Blog 5 - French Presidential Election 2022 - Toulouse, France

Michael Ippolito

2022-11-26

Contents

Background

Since I'm spending the fall in Toulouse, France, I wanted to get a sense of what kind of city I'm living in. Tolouse is the fourth largest city in France (after Paris, Marseille, and Lyon), with a population of about 433,000 in 2022. It is home to Airbus and has a significant industrial and technical community, as well as many expatriates. As an urban center, it isn't surprising that it is overwhelminingly democratic in terms of politics. For this blog post, I wanted to investigate this quantitatively. The city of Toulouse maintains an excellent and extensive collection of data sets about a range of topics, many of which I found useful for this post:

https://data.toulouse-metropole.fr/explore/

To put some boundaries on scope, I focussed on trying to predict percentage of votes during the second round of the 2022 presidential elections for the Toulouse metropolitan area. In France, the presidential elections are held in two rounds, akin to the primary and general elections in the United States. For predictors, I used a subset of this data, along with fifty other data sets I downloaded from the same site:

Predictor
accelerators_incubators
agricultural_zones
art_galleries
bicycle_parking
bicycle_rentals
bowling_alleys
business_centers
cafe_concert_venues
canal_sites
carpool_stations
cemeteries
community_fitness_centers
cultural_centers
dog_parks

Predictor
dog_waste_bags
dumps
elementary_schools
flood_zones_1875
game_libraries
green_spaces
gymnasiums
institutes_of_cultural_instruction
lakes
libraries
markets
museums
park_and_rides
pedestrian_zones
playgrounds
pools
presidential_election_billboards
public_toilets
recharging_stations
regulation_offices
scooter_rentals
senior_restaurants
skate_parks
skating_rinks
social_centers
sociocultural_centers
speed_displays
stadiums
taxi_zones
tennis)courts
theaters
tramway_stations
vaccination_centers
wifi_zones
workers_rights_centers

Each data set includes geographic coordinates (latitude and longitude) that I used to calculate how far each entity is away from each polling station. Then I took the median distance of all the entites in each category to feed into a binary logistic regression model to predict the percentage of the vote each candidate would get.

EDA

The data set I used as the response includes results from all polling places in Toulouse and consists of the following fields:

Field	Description
Sequence	sequence number
Type	election type (PR=présidential)
Année	election year (2022)

Field	Description
Tour	election round (second round)
Département	department (31 = Haute-Garonne)
Commune	commune (555 = Toulouse)
Code canton	voting district code (15 - 25)
Code circonscription	constituency code (varies per voting district)
Numéro du bureau	polling place number (varies per voting district)
Indicatif	informational code (always I)
Nombre d'inscrits	number of participants
Nombre d'abstentions	number of abstentions
Nombre de votants	number of voters = inscrits - abstentions
Nombre de votants d'après les feuilles d'émargement	number of voters according to the attendance sheets (should be same as nombre de votants)
Nombre de bulletins blancs	number of blank ballots
Nombre de bulletins nuls	number of invalid ballots
Nombre d'exprimés	number of valid ballots (votants - bulletins blancs - bulletins nuls)
Nombre de candidats	number of candidates
Sigle du candidat	candidate's name
Nombre de voix du candidat	number of votes for the candidate
Geo Shape	array of geographic coordinates outlining the area of polling place
NOM	name of the polling place
ADRESSE	address of the polling place
geo_point_2d	geographic coordinates of the center of the polling place

The data required some cleaning, including spreading the data from long to wide format and separating the latitude and longtidue coordinates into different fields. The following is a summary of the fields in the response data frame after cleaning.

```
##     district      constituency   polling_place_num   participants
##  Min.   :15.00   Min.   :1.000   Length:265         Min.   :  24.0
##  1st Qu.:17.00   1st Qu.:2.000   Class :character   1st Qu.: 881.0
##  Median :19.00   Median :3.000   Mode  :character   Median : 995.0
##  Mean   :19.35   Mean   :3.611                      Mean   : 976.6
##  3rd Qu.:22.00   3rd Qu.:4.000                      3rd Qu.:1106.0
##  Max.   :25.00   Max.   :9.000                      Max.   :1921.0
##   abstentions       voters1          ballots           blank
##  Min.   : 21.0   Min.   :   3.0   Min.   :   3.0   Min.   :  0.00
##  1st Qu.:231.0   1st Qu.: 619.0   1st Qu.: 619.0   1st Qu.: 41.00
##  Median :282.0   Median : 717.0   Median : 717.0   Median : 52.00
##  Mean   :288.1   Mean   : 688.6   Mean   : 688.5   Mean   : 50.69
##  3rd Qu.:330.0   3rd Qu.: 797.0   3rd Qu.: 797.0   3rd Qu.: 61.00
##  Max.   :613.0   Max.   :1328.0   Max.   :1328.0   Max.   :104.00
##     invalid          valid        polling_place      polling_addr
##  Min.   : 0.00   Min.   :   3.0   Length:265         Length:265
##  1st Qu.:11.00   1st Qu.: 556.0   Class :character   Class :character
##  Median :14.00   Median : 646.0   Mode  :character   Mode  :character
##  Mean   :15.13   Mean   : 622.8
##  3rd Qu.:19.00   3rd Qu.: 730.0
##  Max.   :37.00   Max.   :1190.0
##    geopoint            geo.lat         geo.lon          Le_Pen
##  Length:265         Min.   :43.54   Min.   :1.367   Min.   :  1.0
##  Class :character   1st Qu.:43.58   1st Qu.:1.422   1st Qu.:106.0
```

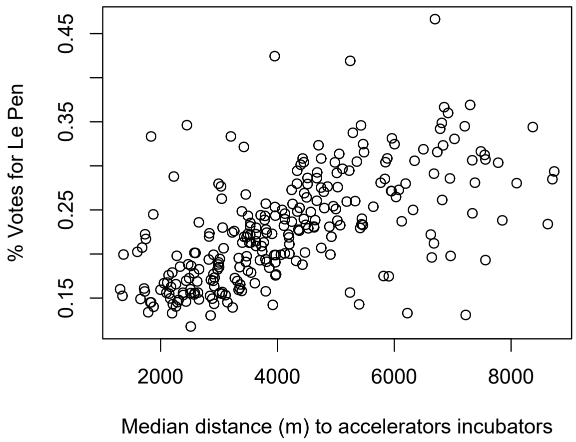
```
##  Mode  :character   Median :43.60   Median :1.444   Median :134.0
##                     Mean   :43.60   Mean   :1.440   Mean   :140.2
##                     3rd Qu.:43.61   3rd Qu.:1.461   3rd Qu.:169.0
##                     Max.   :43.66   Max.   :1.502   Max.   :302.0
##      Macron
##  Min.   :  2.0
##  1st Qu.:416.0
##  Median :497.0
##  Mean   :482.5
##  3rd Qu.:565.0
##  Max.   :890.0
```

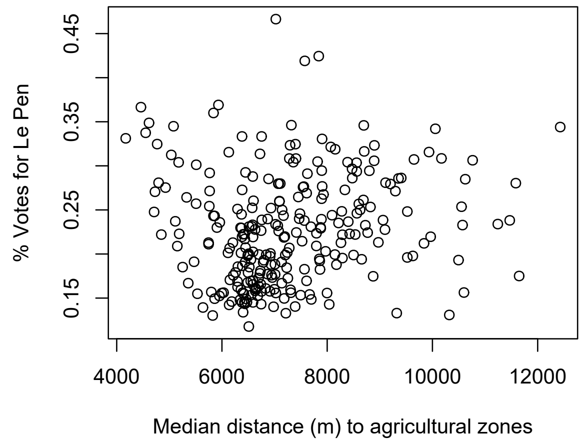
The predictor fields also required some cleaning, including standardizing the name of the field containing geographic coordinates and separating it out into two different columns.

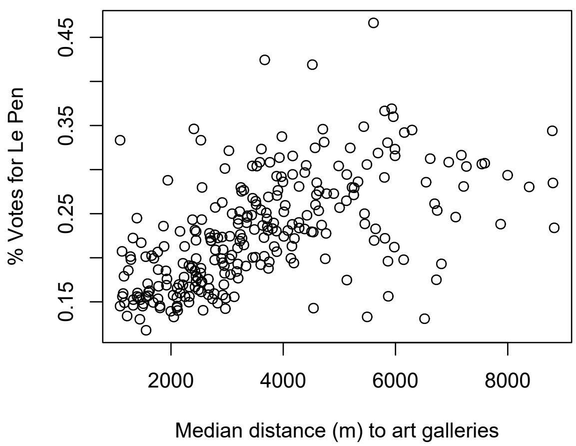
After cleaning the data, I calculated distances from each polling location to the various municipal entities we're using as predictors. We'll use the haversine formula to compute distance, which uses latitude and longitude, taking into account the curvature of the earth.

