddarea Besides the effect of wealth distribution we also tried to capture the effect of party affiliation of one's neighbourhood. Therefore we created new variables which assigned R for republican or D for democrats based on either the total sum of individual contributions per zip or the frequency of individual contributions per zip code.

We used dummy variables to determine which candidate got the largest total sum of donations and the most frequent number of contributions in each zip code. Subsequently we selected only those candidates with the highest values and thus could match this to every unique zip code. This approach is quite different to the previous Quantlets we have presented

so far, as this time we do not add additional data, but rather gain further information from insight the dataset.

2.5 Classify Occupation

FECAaddoccuclass Occupation is a powerful characteristic which allows for a lot of analysis in combination with the donation information. Unfortunately, in the FEC data set occupation is self reported, which led to a total of 27410 unique occupations. If you want to analyse aggregate data, you have to reduce the dimension of occupation. Fortunately, there exists a script, not for R though, which classifies the different occupations.

We used the existing classification keywords and wrote a script which compares the keywords with the self reported occupations in the FEC data. Thus we extracted the keywords into a new data frame occuclassed. The keywords are classified into 28 occupation classes, e.g. "Education, Training and Library" or simply "Retired". Subsequently, we wrote a little for-loop which compared every self reported occupation of each contributor of the FEC data with the keywords in the data frame. Depending on the match, it assigns the respective occupation class to the contributor. To prevent the issue that some occupations could not be assigned to an occupation class, we created the class "Other" and assigned this class to the respective cases.

3 Summary Statistics

In this section, several descriptive statistics are presented for some of the variables in the data set.

3.1 Contribution Amount

Table 1 contains standard summary statistics for the four major candidates Clinton, Cruz, Sanders and Trump, as well as aggregated statistics for all remaining candidates. Besides these summary statistics, the zip codes in which the respective candidate received the contributions most frequently.

Table 1: Summary statistics for contributions to the four major candidates.

Cand.	Tot.	Mean	Med.	Std. Dev.	Min./Max.	n	Frequent Zip Codes
Clinton	95m	139	25	418	0.01/10000	681k	94611, 94110, 94114
Trump	14m	171	80	351	0.80/5400	83k	92067, 92037, 92660
Sanders	20m	51	27	119	1.00/10000	404k	94117, 94114, 94110
Cruz	7m	125	50	472	1.00/10800	56k	90274, 92637, 92705
Other	21m	340	50	760	0.01/10800	62k	90272, 90049, 92660

FECAsummarystatistics

The total sum of donations reflects really well that California is a democratic state: The donations to Clinton and Sanders surpass the donations to Cruz and Trump by a factor of 4.5, not counting donations to candidates other than these four.

Taking a closer look at Clinton, we find that she received the most money as well as most individual donations. Interestingly she has the lowest median of $ 25. Summing up these observations, Clinton received many donations both low value and high value. This is also reflected in the relatively high standard deviation of $ 418.

The donations to Cruz follow a somewhat similar pattern, having the highest standard deviation of the single candidates.

Sanders is a real donation hunter. Although he could not run for presidency eventually, he received more donations from individual contributors, than all other candidates combined, excluding Clinton. However contributions were lower than contributions to the other candidates. Nevertheless, the frequency of donations make up for the overall low individual contributions.

Really interesting is the fact, that one of the top three zip codes from which Trump received

most donations is located in San Diego. As San Diego is next to the border to Mexico, this could be related with Trumps recurring demands for building the wall to Mexico. Of course this hypothesis would need some further investigation.

3.2 Distribution of Contributions to Candidates in California

We now consider the zip codes. The zip codes allow us to draw conclusions on the geography of political donations. Thus we want to draw a political map which illustrates the geographically origin of the donations. We want to assign a zip code to the candidate who has received the largest amount of contributions in that zip code. To draw the map we use R's ggmap package .

Zip codes can represent small and well-defined regions of households with similar lifestyles (Grubesic, 2008). This holds especially true for urban areas, where zip code tend to represent very compact areas. For rural areas, this might not be true. This also means that zip codes vary in size. They correlate with population size, they are however not fixed by population size and do not conform to political boundaries (Grubesic, 2008). Zip codes do not conform to discretely bounded geographic areas and are not available for all regions of the country. Unpopulated areas or those with a very small population might not have a zip code assigned to them. That means one could assign longitudial and latitudial data to a zip code but that would be just one single point on a map and no polygon as one might expect. To approximate zip code polygons, we use the ZIP code tabulation areas (ZCTA) for California. Unlike zip codes, ZCTAs are specific geographic regions that can be codified and listed exactly. ZCTAs were created by the US Census Bureau to allow them to have a specific numeric system for where respondents are located. The here used ZCTAs are based on the 2010 census which are available here for download. Note that the ZCTA come in a shapefile which consists of a folder (`tl_2010_06_zcta510`). To load the shapefile into R, the readOGR command from the package rgdal is used:

