

Online Appendix for: “What’s in a Name? A Method for Extracting Information about Ethnicity from Names.”

December 6, 2014

This appendix contains supplementary materials for the article, “What’s in a Name? A Method for Extracting Information about Ethnicity from Names.” I detail training data preparation, present Monte Carlo simulations, and verification exercises for US and Kenyan data.

1. DATA PREPARATION

This section describes the preparation of training data used in the proposed approach and the classify-and-aggregate approaches. I discuss how mean absolute error (MAE) for the classify-and-aggregate approach depends on $P(\mathbf{Group})$ in the training data, and why this is not a problem with the proposed approach. Two different ways in which the conditional $P(Group|Name)$ can be calculated are examined, each of which corresponds to different assumptions about $P(\mathbf{Group})$ in the training data. This is important because, as shown in the Monte Carlo simulations and US verification exercise, these assumptions affect the performance of the classify-and-aggregate approach.

The sizes of the underlying samples from each group affect classification performance. This is problematic in circumstances where there is little underlying information about the group populations to be estimated. The example data matrix below is an example of what training data on names might look like. The n rows each represent a unique name; the J columns represent the J identity groups. Each cell reports the number of times each name occurs in each group, with name i in group j occurring f_{ij} times. $P(Group|Name)$ can be calculated in two different ways. First,

$P(\text{Group } j | \text{Name } i)$ can be calculated as $P(\text{Group } j | \text{Name } i) = \frac{f_{ij}}{\sum_j f_{ij}}$.

This approach assumes that the absolute sizes of each group in the training data (e.g., the sum of each column j) contain important information regarding classification, beneficial when the target population of interest resembles the training population.¹ In this case, f_{i1} may be larger than f_{i2} , and thus $P(\text{Group } 1 | \text{Name } i) > P(\text{Group } 2 | \text{Name } i)$, which makes sense if the relative sizes of the groups in the target and training populations are similar.

	Group = 1	Group = 2	⋯	Group = J
Name 1	f_{11}	f_{12}	⋯	f_{1J}
Name 2	f_{21}	f_{22}	⋯	f_{2J}
⋮	⋮	⋮	⋮	⋮
Name N	f_{n1}	f_{n2}	⋯	f_{nJ}

However, this approach to calculating $P(\text{Group } j | \text{Name } i)$ can be problematic as large groups may wash out information about names common for small groups. Another approach would be to first divide each column by the column totals, and then divide each row element by the row totals to get $P'(\text{Group } j | \text{Name } i)$. This approach effectively assumes that the absolute sizes of each group are equal. For example, dividing each f_{ij} by $\sum_i f_{ij}$ would get p_{ij} . Then, $P'(\text{Group } j | \text{Name } i) = \frac{p_{ij}}{\sum_j p_{ij}}$ could be calculated. Following the example from the previous paragraph, p_{i1} may be smaller than p_{i2} even though $f_{i1} > f_{i2}$, and thus $P(\text{Group } 1 | \text{Name } i) < P(\text{Group } 2 | \text{Name } i)$.

This simple example serves to illustrate that assumptions made about the relative group sizes in the training data affects the calculation (and classification performance) of $P(\text{Group } j | \text{Name } i)$. Figure 1 presents a simulation that illustrates the question: does the composition of the training data affect the quality of the parameters estimated from the target data? To do so, I constructed a simulation demonstrating the dependence of the accuracy of estimates on the similarity of the training data and the target data when using classify-and-aggregate.

To construct the figure, I first set a diffuse Dirichlet distribution with $\alpha = \text{rep}(1/6, 6)$. Then, I took one draw from the Dirichlet to establish the parameters used to form the training data. This is the functional equivalent of jittering the “true parameters” (in this case, $1/6$ for each of six groups) in order to generate some distance between the simulated training data parameters and the parameters generating the target data. Next, I generated target

data using the “true parameters” represented by $\text{rep}(1/6, 6)$, and classified those names using the training data based on the Dirichlet draw. Finally, I recorded two kinds of mean absolute error (MAE): the mean differences between the training data parameters and the parameters that generated the target data, and the mean differences between the parameters estimated from the target data using the classify-and-aggregate approach and the “true parameters” from which the target data were simulated.