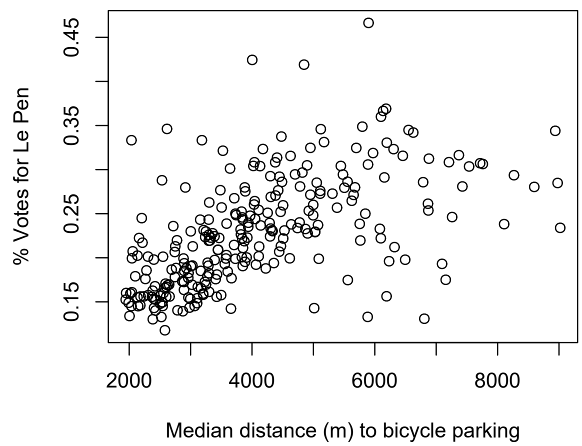
The following plots show the response as a function of predictors. As shown, as the distance increases to entities closely associated with metropolitan activities, the more votes for Le Pen were recorded.

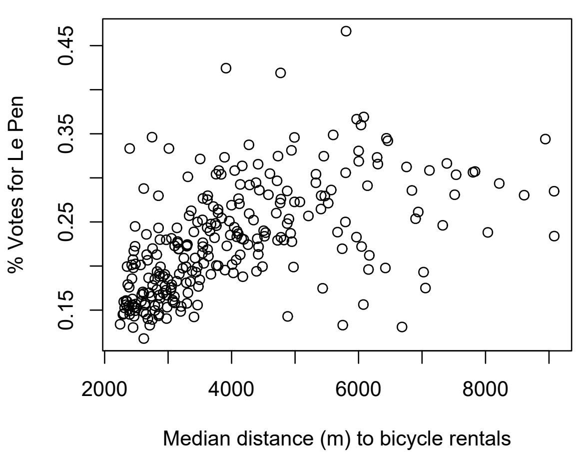
```
# Trim off columns not used in modeling
dfmodel <- df2 %>%
    select (-polling_place_num, -district, -constituency, -polling_place, -polling_addr, -geopoint, -ge

# EDA
for (i in seq_along(pred_files)) {
    newxlab <- gsub('_', ' ', pred_files[i])
    #newxlab <- paste0('Distance (m) to nearest ', newxlab)
    newxlab <- paste0('Median distance (m) to ', newxlab)
    if (i %% 2 == 1) {
        par(mfrow=c(1, 2))
    }
    #plot(dfplots$votes ~ dfplots[, pred_files[i]], col=as.factor(dfplots$candidate),
    #     pch=as.numeric(as.factor(dfplots$candidate)), xlab=newxlab, ylab='Votes')
    plot(dfmodel$Le_Pen / (dfmodel$Le_Pen + dfmodel$Macron) ~ dfmodel[, pred_files[i]],
         xlab=newxlab, ylab='% Votes for Le Pen')
    ax <- par('usr')
}
```

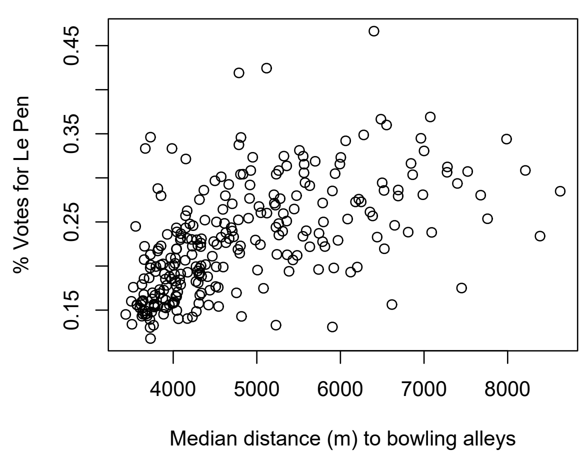

% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
4000
6000
8000
Median distance (m) to accelerators incubators

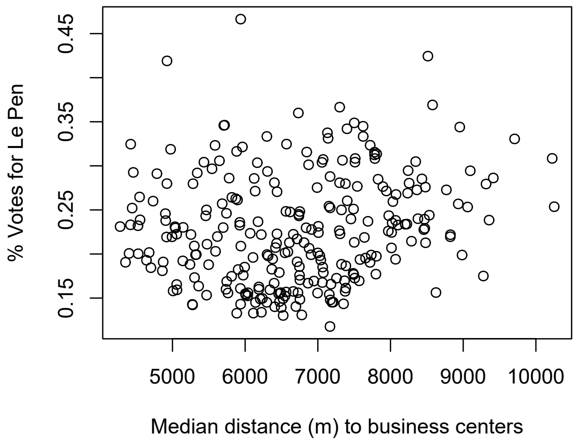

% Votes for Le Pen
0.15
0.25
0.35
0.45
4000
6000
8000
10000
12000
Median distance (m) to agricultural zones

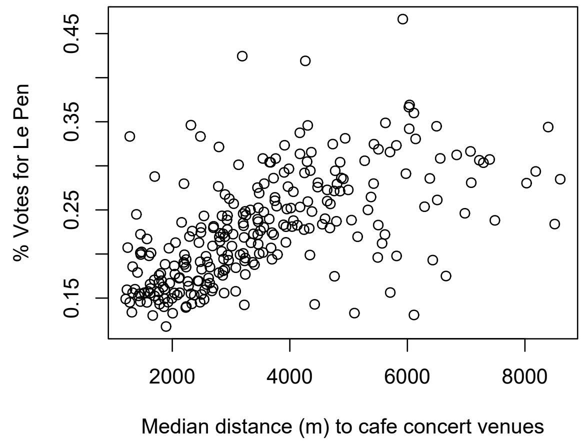

% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
4000
6000
8000
Median distance (m) to art galleries

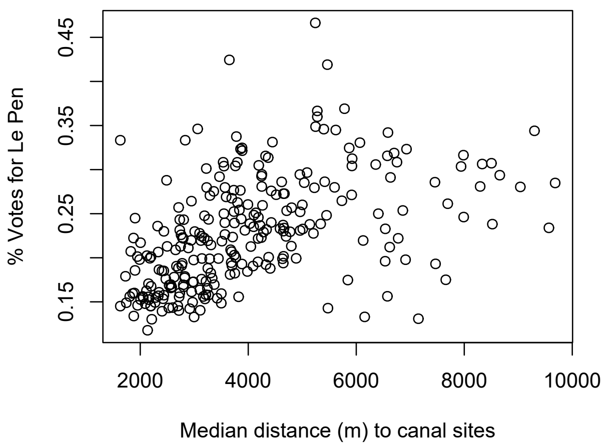

% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
4000
6000
8000
Median distance (m) to bicycle parking

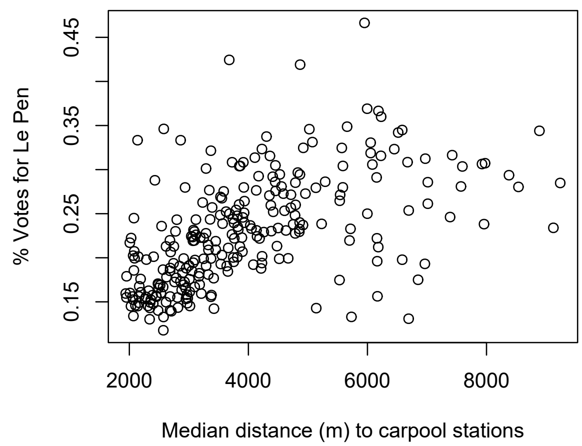

% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
4000
6000
8000
Median distance (m) to bicycle rentals

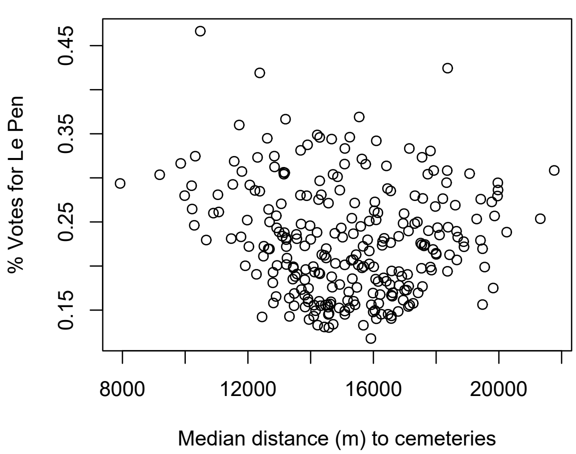

% Votes for Le Pen
0.15
0.25
0.35
0.45
4000
5000
6000
7000
8000
Median distance (m) to bowling alleys

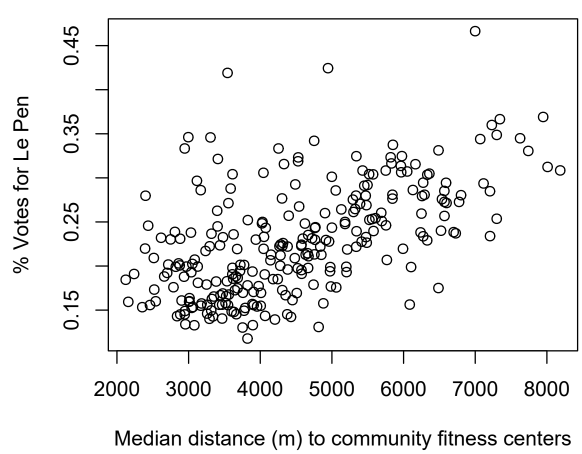

% Votes for Le Pen
0.15
0.25
0.35
0.45
5000
6000
7000
8000
9000
10000
Median distance (m) to business centers