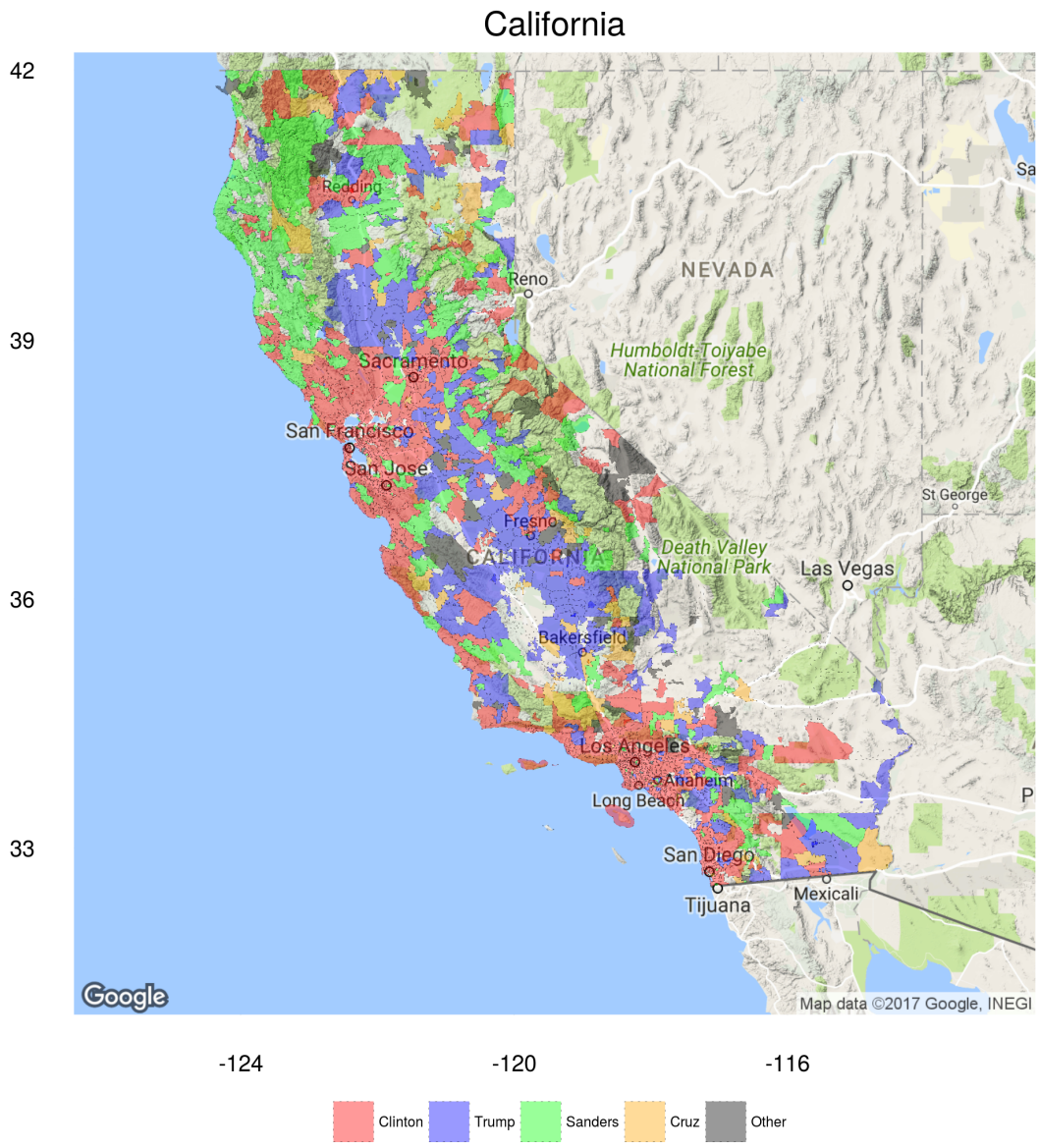
```
shp = readOGR("tl_2010_06_zcta510","tl_2010_06_zcta510") %>%
    spTransform(CRS("+proj=longlat +datum=WGS84"))
```

