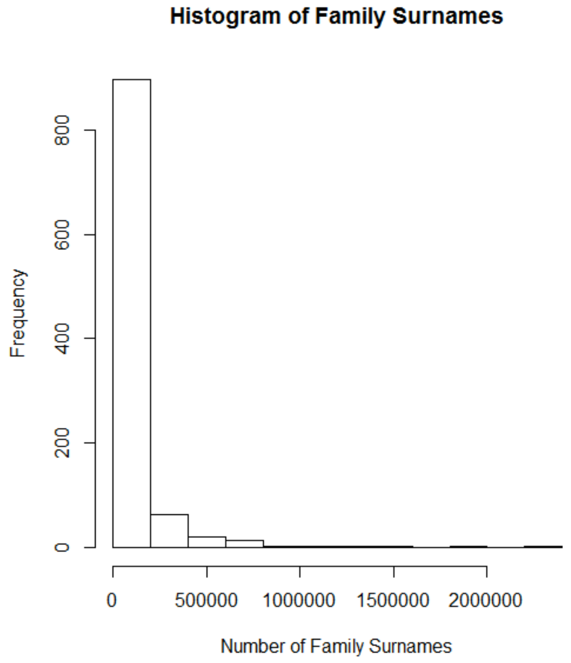
Fitting Power Law Distributions to Data

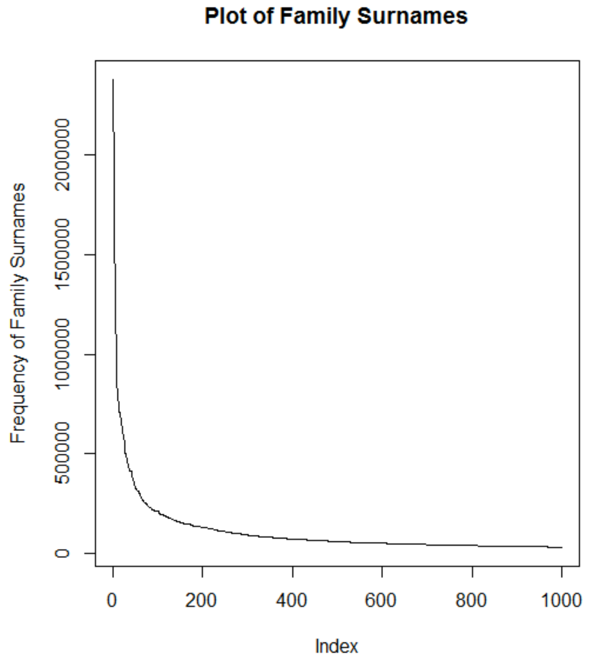
Willy Lai

Introduction

In this paper, we will be testing whether the frequency of family names from the 2000 Census follow a power law distribution. Power law distributions are usually used to model data whose frequency of an event varies as a power of some attribute of that event. In our case, we will see if the frequency of family names vary as a power of the family names itself. In other words, we will see if there are a few family names that are very common and if there are many family names that are not as common. We will extract 1000 family names from this website: http://www.census.gov/genealogy/www/data/2000surnames/index.html, upload the data in R, and analyze the counts for each family name. Clearly the family names are ranked according to frequency, from largest to smallest, which can make it easier for us to follow what we are analyzing. Our procedure for analyzing the data will follow the procedure in the paper: POWER-LAW DISTRIBUTIONS IN EMPIRICAL DATA, while using R code to implement them.

Observing the data:





Based on the histogram and plot of the family surnames, it seems that the shape of the curve and histogram follows some kind of power law distribution. However, the power law does not seem like the only distribution that can fit the data of family names, and we will test other possible distributions later in the paper.

Estimating Parameters

The first step we need to do is to estimate the two important parameters alpha and x_{min}. The trickiest part of this is estimating x_{min}. The procedure we used involves using the Kolmogorov-Smirnoff(KS) statistic, page 11 of the paper, in which we would find the x_{min} that minimizes the value of the KS statistic.

To find the best possible x_{min} value, we run through our data set to and use each data as our $x_{min,}$ truncate our data to include every data above and including our chosen $x_{min,}$ use these data to compute empirical cdf and the theoretical cdf $P(x) = (x/x_{min})$^(-α+1), where x is our data, and take the maximum of the absolute value of the difference between each theoretical and empirical cdf value, which is the KS statistic. Note that the cdf of the power law given in the paper is a complementary cdf, since P(x) was computed by integrating the pdf of the power law from x to infinity. Thus, we would need to compare the power law cdf to the vector (1,(n-1)/n,...,2/n,1/n), which is the empirical cdf. We do this for each data point and compile the KS statistics into one vector. Afterwards, we then pick the minimum of the KS statistics, find the corresponding x_{min} value, and designate this to be our parameter x_{min}.