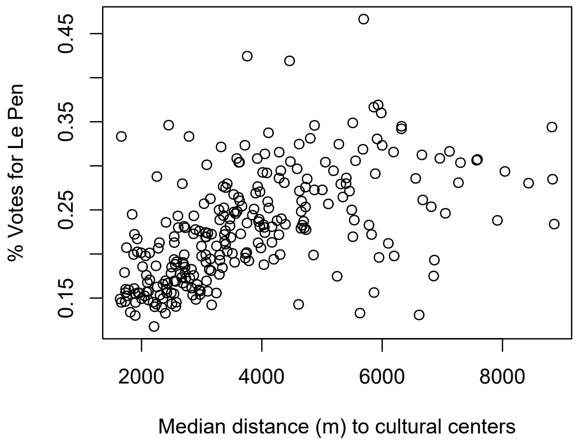

% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
4000
6000
8000
Median distance (m) to cafe concert venues

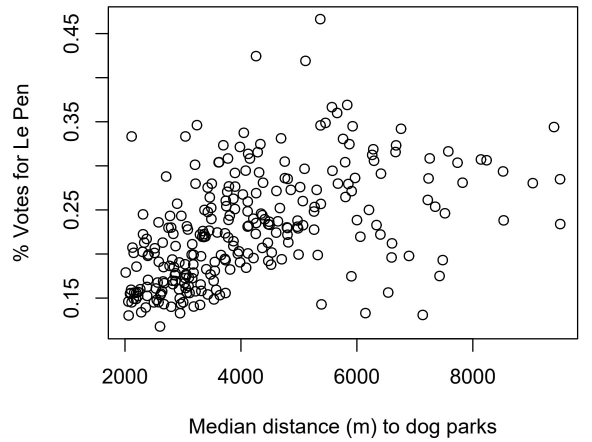

% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
4000
6000
8000
10000
Median distance (m) to canal sites

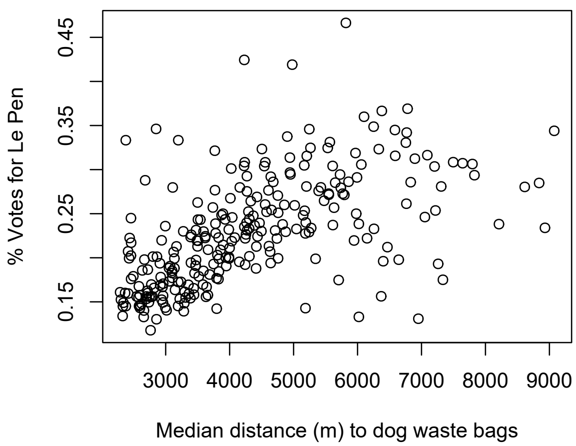

% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
4000
6000
8000
Median distance (m) to carpool stations

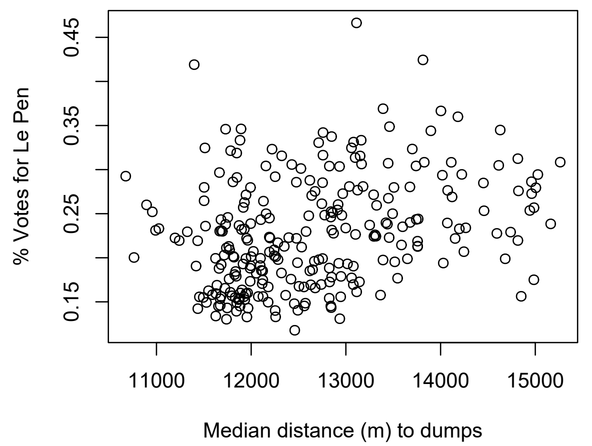

% Votes for Le Pen
0.15
0.25
0.35
0.45
8000
12000
16000
20000
Median distance (m) to cemeteries

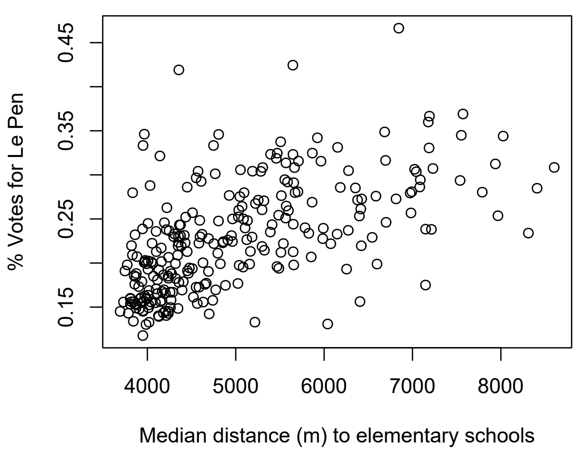

% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
3000
4000
5000
6000
7000
8000
Median distance (m) to community fitness centers

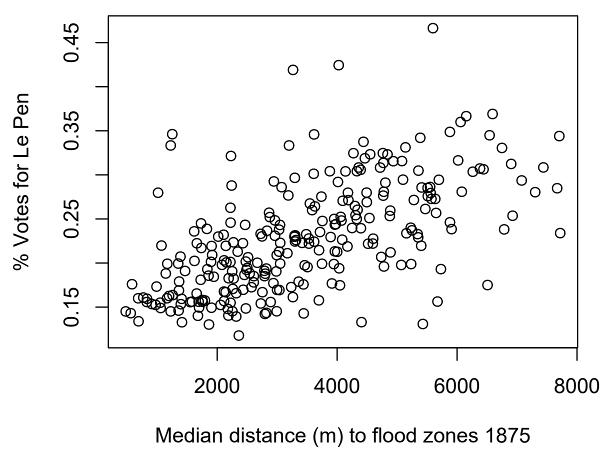

% Votes for Le Pen
0.15 0.25 0.35 0.45
2000 4000 6000 8000
Median distance (m) to cultural centers

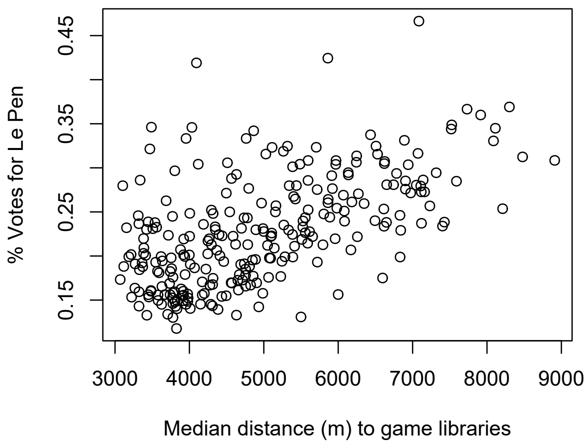

% Votes for Le Pen
0.15 0.25 0.35 0.45
2000 4000 6000 8000
Median distance (m) to dog parks

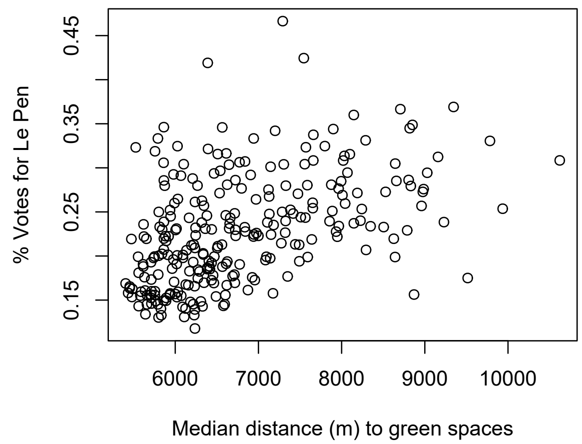

% Votes for Le Pen
0.15 0.25 0.35 0.45
3000 4000 5000 6000 7000 8000 9000
Median distance (m) to dog waste bags

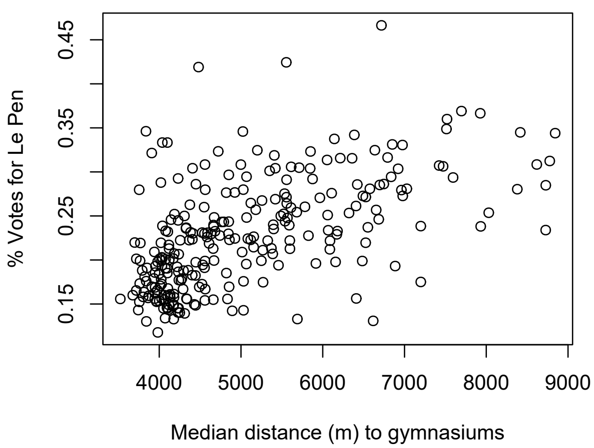

% Votes for Le Pen
0.15 0.25 0.35 0.45
11000 12000 13000 14000 15000
Median distance (m) to dumps