The horizontal axis displays the mean absolute error of the “true” parameters generating the target data and the target data parameters. The vertical axis represents the MAE of the “true” target data parameters and the estimates of those parameters. The figure shows that increasing the difference between the training and target data parameters corresponds to poorer estimates of the training data parameters. The next section provides Monte Carlo and verification empirical evidence that the performance of $P(\text{Group} \mid \text{Name})$ and $P'(\text{Group} \mid \text{Name})$ depends on their similarity to the parameters being estimated.

2. MONTE CARLO SIMULATIONS

This section describes the Monte Carlo simulations carried out to demonstrate the performance of the proposed estimator, as well as estimated from classify-and-aggregate approaches using $P(\text{Group} \mid \text{Name})$ and $P'(\text{Group} \mid \text{Name})$ (as defined above). For US Monte Carlo simulations and North Carolina verification exercises, training data from the United States census on the frequencies of name use in six groups is used.² These training data are then used to calculate the required conditional distributions for the classify-and-aggregate approaches. When $P(\text{Group} \mid \text{Name})$ is used for classification, it is referred to as “US population parameters,” as the conditional is computed directly from cell counts and row sums. Assuming parity across group populations – $P'(\text{Group} \mid \text{Name})$ – is referred to as “uniform population parameters.” For the Kenyan training data, contextual knowledge is used to choose regions of the country that are ethnically homogeneous, and then the voter registers from those areas are used to create the conditional distribution for each ethnic group.

The estimand (e.g., the racial group proportions to be estimated) is manipulated as well as size of the target list of names, both of which are expected to influence estimator performance. For the US data, I generated six sets of racial group parameters from which to generate target lists of differing sample

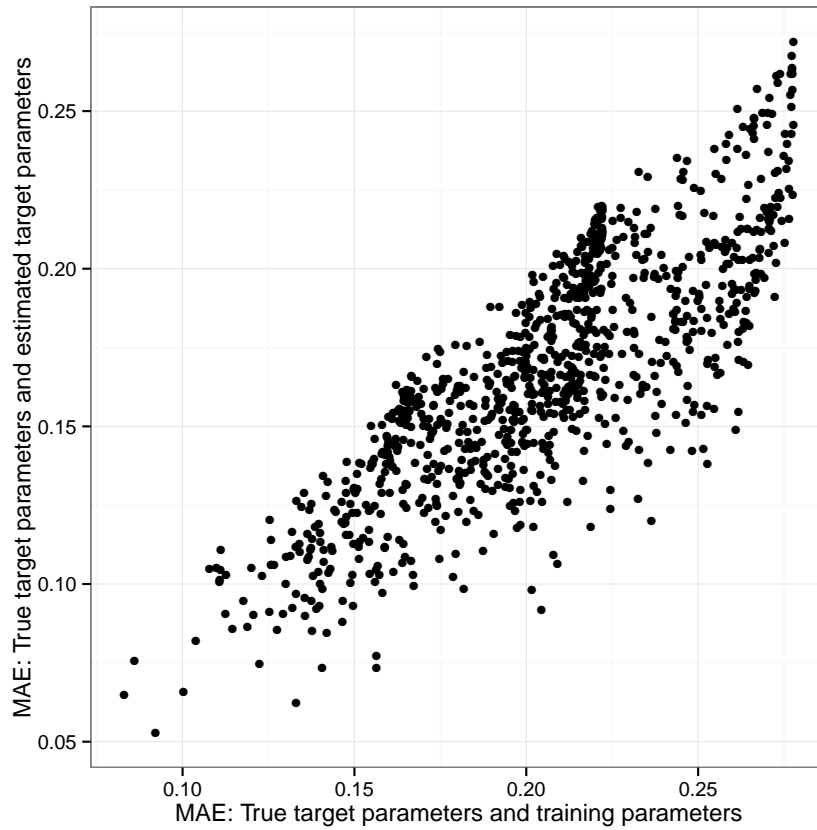


Figure 1: PERFORMANCE OF CLASSIFY-AND-AGGREGATE APPROACH DEPENDS ON SIMILARITY BETWEEN THE TARGET AND TRAINING DATASETS: Horizontal axis shows the mean absolute error of the “true” parameters generating the target data and the target data parameters. The vertical axis represents the MAE of the “true” target data parameters and the estimates of those parameters, given the training data parameters.

sizes ³. Table 1 presents the parameter sets used by row. The first row contains uniform parameters, where each of the six groups comprises one-sixth of the population. The remaining parameter sets define various mixtures of the racial groups. For each combination of parameter set and target list size, I generated 5000 simulated datasets and attempted to recover the estimand using both of the classify-and-aggregate distributions ($P(Group|Name)$ and $P'(Group|Name)$), and four variations on the proposed estimator (which are described below.)