The shapefile data is transformed (second line) to match the settings used by the Google Maps API since we want to plot the zip code polygons onto a Google Map image. If this transformation is not done, the polygons would be drawn at the wrong positions, e.g. a neighborhood of San Francisco could be drawn in the middle of a water area. After merging our prepared data set to the (transformed) shapefile data, ggmap is used to draw the approximated zip code polygons onto a Google Maps image (mapca = "california"%>% get_map(zoom = 6)) of California,

```
plotca1 = ggmap(mapca) +
    geom_polygon(aes(long, lat, group = grp, fill = area1_cand), mapdat,
        alpha = 0.4, color = "black", linetype = "dotted", size = 0.05) +
    scale_fill_manual(values = c("red", "blue", "green", "orange", "black")) +
    theme_minimal()
```

The variable mapdat is the merged data set and the variable area2_cand is the candidate with the largest amount of contributions in the corresponding zip code. Note that we focus only on

the in our opinion most relevant candidates: Clinton, Trump, Sanders and Cruz and refer to the remaining candidates as "Other". The rest of the code snippet is pretty self-explanatory.

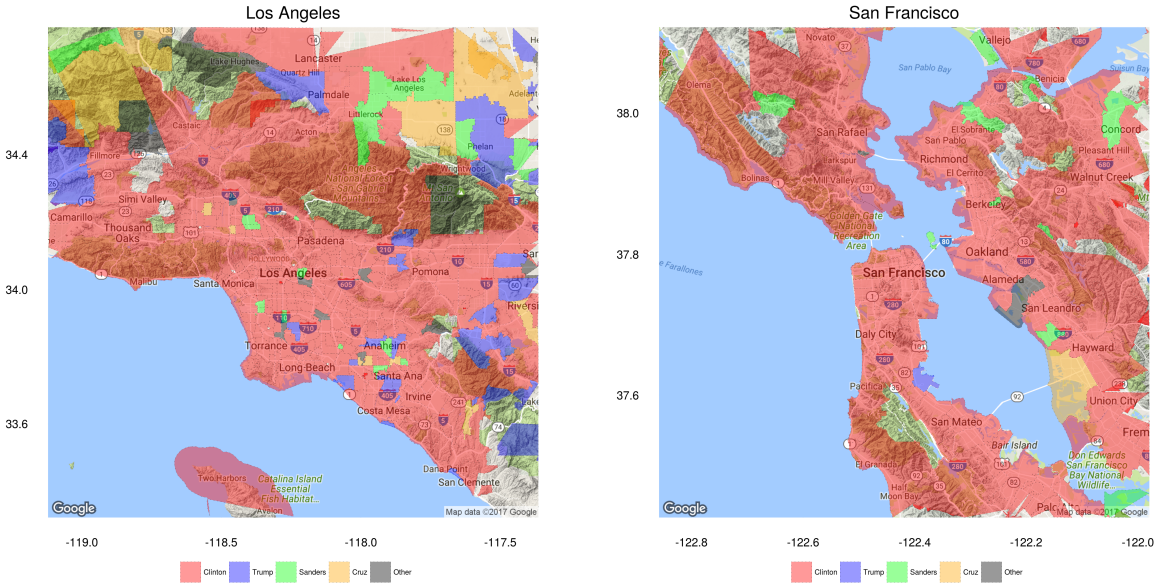


FECAmap

Figure 1: Map of California with zip code polygons approximated by the ZCTA values. The polygons are colored according to the candidate who received the largest amount of contributions in the corresponding zip code.

Figure 1 shows the map of California with the upon drawn polygons. Not surprisingly, Clinton received the largest amount of contributions, especially in and around the metropolitan areas of Los Angeles and San Francisco. The Republican candidates Trump and Cruz on the other hand received the largest amount of contributions in the rural outbacks and in zip codes close in south of Los Angeles, Interesting is that in northern California Sanders received in Northern California the largest amount of contributions in quite many zip codes. Besides these findings,

no specific patterns could be found.



FECAmap

Figure 2: Map of San Francisco and Los Angeles metropolitan area with zip code polygons approximated by the ZCTA values. The polygons are colored according to the candidate who received the largest amount of contributions in the corresponding zip code.

Figure 2 "zooms" into the metropolitan areas of Los Angeles and San Francisco. However, no specific neighborhood effects can be found: Clinton received the largest amount of contribution in almost all regarded zip codes.

3.3 Candidates and contributors gender

As mentioned previously, influence of gender on politics is still a controversial discussion. Especially since Clinton has run for the US presidency as first female candidate for one of the major parties. In this section we want to further analyse whether there is indeed a gender effect, i.e. do female contributors prefer to donate to female candidates?