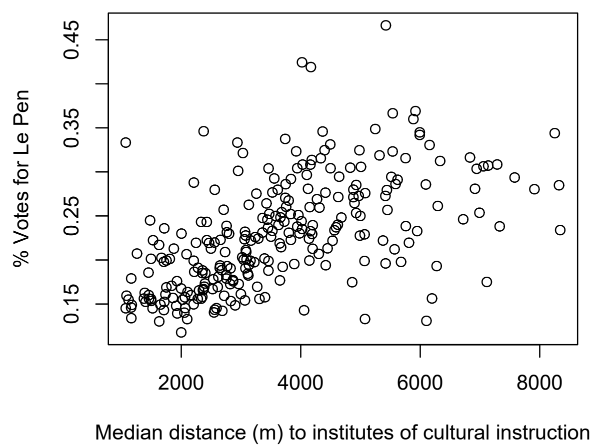

% Votes for Le Pen
0.15 0.25 0.35 0.45
4000 5000 6000 7000 8000
Median distance (m) to elementary schools

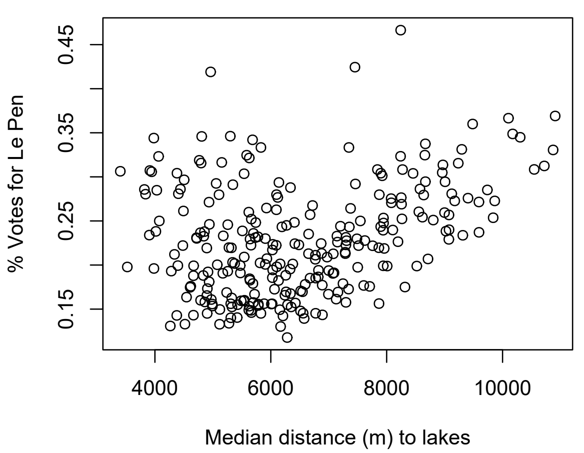

% Votes for Le Pen
0.15 0.25 0.35 0.45
2000 4000 6000 8000
Median distance (m) to flood zones 1875

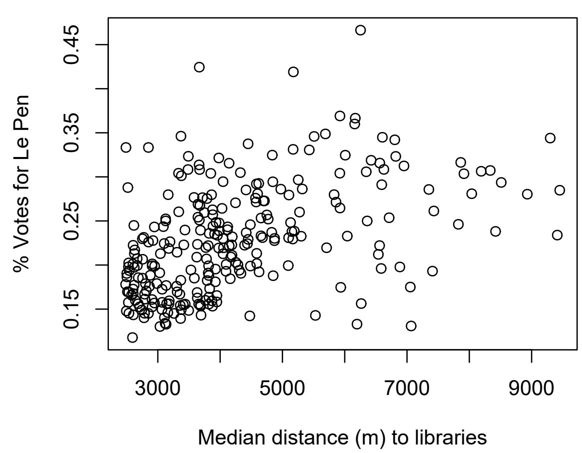

% Votes for Le Pen
0.15
0.25
0.35
0.45
3000
4000
5000
6000
7000
8000
9000
Median distance (m) to game libraries

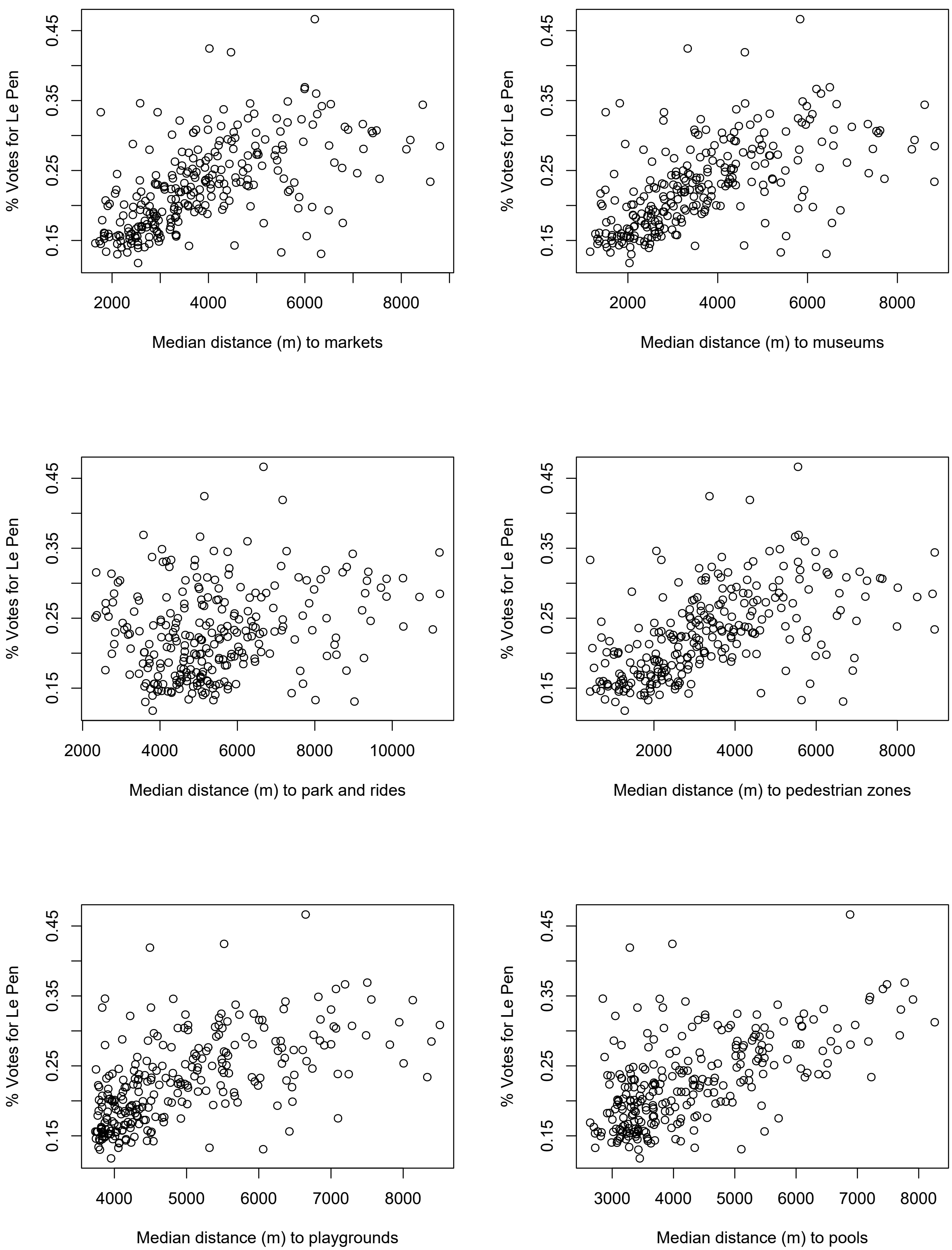

% Votes for Le Pen
0.15
0.25
0.35
0.45
6000
7000
8000
9000
10000
Median distance (m) to green spaces

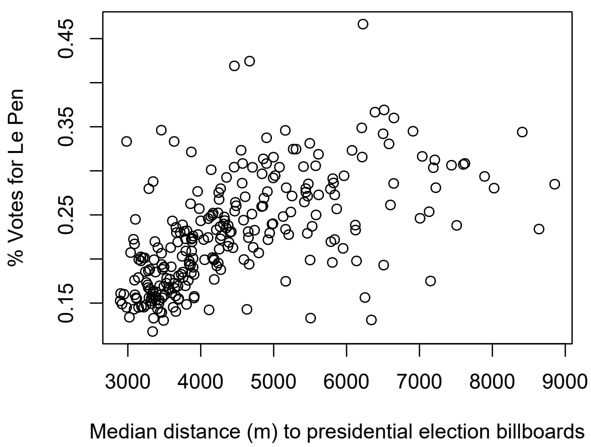

% Votes for Le Pen
0.15
0.25
0.35
0.45
4000
5000
6000
7000
8000
9000
Median distance (m) to gymnasiums


% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
4000
6000
8000
Median distance (m) to institutes of cultural instruction

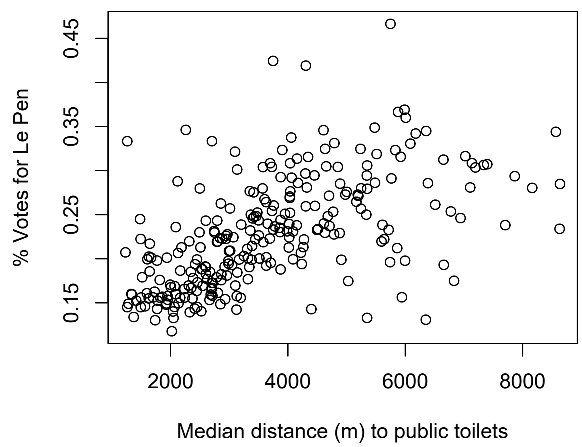

% Votes for Le Pen
0.15
0.25
0.35
0.45
4000
6000
8000
10000
Median distance (m) to lakes