The following routine was used to generate a simulated dataset. For a given parameter set and target list size, the parameter set is multiplied by the target list size N to get the number of names to draw for each group. For instance, for the first parameter set in table 1 and $N = 2000$, each group would comprise one-sixth of the names on the target list, meaning the sample size from each group would be approximately 333 names. Then, for each group j , a sample equal to the group’s defined sample size is drawn from $P(Name|Group j)$ with replacement (as more than one individual may have the same name) as defined by the US census name data. Once each group’s names is sampled from the relevant $P(Name|Group j)$ vector, those unlabelled names are combined into one simulated target dataset of approximately N names.

A classify-and-aggregate estimator based on $P(Group|Name)$ and $P'(Group|Name)$ that draws from the full conditional distribution for each name is deployed.⁴ Doing so enables a more stringent test, as this approach allows all information about a name to be used, rather than linking each name with one group. For each name in a target list, $P(Group|Name)$ or $P'(Group|Name)$ are sampled to get a name-specific classification. Then, these classifications are aggregated for the target list and the racial proportion estimates for the target list are calculated.

The efficacy of the inverse exponentially-weighted Cook’s Distance estimator described in the main text is tested, in addition to the standard approach with no bias correction. No attempt to individually classify names occurs in the proposed approach, as it works directly with a vector of the proportions of each name in the target list. Code used for implementing the proposed method can be found in section 5.

The tables below present Monte Carlo results for both the US and Kenya. Five thousand Monte Carlo simulations were carried out on each parameter-set/sample-size combination. Mean absolute error is calculated as the average absolute difference between each parameter in a parameter set and its true

Table 1: Parameter sets (by row) for US MC simulations.

White	Black	Asian	Am. Ind.	Mixed	Hispanic
.167	.167	.167	.167	.167	.167
.300	.300	.000	.000	.100	.300
.550	.250	.050	.050	.000	.100
.050	.800	.050	.025	.050	.025
.800	.050	.050	.000	.050	.050
.010	.11	.020	.000	.010	.850

value as defined in tables 1 and 2.

2.1. *US MC Results*

Table 2: Parameter sets (by row) for Kenyan Monte Carlo simulations.

Kalenjin	Kamba	Kikuyu	Kisii	Luhya	Luo	Masai	Meru	M'kenda	Pokot	Somali	Turkana
.083	.083	.083	.083	.083	.083	.083	.083	.083	.083	.083	.083
.400	.100	.200	.050	.050	.050	.025	.025	.025	.025	.025	.025
.050	.100	.400	.050	.050	.200	.025	.025	.025	.025	.025	.025
.100	.250	.250	.050	.050	.250	.025	.025	.000	.000	.000	.000
.100	.000	.100	.050	.250	.150	.000	.100	.000	.000	.250	.000
.125	.025	.100	.050	.300	.300	.000	.100	.000	.000	.000	.000
.025	.025	.800	.050	.000	.000	.000	.100	.000	.000	.000	.000
.800	.025	.025	.050	.000	.000	.000	.100	.000	.000	.000	.000

Table 3: MAE of Classify-and-Aggregate, Uniform Population Parameters, US Data.

	1	2	3	4	5	6
N=2000	0.01	0.09	0.11	0.15	0.16	0.09
N=4000	0.00	0.09	0.11	0.15	0.16	0.09
N=8000	0.00	0.09	0.11	0.15	0.16	0.09
N=16000	0.00	0.09	0.11	0.15	0.16	0.09
N=32000	0.00	0.09	0.11	0.15	0.16	0.09

Table 4: MAE of Classify-and-Aggregate, US Population Parameters, US Data.