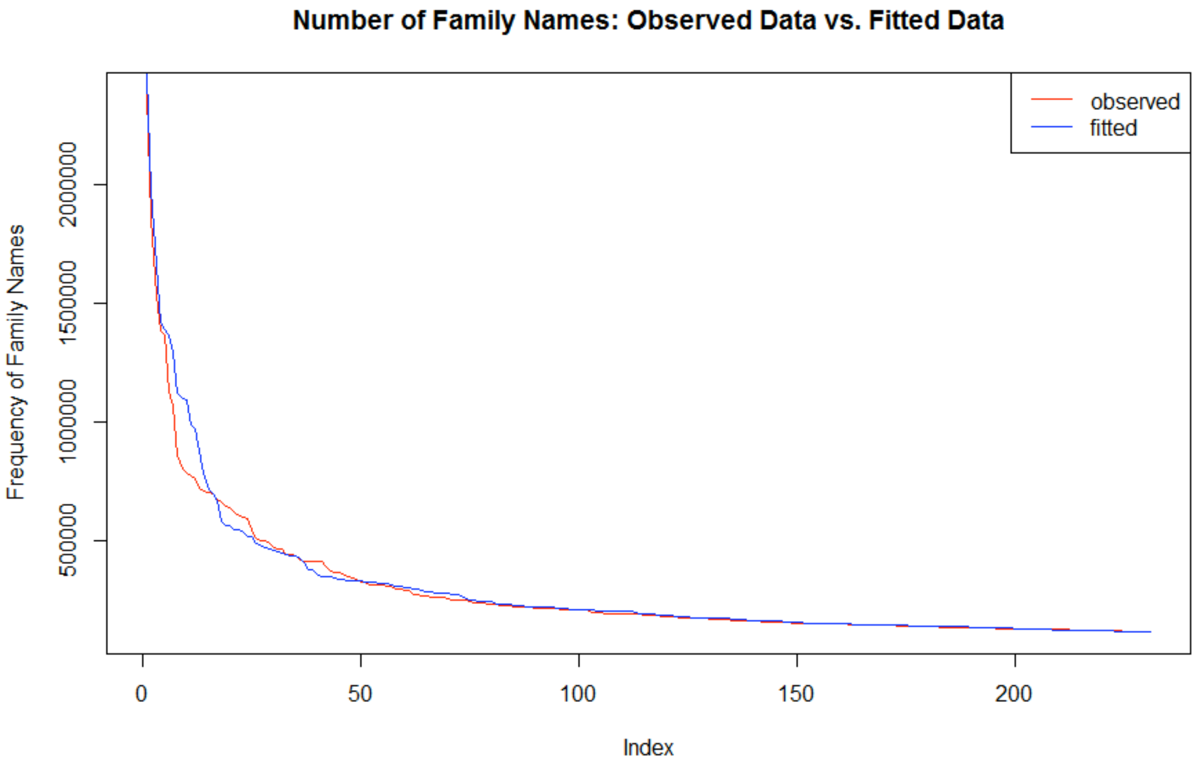
Finding the parameter α is not as tricky, since we have our x_{min} and we would just use the MLE estimate to compute a plausible value for α. After performing these necessary steps, we estimate our parameters to be x_{min}=117939 and α = 2.542679.

Goodness of Fit

To test how well our power law distribution fits our observed data, we will perform the Kolmogorov Smirnoff (KS) test to see if the generated data from the power law distribution with our chosen parameters and the observed data come from the same distributions. We do this by following the paper's procedure, which involves generating a large number of synthetic data sets of the power law distribution with our chosen parameters x_{min}=117939 and α = 2.542679. We then compare each of these data sets to our observed data and see if these two data sets come from a similar distribution by performing the KS test for the observed data and generated data. To determine when we would reject or fail to reject for each KS test, we will use the significance level of 0.10, which means if the p value is greater than .10, we fail to reject the null and conclude that both data sets come from the same distribution, whereas if the p-value is less than or equal to 0.10, we reject the null and conclude that the data sets do not come from a power law distribution. Since we chose the our significance level to be about 0.10, we would expected about 10% or less of the tests to reject the null hypothesis. If this is the case, then our power law distribution with our chosen parameters is a good fit.

Now we just need to choose how many KS tests to perform. We would want our p-values to be correct within two decimal points, since we want to compare our p-value to our chosen significance level 0.10. Thus, we would follow the suggestion of the paper and perform 2500 KS tests to determine whether the observed data and the synthetic data sets generated from a power law random generator with our chosen parameters are from the same distribution. This will help us determine whether the power law distribution with our chosen parameters is a good fit to the data on number of family names.