% Votes for Le Pen
0.15
0.25
0.35
0.45
3000
5000
7000
9000
Median distance (m) to libraries

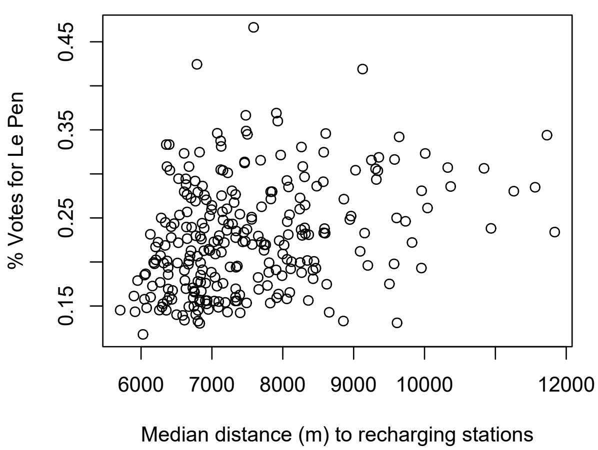

% Votes for Le Pen
Median distance (m) to markets
Median distance (m) to museums
Median distance (m) to park and rides
Median distance (m) to pedestrian zones
Median distance (m) to playgrounds
Median distance (m) to pools



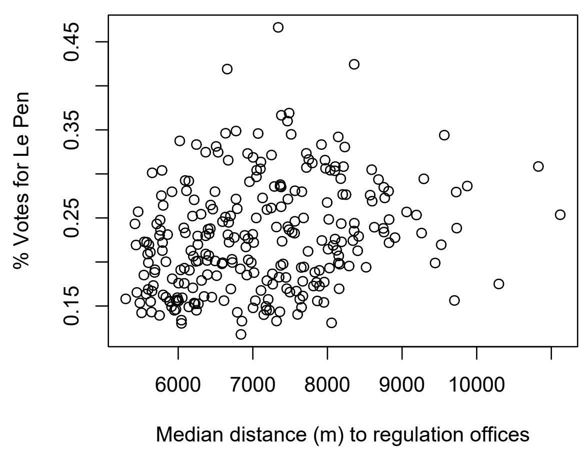
% Votes for Le Pen
0.15
0.25
0.35
0.45
3000
4000
5000
6000
7000
8000
9000
Median distance (m) to presidential election billboards



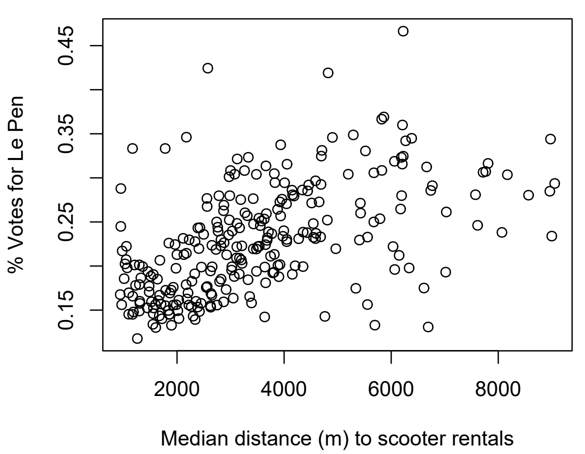
% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
4000
6000
8000
Median distance (m) to public toilets



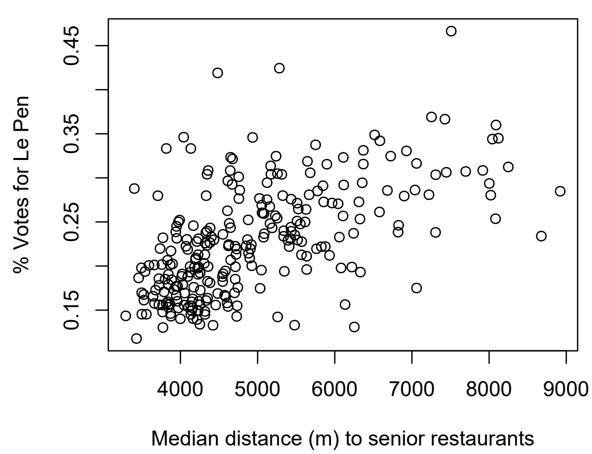
% Votes for Le Pen
0.15
0.25
0.35
0.45
6000
7000
8000
9000
10000
12000
Median distance (m) to recharging stations



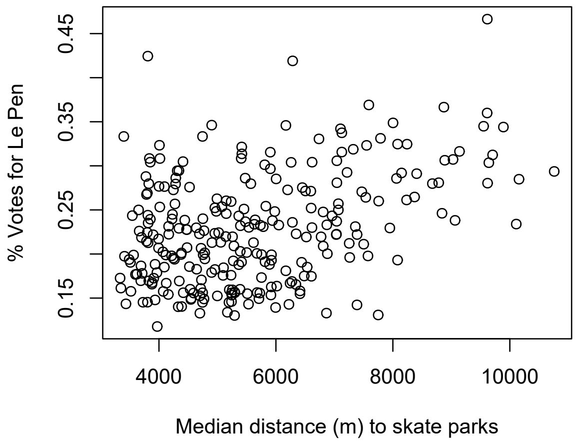
% Votes for Le Pen
0.15
0.25
0.35
0.45
6000
7000
8000
9000
10000
Median distance (m) to regulation offices

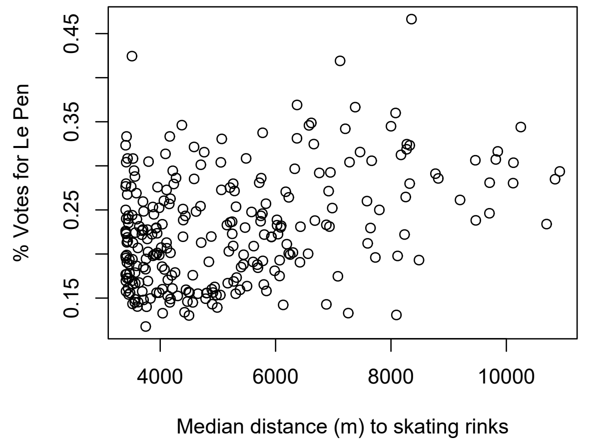


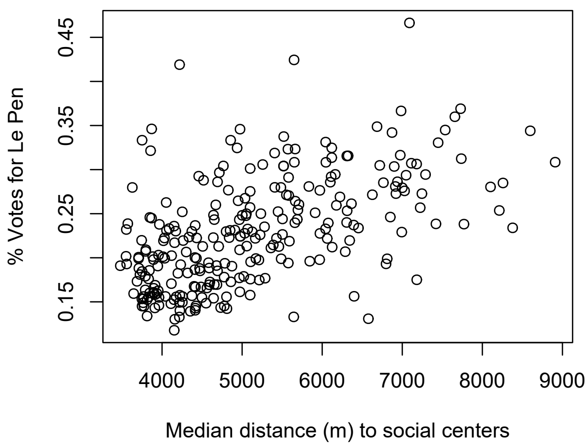
% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
4000
6000
8000
Median distance (m) to scooter rentals

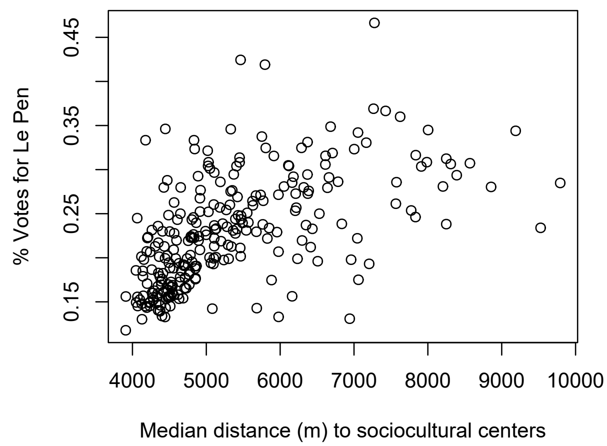


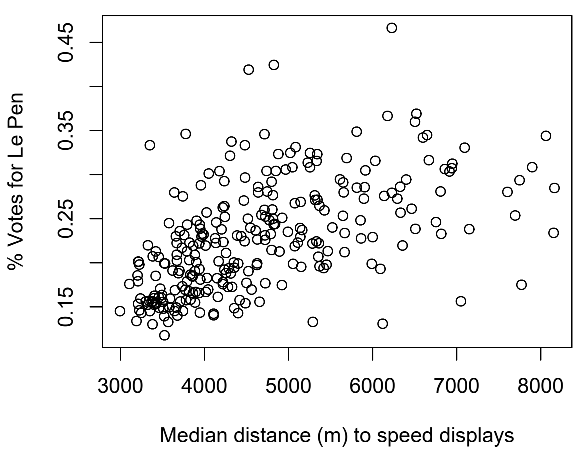
% Votes for Le Pen
0.15
0.25
0.35
0.45
4000
5000
6000
7000
8000
9000
Median distance (m) to senior restaurants