	1	2	3	4	5	6
N=2000	0.12	0.09	0.05	0.20	0.03	0.08
N=4000	0.12	0.09	0.05	0.20	0.03	0.08
N=8000	0.12	0.09	0.05	0.20	0.03	0.08
N=16000	0.12	0.09	0.05	0.20	0.03	0.08
N=32000	0.12	0.09	0.05	0.20	0.03	0.08

Table 5: MAE of Standard Approach, No Weighting, US Data.

	1	2	3	4	5	6
N=2000	0.06	0.05	0.04	0.04	0.04	0.01
N=4000	0.04	0.03	0.02	0.03	0.03	0.01
N=8000	0.03	0.02	0.02	0.02	0.02	0.01
N=16000	0.02	0.01	0.01	0.02	0.01	0.01
N=32000	0.01	0.01	0.01	0.01	0.01	0.01

Table 6: MAE of Standard Approach, Weighting for Bias Reduction, US Data.

	1	2	3	4	5	6
N=2000	0.06	0.05	0.03	0.03	0.03	0.01
N=4000	0.04	0.03	0.02	0.02	0.02	0.01
N=8000	0.03	0.02	0.01	0.02	0.02	0.01
N=16000	0.02	0.01	0.01	0.01	0.01	0.01
N=32000	0.01	0.01	0.01	0.01	0.01	0.01

2.2. Kenya MC Results

Table 7: Ratio of MAE of Proposed Method to Classify-and-Aggregate, Kenya Data.

	1	2	3	4	5	6	7	8
N=2000	6.11	0.66	0.61	0.40	0.49	0.42	0.16	0.17
N=4000	4.65	0.41	0.38	0.25	0.31	0.26	0.11	0.10
N=8000	3.69	0.27	0.25	0.17	0.21	0.18	0.07	0.07
N=16000	3.42	0.18	0.16	0.12	0.15	0.13	0.05	0.05
N=32000	3.29	0.12	0.10	0.08	0.11	0.09	0.04	0.04

Table 8: Ratio of MAE Weighted Proposed Method to Unweighted Proposed Method, Kenya Data.

	1	2	3	4	5	6	7	8
N=2000	1.00	0.99	0.98	0.96	0.95	0.95	0.89	0.96
N=4000	0.99	0.97	0.95	0.91	0.93	0.92	0.84	0.94
N=8000	0.94	0.95	0.92	0.87	0.90	0.88	0.81	0.92
N=16000	0.91	0.91	0.88	0.85	0.88	0.87	0.77	0.91
N=32000	0.91	0.88	0.85	0.83	0.85	0.84	0.75	0.87

Table 9: MAE for Classify-and-Aggregate Approach, Kenya Data.

	1	2	3	4	5	6	7	8
N=2000	0.00	0.02	0.03	0.03	0.02	0.02	0.05	0.04
N=4000	0.00	0.02	0.03	0.03	0.02	0.02	0.05	0.04
N=8000	0.00	0.02	0.03	0.03	0.02	0.02	0.05	0.04
N=16000	0.00	0.02	0.03	0.03	0.02	0.02	0.05	0.04
N=32000	0.00	0.02	0.03	0.03	0.02	0.02	0.05	0.04

Table 10: MAE for Standard Approach, No Weighting, Kenya Data.

	1	2	3	4	5	6	7	8
N=2000	0.02	0.02	0.02	0.01	0.01	0.01	0.01	0.01
N=4000	0.01	0.01	0.01	0.01	0.01	0.01	0.00	0.00
N=8000	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.00
N=16000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
N=32000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 11: MAE for Standard Approach, Weighting for Bias Reduction, Kenya Data.

	1	2	3	4	5	6	7	8
N=2000	0.02	0.02	0.02	0.01	0.01	0.01	0.01	0.01
N=4000	0.01	0.01	0.01	0.01	0.01	0.01	0.00	0.00
N=8000	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.00
N=16000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
N=32000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

3. VERIFICATION EXERCISES: FLORIDA AND NORTH CAROLINA

The previous section demonstrated the performance of the proposed estimator on simulated data. This section shows that the method outperforms the alternative in North Carolina and Florida.