After performing 2500 tests, 2483 of the tests failed to reject the null hypothesis, meaning that less than 10% of the tests rejected the null. This means that our power law distribution fit is a good fit to the data of family names.



One of the curves of the above plot includes the observed data of the family names and a data set of randomly generated power law distribution with the parameters x_{min}=117939 and α = 2.542679. This graph is an example of how a randomly generated data of power law distribution is very closely related to the observed data of family names, which suggests that the family names do follow the power law distribution very closely.

Alternative Distributions

Just because we came to the conclusion that the power law distribution is a good fit to the data of family names, it does not mean that the power law is the best fit. There can be other distributions that can be just as good or even a better fit. Other possible distributions that can potentially be a good fit to our data are the exponential and log-normal distributions. Thus, we need to fit an exponential and log-normal distribution to our data and perform a goodness of fit test to see if these fits are any good. We picked these two distributions to test because they seem to be the next closes distributions that could fit our data. The procedures for fitting exponential and log normal distributions and testing their goodness of fit will be similar to the procedures for the power law distribution. To ensure that either or both of these distributions are a good fit or not, we will fit each of these distributions with and without x_{min} values. If either the exponential and log-normal distributions is a good fit to our data, then this means that the power law is not necessarily the best fit for the data. If both the exponential and log-normal distributions are not good fits, then the power law is a decent fit to the model.

i) Exponential w/ x_{min}:

Estimated Parameters:

```
> xmin
[1] 276400
```

Note, when calculating the KS statistic, we compared the cdf of the exponential with the vector (1/n,2/n,...,(n-1)/n,1), since the cdf of the exponential here is the integration of the pdf from negative infinity to x, unlike the cdf of the power law. This same method will be used for the log normal distribution with x_{min}.

```
> lambda
[1] 3.088112e-06
```

After performing 2500 KS tests, none of the KS test fails to reject the null, which means the exponential data sets and the family name data sets do not come from the same distribution. This implies that number of family names do not follow an exponential distribution.

ii) Exponential w/out xmin:

Estimated Parameter:

```
> lambda2
[1] 9.137274e-06
```

Similarly, none of the KS test fails to reject the null, further proving that the number of family names do not follow an exponential distribution.

Thus, the exponential distribution is not a good fit to the data of family names.

iii) Log Normal w/ x_{min}:

Estimated Parameters:

```
> mu
[1] 13.01822
> sigmasq
[1] 0.2733221
```

After performing the KS tests, none of the tests fail to reject the null hypothesis, which means the number of family names do not follow the log-normal distribution.

iv) Log Normal w/out x_{min}:

Estimated Parameters:

```
> mu2
[1] 11.21085
> sigmasq2
[1] 0.5531002
```

None of the KS tests fail to reject the null, further proving that the data on family names do not follow the log-normal distribution.

Thus, the log-normal distribution do not fit the data on family names.

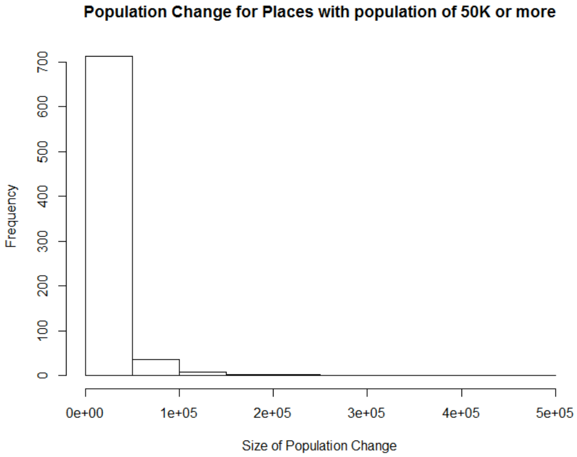
Conclusion

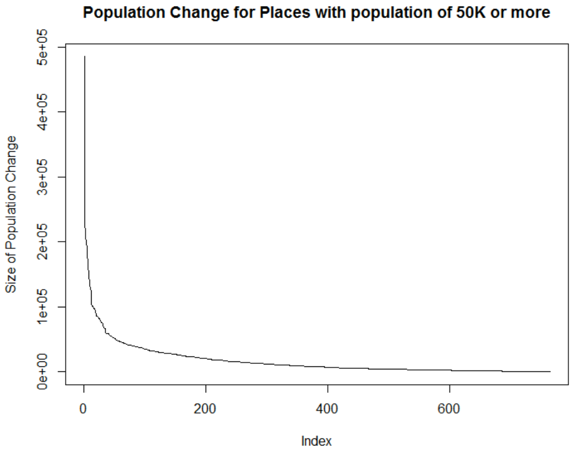
Overall, we have ruled out the possibility that the closest distributions, the exponential and log-normal distributions, can be a good fit to the data of family names. This means that the likelihood ratio test is not necessary, since based on our tests, the power law is a good fit, while the exponential and log-normal distributions are not. Therefore, we can conclude that the power law distribution is a good fit to the data of family names from the 2000 Census.