Table 2: 2×2 contigency table of candidate and contributor gender.

	Candidate Female	**Candidate Male**
Contributor Female	388978	231764
Contributor Male	225462	302054

FECAgenderphi

Table 2 shows the contingency table of the candidates and contributors gender. Note that we omit the observations for which the gender could eventually not be determined. In absolute

numbers, more female contributors contributed more to female candidates (388978) than to male candidates (231764). The same holds true for male contributors: 302054 contributed to male candidates and 225462 to female candidates. To measure the association between candidates and contributors gender, we use the ϕ-coefficient. The ϕ-coefficient is a common and often-used quantity to measure such "categorial correlations". It is given by

$$\phi = \frac{n_{11}n_{22} - n_{21}n_{12}}{\sqrt{\prod_{k=1}^{2}\prod_{\ell=1}^{2}(n_{k\ell} + n_{\ell k})}},$$

where n_{ij} denotes the ij^{th} element of the corresponding 2×2 contigency table. The ϕ-coefficient (of a 2×2 contingency table) lies between -1 and $+1$. Low absolute values refer to no or only a weak association between the regarded categorial variables whereas high values refer to a strong (positive or negative) association. In our case the value of the ϕ-coefficient is $\phi = 0.199$, i.e. the association between candidates and contributors gender can be considered as rather weak. In other words, female (or male) contributors seem not to favor the own gender of their designated candidate.

4 Inferential Statistics

As our goal in analysing the data was not only descriptive but also inductive, we wanted to build a predictive model, while also drawing conclusions on the influence of certain predictor variables on donation behaviour. Due to the large number of observations, classical statistical inference might be misleading (as even negligible small effects are statistically significant). We therefore do not rely on statistical tests such as t- or χ^2-tests but use machine-learning techniques instead to analyze the data set. We chose to use the random forest algorithm due to its predictive power. The predictive model was built using R's caret package, as it has a large array of features that are important in predictive modeling. A separate random forest model was built using the randomForest package, which allows us to model the effect of a given variable on the response variable using partial-dependence-plots and to draw conclusions about the relative importance of the predictor variables with variable importance plots.

4.1 Random Forest Algorithm

Random forest rank among the most popular algorithms currently used in the machine-learning world. They can be best thought of as an ensemble of decision trees, each built with only a fraction of the variables (James et al., 2013).

Decision trees are tree-like models that partition the data into smaller and smaller subsets based on a purity measure, with the Gini index and cross-entropy being the most common measures. At each split, the variable minimising the measure is chosen to split on. Most commonly, there are stopping rules, after which no new splits are formed. Decision trees

tend to work well with non-linear relationships and are easily interpretable. However, they are non-robust, meaning that a small change in the data can lead to a much bigger change in the model (James et al., 2013).

Random forests are a way of addressing this issue while also improving predictive power. Instead of just growing one tree, an ensemble of trees is grown. Each of these trees is grown on a subset of the data and at every partition, the tree may only choose from a fraction of the variables. This leads to a higher variance among the trees (James et al., 2013). Afterwards, all the trees are combined by voting, i.e. the prediction is based on the y-values predicted by the majority of trees. Random forest are thus easy to implement, since only the number of trees and the number of variables at each split has to be determined. While the number of trees is only limited by computational resources, the number of predictors at each split is usually chosen as the square root of the number of predictors in the model.

A major issue we had while building the predictive model was class imbalance. Class imbalance refers to a response variable that is not equally distributed. If the imbalance is severe, it may influence prediction (Chawla et al., 2002). In the reduced donation dataset, approximately 90% of the contributions received democratic candidates. This would lead to the random forest algorithm classifying almost all of the observations as Democrats, a trivial result. In order to let the algorithm place higher importance on the minority class we decided to resample the dataset. We undersampled the minority class to achieve an equal occurrence of observations in both classes.

The caret package allows us to specify a vector of possible values of variables considered at each split. This is why we chose to set a range of values around the square root of the number of variables in our predictive model. Afterwards, a decision tree was grown using the already partitioned dataset and the twowaysummary as the performance metric, which will compute the sensitivity, specificity and auc measures.