3.1. *Data Preparation*

The names in the North Carolina and Florida data were cleaned to correspond to cleaning done on the Census data. This entailed removing punctuation and spaces and capitalizing all letters (e.g., “O’Reilly” becomes “OREILLY”). For hyphenated names, I randomly chose one of the two names to represent that individual. These steps ensured comparability between the target and training data and minimized the amount of data discarded due to differences in how a given name was recorded.⁵

There is striking consonance between the categories provided in the North Carolina and Florida voter registers⁶ (which comprise the target data for the verification exercises) and the census names data.⁷ The main difference is that the “Hispanic” category exists in the Census data, but, for the NC and FL data, it is contained in a separate field that records whether or not the voter is Hispanic. I combined these two columns in the voter register data, where the value for a person’s race is replaced by “Hispanic” only when that person is identified as Hispanic in the separate field.

Next, the six racial proportions from each county’s voter register were estimated using the classify and aggregate approaches (uniform population parameters) in addition to the proposed approach with and without the bias reduction weighting.⁸ Finally, the mean absolute error (MAE) for each county was calculated as the average absolute difference between the actual group proportion (calculated using data on race from voter register) and the estimated group proportion for each group. Both states included categories for “other” and “unknown” in their data. As the Census did not provide name data for those groups, those estimates were assumed to be equal to zero for the estimators tested, and were included in the MAE calculations.

Figure 2 compares the MAE for all 100 counties in North Carolina for all approaches using box-and-whiskers plots. The leftmost column presents classify-and-aggregate results based on $P'(\text{Group}|\text{Name } i)$, with a median MAE of about 0.13. The middle column shows the improved approach, with a median MAE near .03. The rightmost column shows the efficacy of the bias

reduction technique, further reducing the mean and dispersion of MAE.⁹

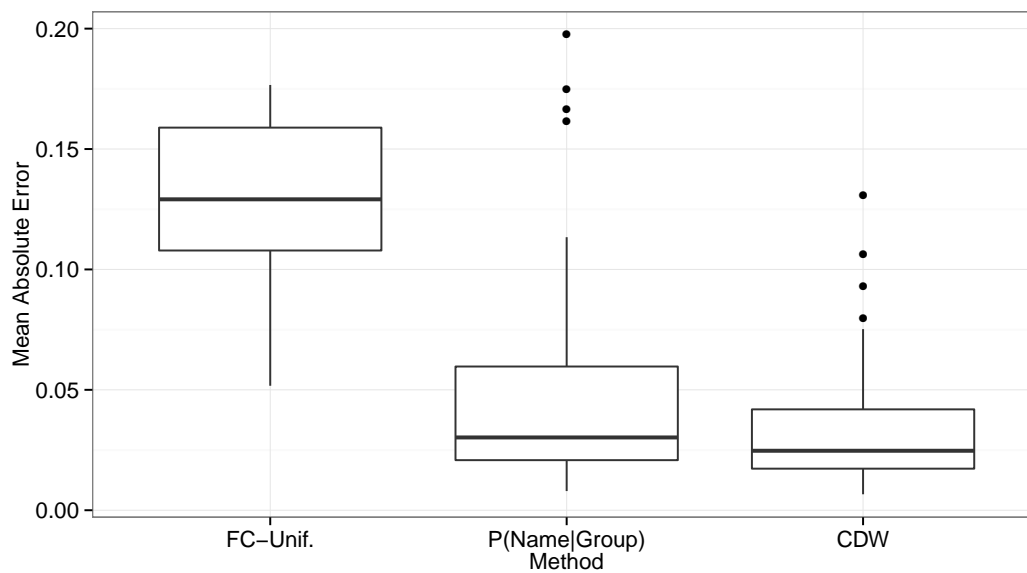


Figure 2: NORTH CAROLINA COUNTY-LEVEL MAE: Distribution of mean absolute error for each of the 100 NC counties in a box-and-whiskers plot for each estimation strategy. The leftmost column presents the standard approach based on $P'(\text{Group}|\text{Name } i)$. The middle column shows improvement in MAE using the proposed approach based on $P(\text{Name} | \text{Group})$. The rightmost column finds additional improvement from the bias-reduction weighting.