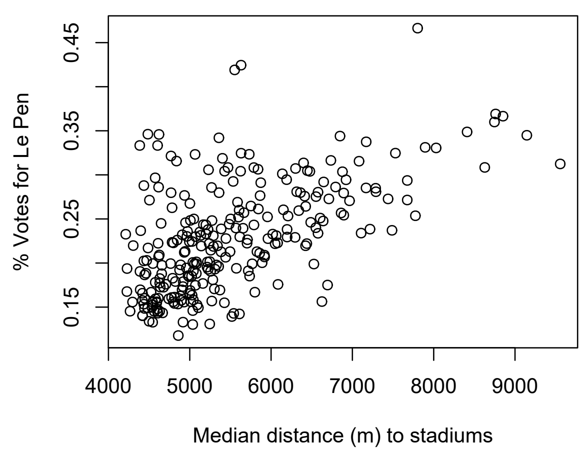

% Votes for Le Pen
0.15
0.25
0.35
0.45
4000
6000
8000
10000
Median distance (m) to skate parks

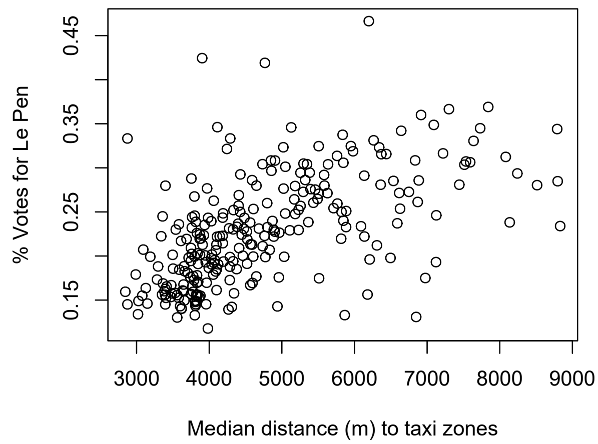

% Votes for Le Pen
0.15
0.25
0.35
0.45
4000
6000
8000
10000
Median distance (m) to skating rinks

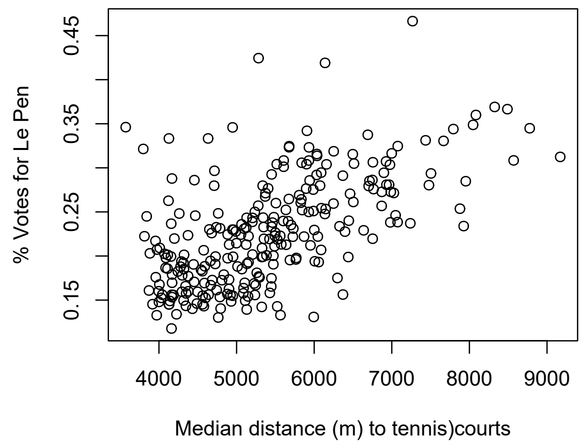

% Votes for Le Pen
0.15
0.25
0.35
0.45
4000
5000
6000
7000
8000
9000
Median distance (m) to social centers

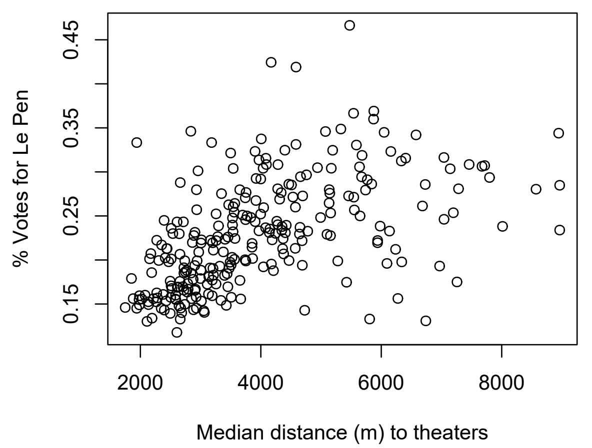

% Votes for Le Pen
0.15
0.25
0.35
0.45
4000
5000
6000
7000
8000
9000
10000
Median distance (m) to sociocultural centers

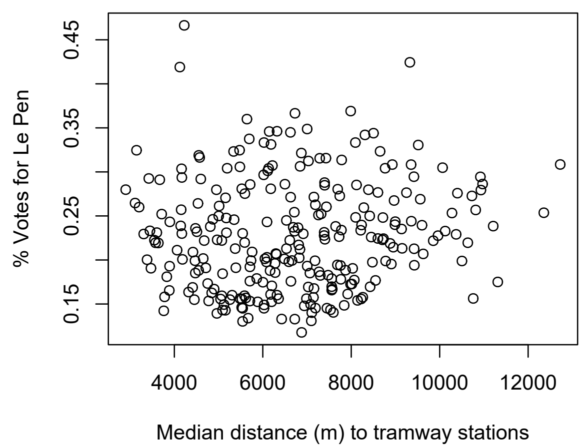

% Votes for Le Pen
0.15
0.25
0.35
0.45
3000
4000
5000
6000
7000
8000
Median distance (m) to speed displays

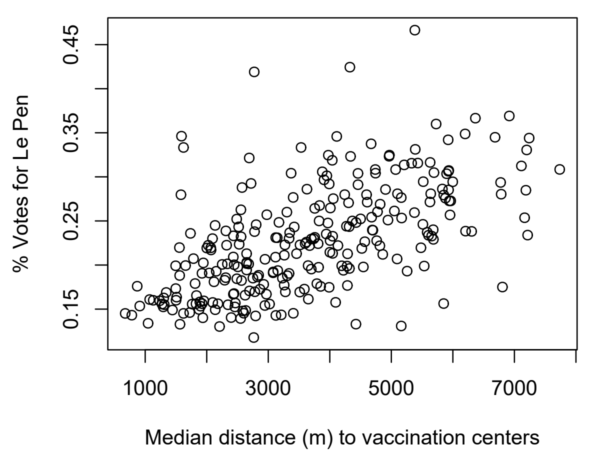

% Votes for Le Pen
0.15
0.25
0.35
0.45
4000
5000
6000
7000
8000
9000
Median distance (m) to stadiums

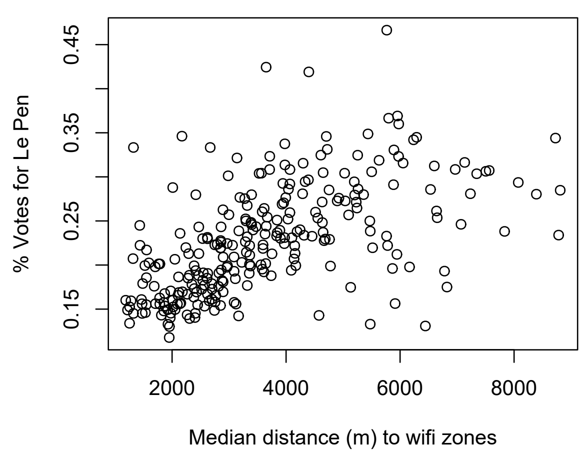

% Votes for Le Pen
0.15
0.25
0.35
0.45
3000
4000
5000
6000
7000
8000
9000
Median distance (m) to taxi zones


% Votes for Le Pen
0.15
0.25
0.35
0.45
4000
5000
6000
7000
8000
9000
Median distance (m) to tennis)courts


% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
4000
6000
8000
Median distance (m) to theaters


% Votes for Le Pen
0.15
0.25
0.35
0.45
4000
6000
8000
10000
12000
Median distance (m) to tramway stations

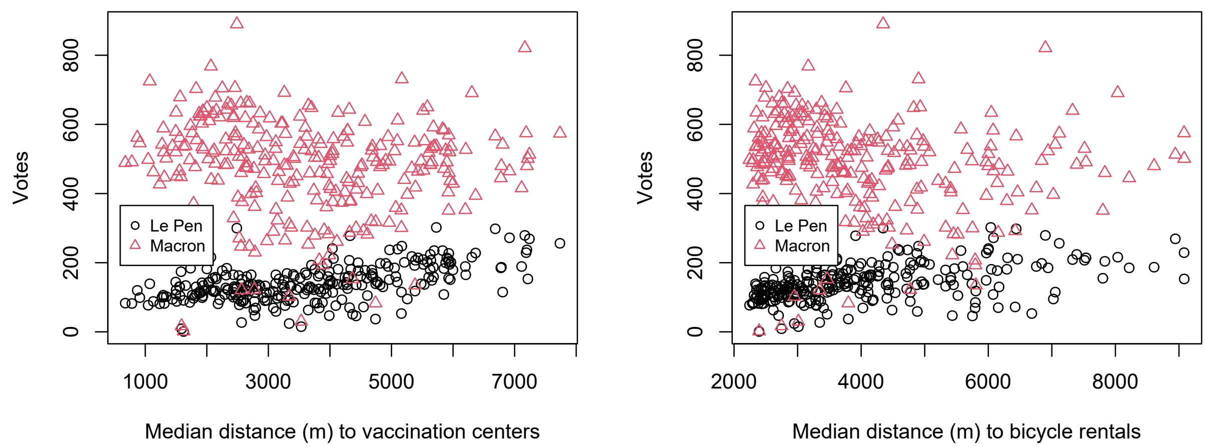

% Votes for Le Pen
0.15
0.25
0.35
0.45
1000
3000
5000
7000
Median distance (m) to vaccination centers