To better assess model performance during training, we used cross-fold validation. In this procedure, only a part of the training set is used for model-building and then the remaining part is predicted using the hold-out set. This procedure is repeated as many times as there are folds. Using 5 to 10 folds is common, the higher the number the lower the bias (Hastie, Tibshirani, and Friedman, 2009). To achieve an even lower bias, this procedure was repeated 5 times:

```
rf.caret = train( party ~ area2_party + gender + class + distr,
    data = train,              # dataset used, here: the train data defined above
    method = "rf",             # method used, random forest in this case
    ntree = 500,               # specifies number of trees grown
    tuneGrid = rf.parms,       # the tune grid used, see above
    metric = "ROC",            # specifies the metric for model performance
    trControl = model.control # the model.control parameters defined before
)
```

Our trained model was then used to predict the remaining cases of our test data using the predict function in R.

To measure performance of the trained models we used the auc metric computed by caret. It measures the rate of true positives versus true negatives over all thresholds and is thus sensitive to class imbalance. For the trained model, the auc is 0.729 when considering 3 variables at each split and 0.734 when considering 4 variables. Auc values range between 0.5 and 1, a value of 0.734 can thus already be considered good in performance.

As it optimizes the auc metric, the model with 4 variables considered at each split was chosen. That means that each tree of the random forest is grown with all variables available and variance is only introduced by varying the samples the trees are grown on. This deviates from the general suggestion to grow a tree with only a fraction of the variables available, but seems sensible given the better performance of the model.

To better assess model performance we attempted to predict the party donated to, of the remaining observations in the test set. To this end we used the predict function available in caret. In the prediction we used a threshold of 0.5 for the prediction as we had no clear preference which of the classes we wanted to predict correctly. If instead we wanted to correctly predict e.g. Republicans we could vary the threshold so that the rate of correctly classified Republicans is higher while not considering Democrats.

As can be seen in the confusion matrix (Table 3), predictions are far from optimal. The accuracy, which measures the number of correctly classified Democrats relative to population size and ranges between 0 and 1 is 0.672. This is not perfect, but much better than randomly assigning classes. These are decent results keeping in mind the quality of the variables used.

Table 3: Confusion matrix of model results.

	Reference	
Prediction	**Democrats**	**Republicans**
Democrats	143600	12073
Republicans	68558	25565

FECArandomforest

Our focus was on using information readily available to political analysts that is not directly related to the donation. Thus, contribution behaviour of persons that have not yet donated could be predicted. It seems impressive that with only the occupation, zip code of residence and the name of a person available we could run predictions of their contribution behaviour that are far above randomly assigning classes.

4.2 Variable Importance

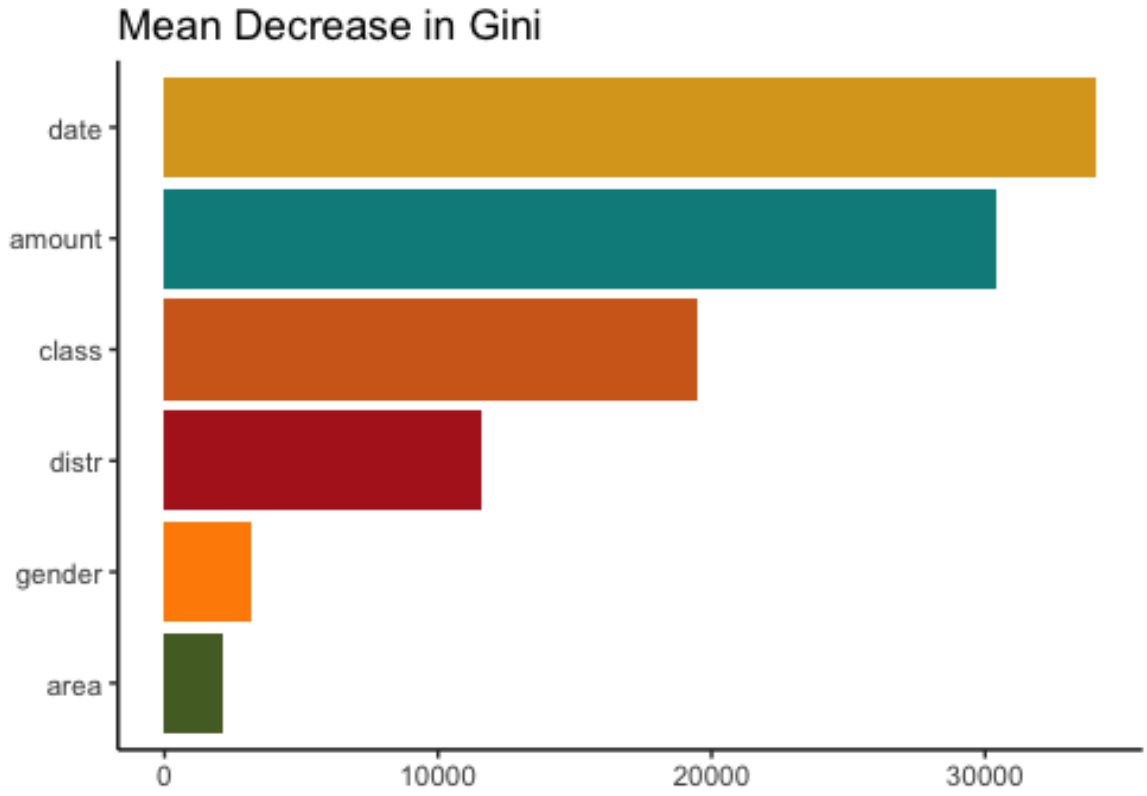
For the partial-dependence plots and the variable importance measure we instead opted to use the default parameters of the randomForest package. This was done for reasons of computational