Figure 3 compares the MAE for all 67 counties in Florida for all three approaches using box-and-whiskers plots. Results are similar to North Carolina. The leftmost column is uniform parameter classify-and-aggregate approach based on $P'(\text{Group}|\text{Name } i)$, with a median MAE of about 0.13. The middle column shows the improved approach, with a median MAE near .04. The rightmost column shows the efficacy of the bias reduction technique, further reducing the mean and dispersion of MAE.

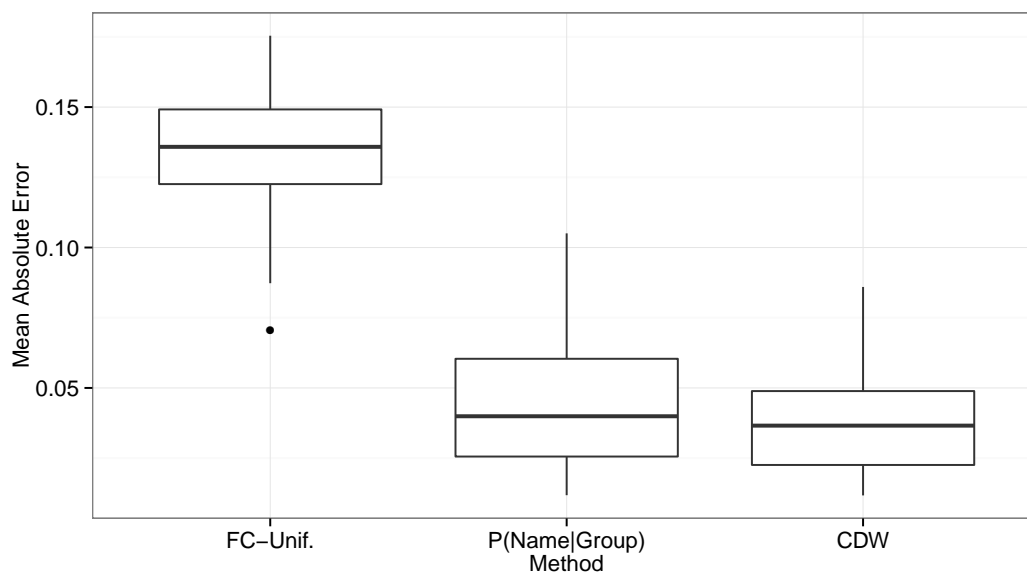


Figure 3: FLORIDA COUNTY-LEVEL MAE: Distribution of mean absolute error for each of the 67 FL counties in a box-and-whiskers plot for each estimation strategy. Results are similar to those for North Carolina.