Other Data:

Other data sets were obtained and analyzed using the same methods as described above. For each data set, the plot and histogram of the data set are shown, the estimated parameters are stated for the distribution being analyzed, and what the tests results say regarding whether the distribution is a good fit or not.

POPULATION CHANGE OF CITIES WITH 50K POPULATION OR MORE





POWER LAW:

The parameters calculated for the power law distribution were xmin = 28167 and α = 2.764328. The test results were that 2451 out of 2500 KS tests failed to reject the null hypothesis that the data were from different distributions. Since 2451/2500 is clearly greater than 90%, this means the power law distribution could be a good fit.

EXPONENTIAL w/ xmin:

The parameters calculated are xmin = 9343 and lambda = 3.964158e-05. None of the ks tests failed to reject the null, which means the exponential distribution with the xmin parameter were a good fit

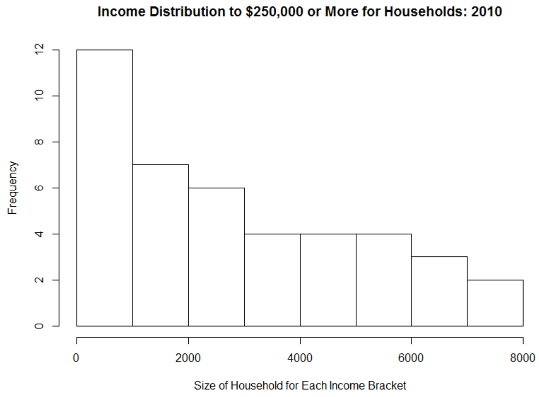
EXPONENTIAL:
The parameter calculated to fit an exponential distribution is lambda = 5.673477e-05. Similar to the exponential with xmin, all of the KS test rejected the null hypothesis that the data and generated data from the exponential distribution came from the same distribution.

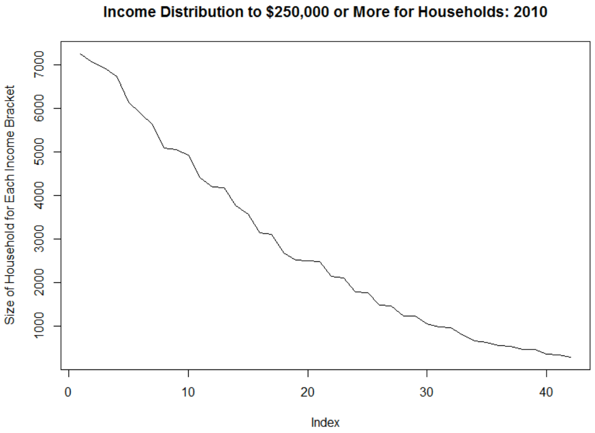
LOG NORMAL W/ Xmin:
The parameters calculated were $\mu = .999135$ and $\sigma^2 = 1.754466$. 1781 out of 2500 KS tests failed to reject the null, which is only 71.24 percent of the tests that failed to reject, meaning more than 10% rejected the null. This implies that the log normal with the xmin parameter is not a good fit to the data.

LOG NORMAL W/out xmin:
The parameters calculated are $\mu = 8.852362$ and $\sigma^2 = 2.303018$. About 1935 out of 2500 tests failed to reject the null, which is 77.4 percent of the tests failed to reject, meaning more than 10 percent of the tests failed to reject. Thus, the log normal distribution without the xmin parameter is not a good fit.

Overall, since the exponential and log-normal distributions, with or without the xmin parameter, were not good fits to the data set and only the power law fit, this means the power law distribution is a good fit to this data set.

DATA ON INCOME OF HOUSEHOLDS





POWER LAW:
The parameters calculated are $\alpha = 6.041705$ and xmin = 4933. About 2481 out of 2500 KS tests failed to reject the null, which suggests that the power law distribution may be a good fit.

EXPONENTIAL W/ XMIN:
The parameters calculated are $\lambda = 0.0003855868$ and xmin = 358. About 2220 out of 2500 KS tests failed to reject the null, which is about 88.8 % of the tests, which means the exponential distribution with xmin is not a good enough fit.