efficiency, since predictive accuracy was not as important in this case.

```
partial.rf = randomForest(party ~ amount + gender + class + distr + date + area2_party, data =
    dat_unders)
```

Variable importance measures allow us to estimate the relative importance of the predictor variables in the model. The random forest algorithm assesses the importance of variables by permuting out-of-bag data for a giving variable and describing how much prediction error increases. Out-of-bag observations are the on average one-third of the training set not used for a given tree, due to bootstrap sampling. The Out-of-bag importance measure thus provide a robust statistic of the variable's importance in the model.

To gain an insight into the relative importance of the individual predictor variables we looked at the mean decrease in gini computed by the randomForest package. The most important variables in the model turn out to be the date and the amount of the donation, with date being the most important variable. Apparently, knowing when a donation took place is paramount in knowing to whom the donation goes. This seems intuitive, as Trump received campaign donations only much later in the race while Sanders stopped receiving donations after he did not turn out to be the Democratic candidate.



FECArandomforest

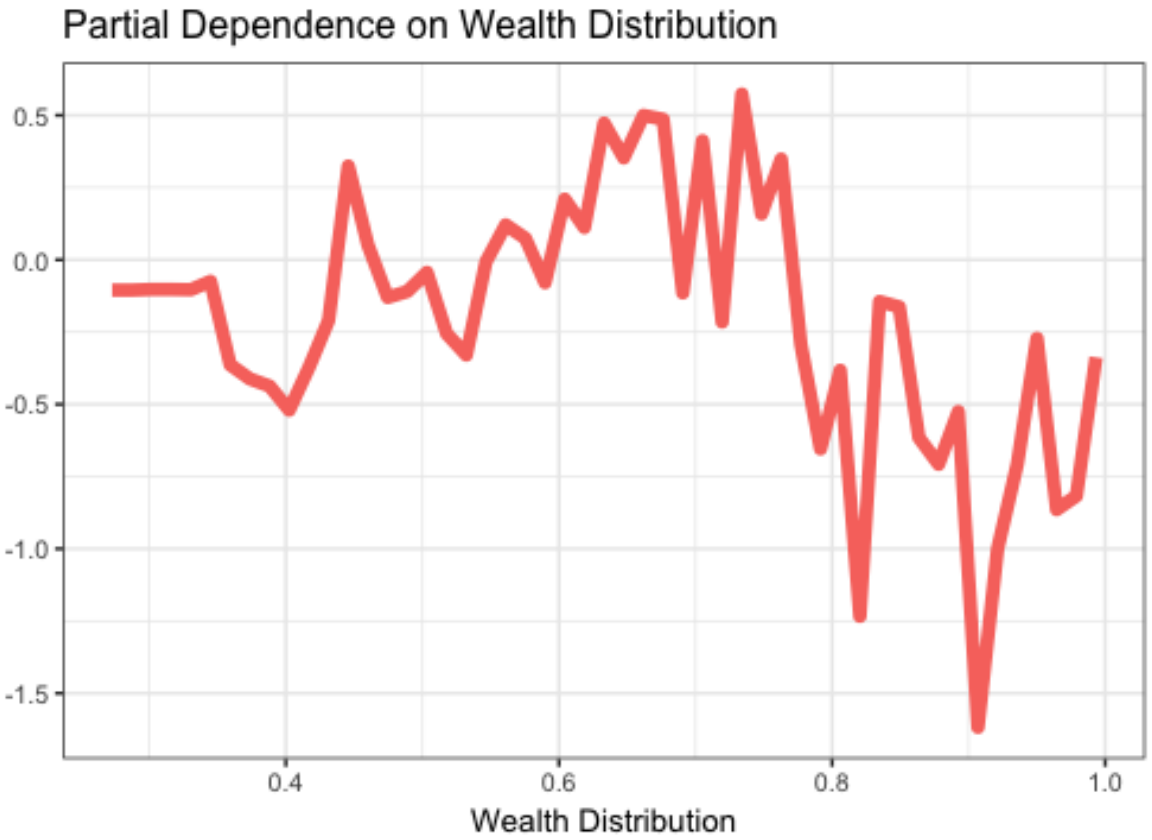
Figure 3: Gini measure for variable importance.