% Votes for Le Pen
0.15
0.25
0.35
0.45
2000
4000
6000
8000
Median distance (m) to wifi zones

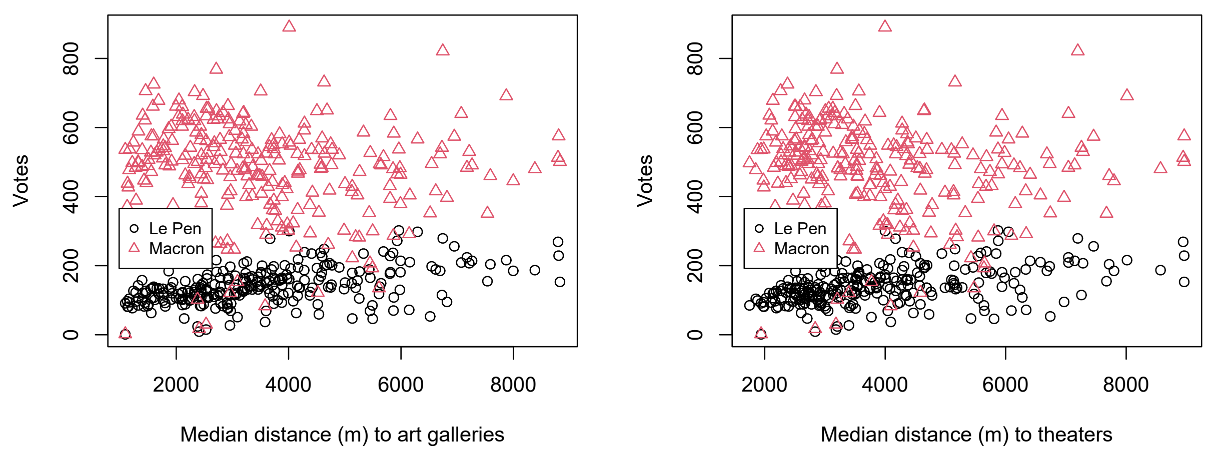
The following plots illustrate which predictors are the most closely associated with each candidate.

```
# Gather
dfplots <- df2 %>%
    select (-district, -constituency, -polling_place, -polling_addr, -geopoint, -geo.lat, -geo.lon) %>%
    gather(c('Le_Pen', 'Macron'), key='candidate', value='votes')

# Predictors associated with Le Pen
ct <- 0
for (i in c(47, 5, 3, 45)) {
    ct <- ct + 1
    newxlab <- gsub('_', ' ', pred_files[i])
    #newxlab <- paste0('Distance (m) to nearest ', newxlab)
    newxlab <- paste0('Median distance (m) to ', newxlab)
    if (ct %% 2 == 1) {
        par(mfrow=c(1, 2))
    }
    plot(dfplots$votes ~ dfplots[, pred_files[i]], col=as.factor(dfplots$candidate),
         pch=as.numeric(as.factor(dfplots$candidate)), xlab=newxlab, ylab='Votes')
    ax <- par('usr')
    legend(ax[1] + 200, ax[3] + 400, legend=c('Le Pen', 'Macron'), col=c(1, 2), pch=c(1, 2), cex=0.8)
    mtext('Predictors associated with Le Pen', side=3, line=-1, outer=T)
}
```

Predictors associated with Le Pen

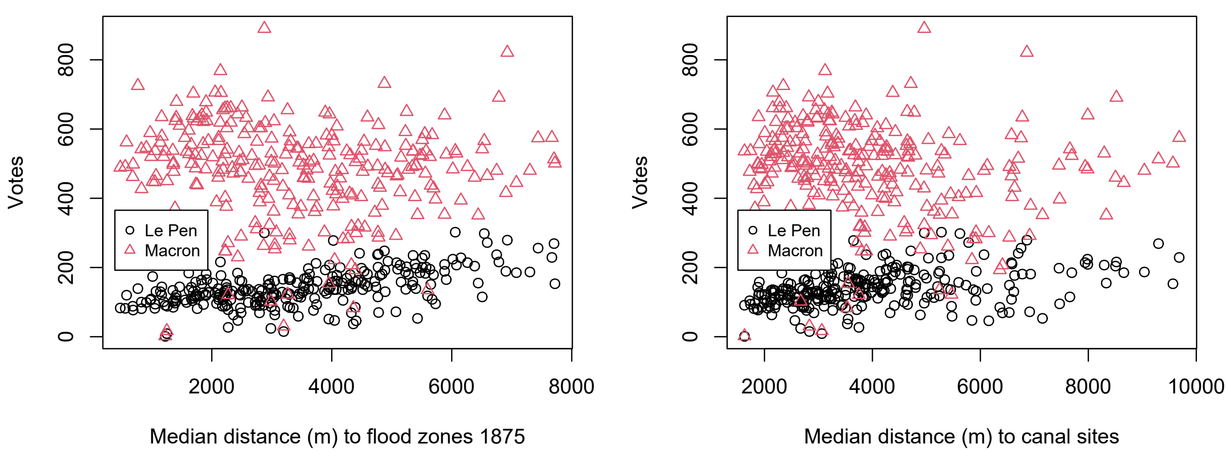


Predictors associated with Le Pen

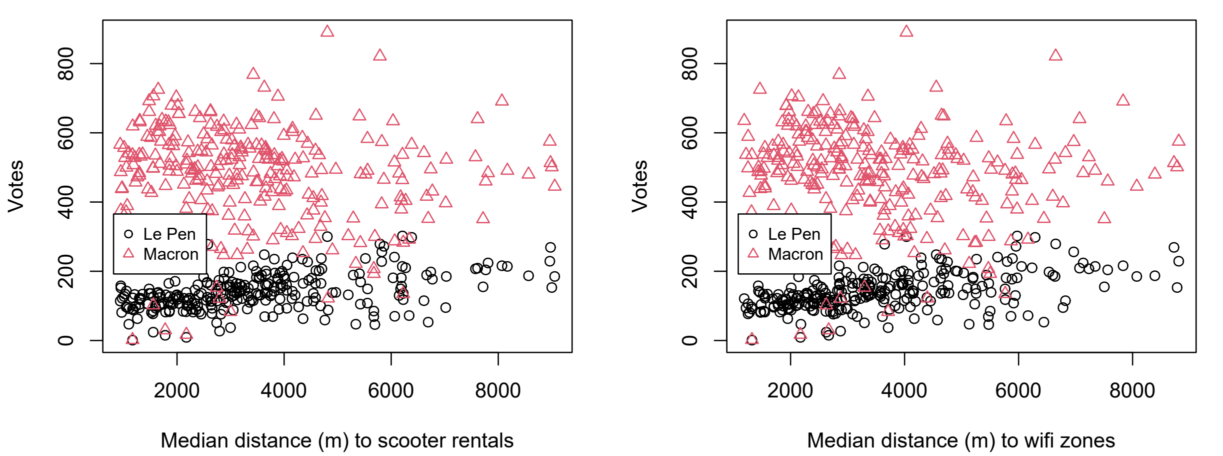



```
# Predictors associated with Macron
ct <- 0
for (i in c(18, 9, 35, 48)) {
    ct <- ct + 1
    newxlab <- gsub('_', ' ', pred_files[i])
    #newxlab <- paste0('Distance (m) to nearest ', newxlab)
    newxlab <- paste0('Median distance (m) to ', newxlab)
    if (ct %% 2 == 1) {
        par(mfrow=c(1, 2))
    }
    plot(dfplots$votes ~ dfplots[, pred_files[i]], col=as.factor(dfplots$candidate),
         pch=as.numeric(as.factor(dfplots$candidate)), xlab=newxlab, ylab='Votes')
    ax <- par('usr')
    legend(ax[1] + 200, ax[3] + 400, legend=c('Le Pen', 'Macron'), col=c(1, 2), pch=c(1, 2), cex=0.8)
    mtext('Predictors associated with Macron', side=3, line=-1, outer=T)
}
```

Predictors associated with Macron



Predictors associated with Macron



Modeling

Since the response is binary (Le Pen vs Macron), I used a binary logistic regression model. Using backward elimination, I reduced the model to its most significant predictors.