3.2. Model Fit and Mean Absolute Error

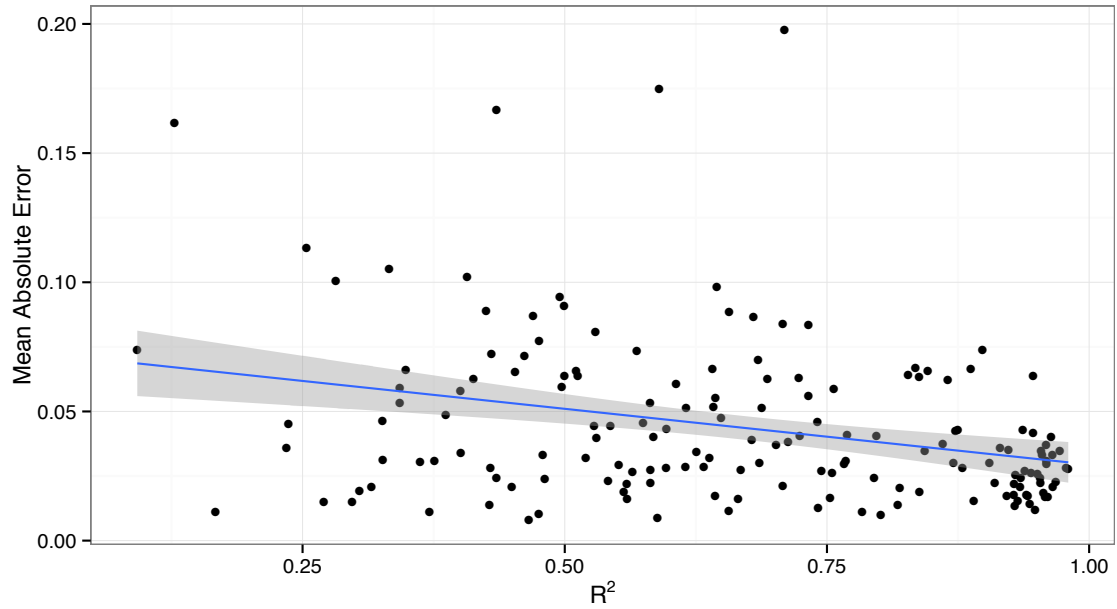


Figure 4: BETTER MODEL FIT AND LOWER MEAN ABSOLUTE ERROR: Higher R^2 is negatively related to mean absolute error. These data – from county-level data in Florida and North Carolina – suggest that better model fit corresponds to lower MAE.

4. APPLICATION: BLACK TURNOUT IN NORTH CAROLINA

Table 12: North Carolina Summary Statistics by County, Black Voters.

County	No. Black	% Black	County	No. Black	% Black
ALAMANCE	21369	0.20	JOHNSTON	19695	0.16
ALEXANDER	1176	0.04	JONES	3009	0.35
ALLEGHANY	92	0.01	LEE	8297	0.22
ANSON	8459	0.44	LENOIR	18841	0.42
ASHE	115	0.01	LINCOLN	3375	0.06
AVERY	105	0.01	MACON	216	0.01
BEAUFORT	9747	0.26	MADISON	163	0.01
BERTIE	9786	0.60	MARTIN	8947	0.44
BLADEN	9656	0.38	MCDOWELL	975	0.03
BRUNSWICK	9570	0.10	MECKLENBURG	229647	0.32
BUNCOMBE	12634	0.06	MITCHELL	23	0.00
BURKE	3928	0.06	MONTGOMERY	4130	0.22
CABARRUS	20844	0.16	MOORE	9484	0.13
CALDWELL	3185	0.05	NASH	28578	0.38
CAMDEN	1243	0.15	NEWHANOVER	24645	0.14
CARTERET	3050	0.05	NORTHAMPTON	10065	0.58
CASWELL	6214	0.36	ONSLOW	19746	0.19
CATAWBA	10483	0.09	ORANGE	15612	0.12
CHATHAM	7164	0.14	PAMLICO	2172	0.20
CHEROKEE	239	0.01	PASQUOTANK	12555	0.39
CHOWAN	3744	0.32	PENDER	7479	0.18
CLAY	39	0.00	PERQUIMANS	2651	0.24
CLEVELAND	15288	0.22	PERSON	7892	0.28
COLUMBUS	13255	0.32	PITT	43919	0.34
CRAVEN	18063	0.23	POLK	826	0.05
CUMBERLAND	95258	0.40	RANDOLPH	6221	0.06
CURRITUCK	1113	0.06	RICHMOND	11277	0.33
DARE	641	0.02	ROBESON	24769	0.29
DAVIDSON	10791	0.09	ROCKINGHAM	13381	0.20
DAVIE	2121	0.07	ROWAN	17552	0.17
DUPLIN	10475	0.32	RUTHERFORD	5139	0.11
DURHAM	90706	0.38	SAMPSON	13291	0.32
EDGECOMBE	26529	0.60	SCOTLAND	9999	0.40
FORSYTH	75782	0.28	STANLY	4888	0.11
FRANKLIN	12326	0.28	STOKES	1592	0.05
GASTON	22958	0.16	SURRY	2032	0.04
GATES	3146	0.35	SWAIN	156	0.01
GRAHAM	3	0.00	TRANSYLVANIA	1050	0.04
GRANVILLE	13579	0.34	TYRRELL	965	0.33
GREENE	4859	0.38	UNION	18387	0.12
GUILFORD	133422	0.34	VANCE	17410	0.52
HALIFAX	22615	0.53	WAKE	148745	0.21
HARNETT	17670	0.23	WARREN	8518	0.53
HAYWOOD	480	0.01	WASHINGTON	5196	0.50
HENDERSON	2462	0.03	WATAUGA	970	0.02
HERTFORD	10567	0.62	WAYNE	28153	0.34
HOKE	12443	0.39	WILKES	1914	0.04
HYDE	1003	0.25	WILSON	25068	0.41
IREDELL	14617	0.12	YADKIN	871	0.03
JACKSON	574	0.02	YANCEY	107	0.01