According to the model, knowing how big the donation was tells you a lot about the target of the donation. This seems interesting and will be further investigated using a partial-dependence

plot in the next section. The occupation of the donor is the next important variable, while the variables we added fare worse. Only the wealth of the zip code is somewhat important, while the gender of the donor and the party most people in the area donated to turn out to be only of minor importance. This seems counter-intuitive, as knowing how the majority of the population in a zip code voted should be a powerful predictor of an individual's voting behaviour, at least in zip codes with a strong tendency to vote for either party. It might be that the undersampling made the variable less informative.

4.3 Partial Dependence plot

The partial-dependence plot depicted below shows the marginal effect of the zip codes wealth on the propensity to vote Republican. It has a high variance, which could be due to the low number of observations in some of the zip codes. Overall and according to the model, there is a tendency for people from middle income zip codes to vote Republican, with middle income being between 0.4 and 0.8. People from zip codes outside of that range have a strong tendency to vote Democratic, with very high income zip codes having the strongest tendency. This confirms previous findings, e.g. Edsal (2017).



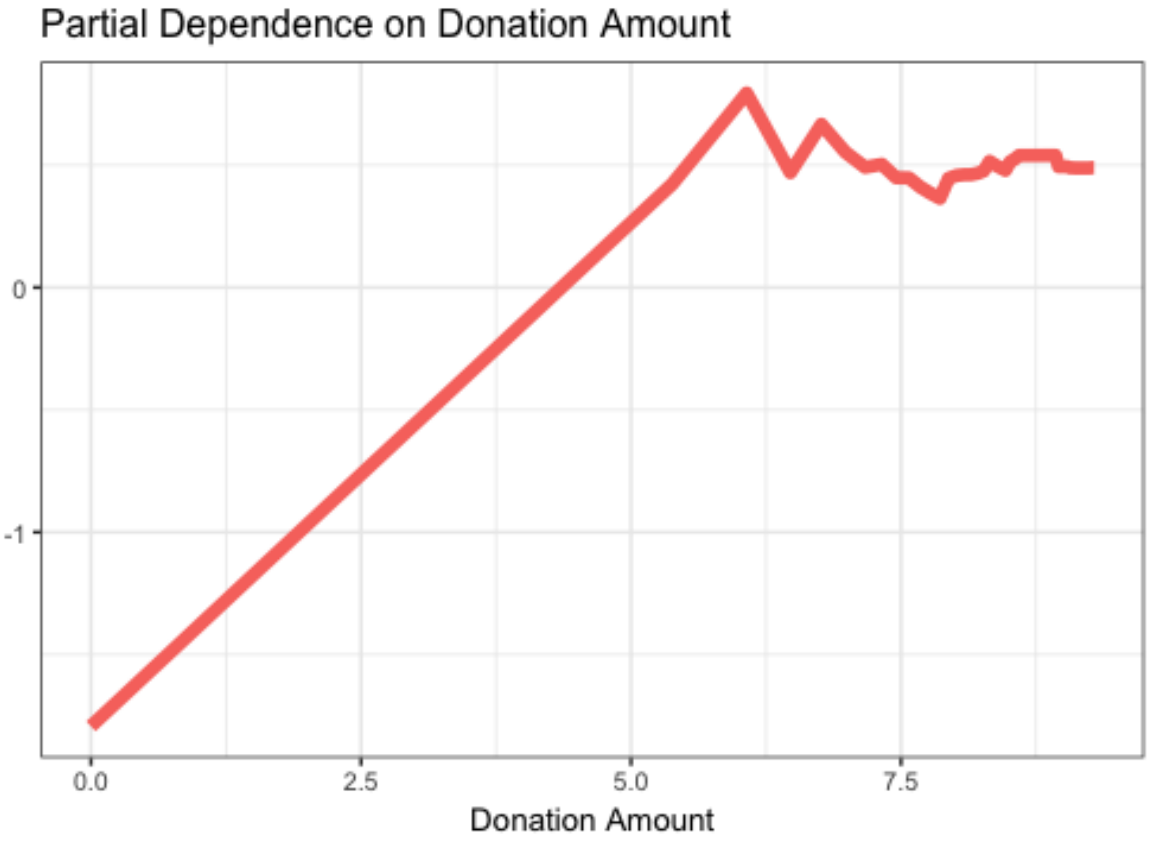
FECAvariableimportance

Figure 4: Partial dependence on wealth distribution.