The following shows a summary of the reduced model:

```r
# Reduced model
summary(bmod2)
```

```
##
## Call:
## glm(formula = cbind(Le_Pen, Macron) ~ participants + abstentions +
##     ballots + blank + invalid + accelerators_incubators + agricultural_zones +
##     art_galleries + bicycle_rentals + business_centers + cafe_concert_venues +
##     canal_sites + carpool_stations + cemeteries + community_fitness_centers +
##     dog_parks + dumps + flood_zones_1875 + game_libraries + gymnasiums +
```

```
##     institutes_of_cultural_instruction + lakes + park_and_rides +
##     pools + public_toilets + recharging_stations + scooter_rentals +
##     skate_parks + skating_rinks + social_centers + sociocultural_centers +
##     taxi_zones + theaters + vaccination_centers + wifi_zones,
##     family = binomial(), data = dfmodel)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -6.2014  -1.3403  -0.1514   1.1146   8.0248
##
## Coefficients:
##                                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)                         6.967e-01  4.143e-01   1.681 0.092671 .
## participants                        2.730e-02  1.323e-02   2.063 0.039077 *
## abstentions                        -2.684e-02  1.323e-02  -2.029 0.042493 *
## ballots                            -2.801e-02  1.324e-02  -2.116 0.034348 *
## blank                               3.882e-03  7.046e-04   5.510 3.59e-08 ***
## invalid                             2.758e-03  1.383e-03   1.994 0.046098 *
## accelerators_incubators             1.803e-04  3.411e-05   5.285 1.26e-07 ***
## agricultural_zones                  8.544e-05  2.603e-05   3.282 0.001029 **
## art_galleries                       3.551e-04  9.748e-05   3.643 0.000270 ***
## bicycle_rentals                     3.684e-04  1.222e-04   3.015 0.002566 **
## business_centers                   -8.632e-05  2.636e-05  -3.275 0.001058 **
## cafe_concert_venues                 2.242e-04  8.690e-05   2.580 0.009880 **
## canal_sites                        -2.960e-04  5.710e-05  -5.184 2.17e-07 ***
## carpool_stations                   -1.695e-04  7.632e-05  -2.221 0.026376 *
## cemeteries                         -1.425e-04  2.377e-05  -5.995 2.04e-09 ***
## community_fitness_centers           2.244e-04  4.574e-05   4.905 9.32e-07 ***
## dog_parks                          -1.906e-04  7.870e-05  -2.422 0.015436 *
## dumps                              -1.410e-04  2.425e-05  -5.812 6.17e-09 ***
## flood_zones_1875                   -4.330e-04  8.603e-05  -5.033 4.84e-07 ***
## game_libraries                      1.126e-04  4.181e-05   2.692 0.007093 **
## gymnasiums                          9.658e-05  4.177e-05   2.312 0.020766 *
## institutes_of_cultural_instruction  1.840e-04  7.745e-05   2.376 0.017505 *
## lakes                               4.636e-05  2.072e-05   2.238 0.025243 *
## park_and_rides                      4.571e-05  2.618e-05   1.746 0.080760 .
## pools                              -9.239e-05  3.574e-05  -2.585 0.009741 **
## public_toilets                     -2.027e-04  1.317e-04  -1.539 0.123715
## recharging_stations                 1.516e-04  2.630e-05   5.766 8.10e-09 ***
## scooter_rentals                    -2.669e-04  6.866e-05  -3.888 0.000101 ***
## skate_parks                        -6.308e-05  3.570e-05  -1.767 0.077191 .
## skating_rinks                      -1.838e-04  3.815e-05  -4.818 1.45e-06 ***
## social_centers                     -1.843e-04  4.453e-05  -4.139 3.48e-05 ***
## sociocultural_centers               1.928e-04  3.935e-05   4.900 9.61e-07 ***
## taxi_zones                         -1.562e-04  3.668e-05  -4.258 2.06e-05 ***
## theaters                            2.392e-04  7.920e-05   3.020 0.002531 **
## vaccination_centers                 3.785e-04  5.513e-05   6.866 6.59e-12 ***
## wifi_zones                         -2.129e-04  1.347e-04  -1.580 0.114002
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3321.1  on 264  degrees of freedom
```

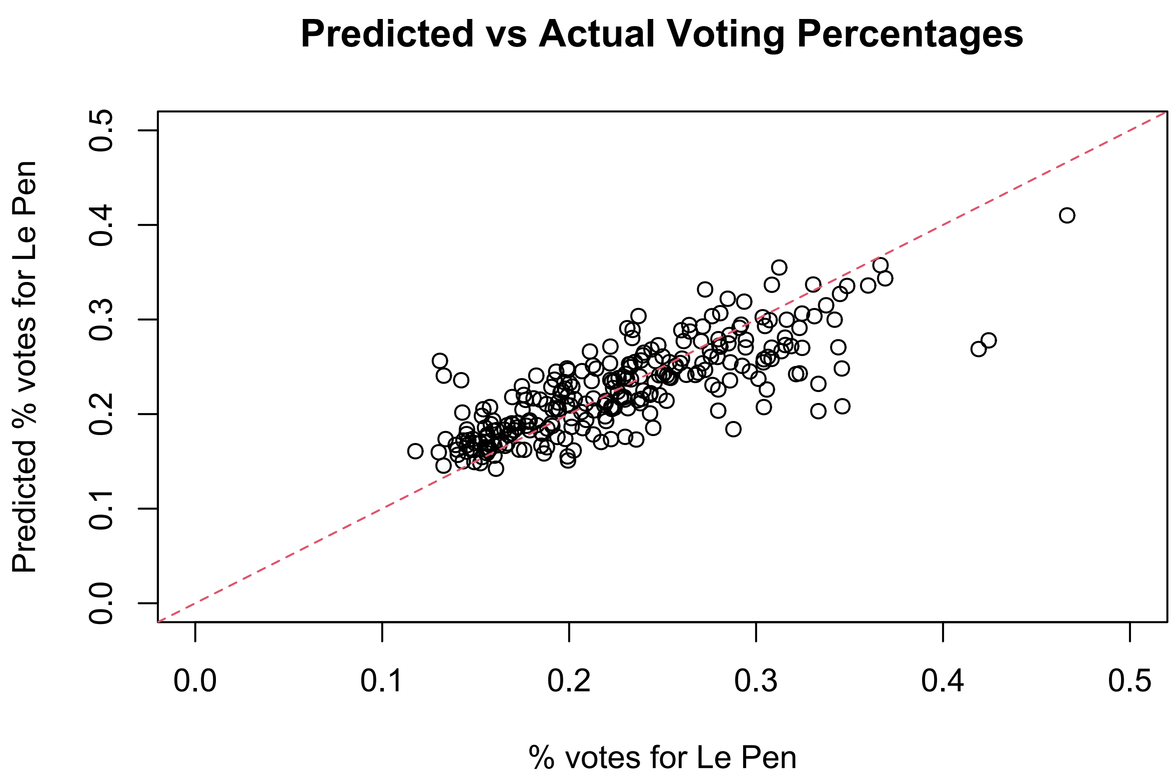
```
## Residual deviance: 1033.4  on 229  degrees of freedom
## AIC: 2808.1
##
## Number of Fisher Scoring iterations: 4
```

As shown, the residual deviance was much less than the null deviance on 229 degrees of freedom, indicating a good fit.

Conclusion

Using the parameters estimated by the model, predictions were made of the candidates' percentages of the vote. As shown below, the model performed well, with an R-squared value of 0.63 comparing the predicted versus the actual percentage. It can be concluded that calculating the median distance between each polling place and various civic features is a fairly good means of predicting election results. An additional model might be created for the first round of the election that included more than just the two finalists.

```
# Calculate predicted probabilities
dfmodel$p <- dfmodel$Le_Pen / (dfmodel$Le_Pen + dfmodel$Macron)
dfmodel$pred_p <- ilogit(predict(bmod2, newdata=dfmodel))
dfmodel$winner <- ifelse(dfmodel$Le_Pen == dfmodel$Macron, 'Draw', ifelse(dfmodel$Le_Pen > dfmodel$Macr
dfmodel$pred_winner <- ifelse(dfmodel$pred_p == 0.5, 'Draw', ifelse(dfmodel$pred_p > 0.5, 'Le_Pen', 'Ma

# Plot predicted probabilities vs actual
plot(pred_p ~ p, data=dfmodel, xlab='% votes for Le Pen', ylab='Predicted % votes for Le Pen',
     main='Predicted vs Actual Voting Percentages', xlim=c(0, 0.5), ylim=c(0, 0.5))
abline(0, 1, col=2, lt=2)
```



```r
# Evaluate model
lmod <- lm(pred_p ~ p, data=dfmodel)
summary(lmod)
```

```
##
## Call:
## lm(formula = pred_p ~ p, data = dfmodel)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.091072 -0.017126 -0.001688  0.019329  0.089835
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.085643   0.006944   12.33   <2e-16 ***
## p           0.617356   0.029401   21.00   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0298 on 263 degrees of freedom
## Multiple R-squared:  0.6264, Adjusted R-squared:  0.6249
## F-statistic: 440.9 on 1 and 263 DF,  p-value: < 2.2e-16
```