Listing 1: R code to implement the proposed method.

```
1 require(corpcor)
2 require(quadprog)
3
4 nameEst <- function(cond, targ){
5   Y <- targ
6   X <- cond
7   Yt <- Y
8   orderind <- intersect(names(Yt), rownames(X))
9   Yt <- Yt[orderind]
10  X <- X[orderind,]
11  Yt <- Yt[row.names(X)]
12  designmat <- model.matrix(Yt ~ -1 + X)
13  Dmat <- crossprod(designmat, designmat)
14  pdind <- is.positive.definite(Dmat)
15  if(pdind == FALSE){Dmat <- make.positive.definite(Dmat)}
16  dvec <- crossprod(designmat, Yt)
17  Amat <- cbind(matrix(1, nr=length(dvec), nc=1), diag(length(dvec)), -diag(length(dvec)))
18  bvec <- c(1, rep(0, length(dvec)), rep(-1, length(dvec)))
19  meq <- 1
20  resTmp <- tryCatch(expr = solve.QP(Dmat, dvec, Amat, bvec, meq, factorized = F)[[1]],
21    error = function(e) rep(NA, ncol(X)))
22  out <- round(resTmp, 5)
23  names(out) <- colnames(X)
24  return(out)
25 }
```

5. CODE

This code uses the quadratic programming methods described in Goldfarb and Idnani (1982, 1983) to estimate the parameters of interest. Implementation is in R using the `quadprog` package.

Notes

¹Though it is worth mentioning that if the researcher knows that the target population resembles the training population, it is unclear why she would undertake this endeavor in the first place. This suggests another rationale for using uniform population parameters as done below: they represent a flat prior favoring no single group.

²These groups are: Hispanic, Black, White, American Indian, Asian/Pacific Islander, and Mixed Race. These data are available at <http://www2.census.gov/topics/genealogy/2000surnames/names.zip>.

³ $N = 2000, 4000, 8000, 16000, 32000$; for the Kenyan data, we generated eight sets of group parameters, given that there are more groups in the Kenyan data.

⁴For the Kenyan data, only $P'(Group|Name)$ (uniform population parameters) is considered because, as is likely in most settings where the proposed method would be applicable, name counts by racial group from the census do not exist.

⁵Florida voter registration data was downloaded from <http://flvoters.com/downloads.html> and North Carolina voter registration data can be downloaded from <ftp://alt.ncsbe.gov/ENRS/>.

⁶The categories are BLACK or AFRICAN AMERICAN, AMERICAN INDIAN or ALASKA NATIVE, OTHER, WHITE, UNDESIGNATED, ASIAN, and TWO or MORE RACES.

⁷The categories are WHITE, BLACK, ASIAN PACIFIC ISLANDER, AMERICAN INDIAN, TWO or MORE RACES, and HISPANIC.

⁸For the classify-and-aggregate approach, 100 draws were taken from the conditional, and the mean was taken as the final estimate.

⁹The four outliers (black dots) in figure 2 are counties with relatively high American Indian population concentrations, which are easily conflated with White and Black names, generating high MAE.

REFERENCES

- Goldfarb, D. and A. Idnani (1982). Dual and primal-dual methods for solving strictly convex quadratic programs. In J. P. Hennart (Ed.), *Numerical Analysis*, pp. 226–239. Berlin: Springer-Verlag.
- Goldfarb, D. and A. Idnani (1983). A numerically stable dual method for solving strictly convex quadratic programs. *Mathematical Programming* 27, 1–33.
- Word, D., C. Coleman, R. Nunziata, and R. Kominski (n.d.a). Data accompanying “Demographic Aspects of Surnames from Census 2000”. Data, United States Census Bureau, Washington, D.C.
- Word, D., C. Coleman, R. Nunziata, and R. Kominski (n.d.b). Demographic aspects of surnames from Census 2000. Technical report, United States Census Bureau, Washington, D.C.