While our variable importance measure pointed to the donation amount being of immense

importance for the model, we had no clear idea what the direction of the effect could be. This is why we also looked at the partial-dependence plot for the donation amount. As most donations were in the low range, with some going up to $ 2700, we decided to log-transform the donation amount to gain a better understanding of the effect in the low ranges. There is a clear linear trend. The higher the donation amount, the more likely a person to donate to a Republican candidate according to the model. This falls in line with the descriptive analysis of the data.

While our variable importance measure pointed to the donation amount being of immense importance for the model, we had no clear idea what the direction of the effect could be. This is why we also looked at the partial-dependence-plot for the donation amount. As most donations were in the low range, with some going up to $ 2700, we decided to log-transform the donation amount to gain a better understanding of the effect in the low ranges.



FECAvariableimportance

Figure 5: Partial dependence on logarithmized donation amount.

There is a clear linear trend. The higher the donation amount, the more likely a person to donate to a Republican candidate according to the model. This falls in line with the descriptive analysis of the data.

5 Final Remarks

In this seminar paper we have worked with, enriched and analysed the FEC dataset which includes data on individual contributions to the US presidential candidates. We have described the process of our data preparation as well as the integration of further characteristics to the dataset, like gender, income distribution, party affiliation and classified the self reported occupations. We further build and analysed a political map of donations in California which supports the claim that rural areas tend to vote for Trump. We have shown that there is no evidence that gender plays a decisive role when deciding for a presidential candidate.

In the last section we captured the importance of various variables on donation behaviour and build a random forest model which predicts the likelihood of potential donors based solely on publicly available data. The good thing about this project is the opportunity for more and more research. The FEC data is continuously growing and methods to analyse those huge datasets are developing quickly. For further research it would be nice to improve the prediction power of our model. This could be done by trying and implementing different machine learning algorithms as well as trying new methods. For example it would be interesting to try cluster analysis on this data.

Summarising our intentions for this seminar paper, it is fair to say that we reached our goals. We did not only manage to find, work with and analyse publicly available data, but we also improved our programming skills.

References

Burkhauser, R. V., S. Feng, S. P. Jenkins, et al. (2012). "Recent trends in top income shares in the United States: reconciling estimates from March CPS and IRS tax return data". In: *Review of Economics and Statistics* 94.2, pp. 371–388.

Chawla, N. V., K. W. Bowyer, L. O. Hall, et al. (2002). "SMOTE: synthetic minority over-sampling technique". In: *Journal of artificial intelligence research* 16, pp. 321–357.

Edsal, T. B (2017). *How did the Democrats become favourites of the rich?* URL: https://www.nytimes.com/2015/10/07/opinion/how-did-the-democrats-become-favorites-of-the-rich.html.

FEC (2017). *Quick Answers to General Questions*. URL: http://www.fec.gov/ans/answers_general.shtml.

Galster, George, Roger Andersson, and Sako Musterd (2010). "Who is affected by neighbourhood income mix? Gender, age, family, employment and income differences". In: *Urban Studies* 47.14, pp. 2915–2944.

Grubesic, T. H. (2008). "Zip codes and spatial analysis: Problems and prospects". In: *Socio-Economic Planning Sciences* 42.2, pp. 129–149.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman (2009). "Unsupervised learning". In: *The elements of statistical learning*. Springer, pp. 485–585.

IRS (2017). *SOI Tax Stats. Individual Income Tax Statistics. Zip Code Data*. URL: https://www.irs.gov/uac/soi-tax-stats-individual-income-tax-statistics-zip-code-data-soi.

James, Gareth, Daniela Witten, Trevor Hastie, et al. (2013). *An introduction to statistical learning*. Vol. 6. Springer.

Johnson, M., W. P. Shively, and R. M. Stein (2002). "Contextual data and the study of elections and voting behavior: connecting individuals to environments". In: *Electoral Studies* 21.2, pp. 219–233.

Kahle, David, Hadley Wickham, and Maintainer David Kahle (2016). *Package 'ggmap'*.

Krieg, G. J. (2017). *What is a Super PAC? A Short History*. URL: http://abcnews.go.com/Politics/OTUS/super-pac-short-history/story?id=16960267.

Miller, W. L. (1977). *Electoral dynamics in Britain since 1918*. Springer.