EXPONENTIAL

The parameter calculated is λ = 0.0003538839. About 2475 out of 2500 KS tests, that is about 99% of the tests, failed to reject the null, which suggests an exponential distribution is a good fit.

LOGNORMAL w/ XMIN

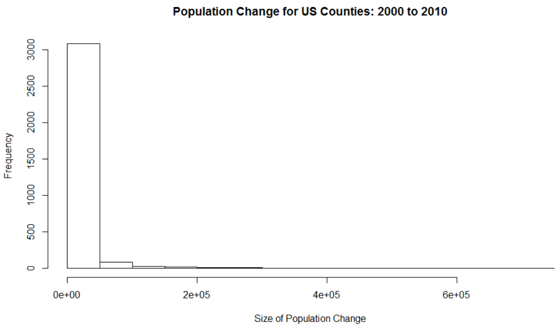
The parameters calculated are xmin = 1236, μ = 8.107043, σ^2 = 0.2988621. About 34 out of 2500 KS tests reject the null, which means the log normal distribution with the xmin parameter is not a good fit.

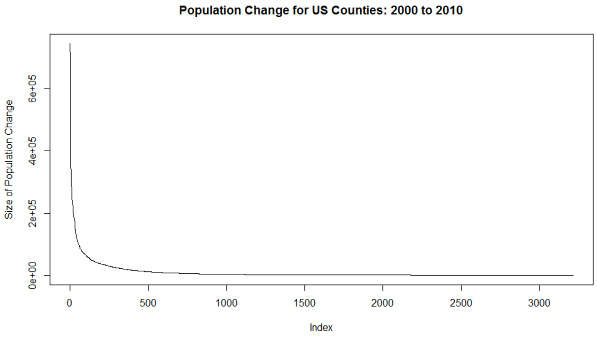
LOGNORMAL w/out XMIN

The calculated parameters are μ = 7.565269 and σ^2 = 0.9121128. About 2461 out of 2500 tests(about 98.44%) failed to reject the null, which suggests that the log normal distribution is a good fit to the data.

Overall, it seems that the power law distribution, exponential, and log normal distributions are good fits to the data, despite not seeming so when observing the graphs alone. This means that the power law distribution is not necessary a good fit as there are other distributions that could be a better fit to the data.

DATA ON POP CHANGE IN US COUNTIES:





POWER LAW:

The parameters calculated are α = 2.378352 and xmin = 34262. About 2440 out of 2500 KS tests (about 97.6% of the tests) fail to reject the null, which means the power law distribution may be a good fit.

EXPONENTIAL W/ XMIN:

The calculated parameters are λ = 8.367428e-06 and xmin = 107311. None of the ks tests fail to reject the null, which means the exponential with xmin does not fit the data.

EXPONENTIAL REGULAR:

The parameter calculated is λ = 0.0001020548. None of the KS tests fail to reject the null, which means the exponential distribution is not a good fit either.

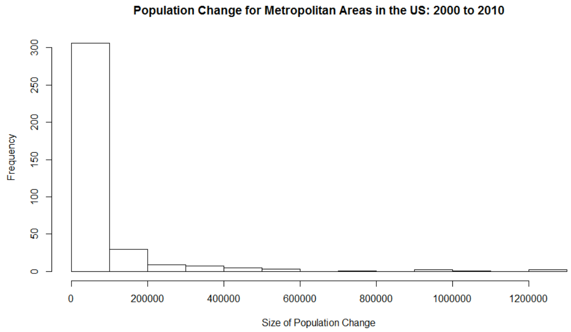
LOGNORMAL W/ XMIN

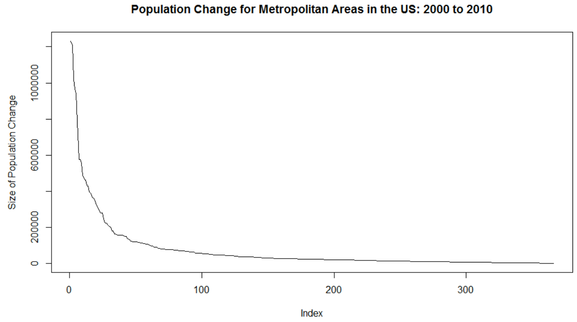
The parameters are μ = 7.289496 and σ^2 = 3.723756, and xmin = 1. About 51 of the 2500 KS tests fail to reject the null, which is clearly less than 10%, which means the log normal distribution with the xmin parameter does not fit the data.

LOGNORMAL REGULAR:
The parameters are μ = 7.289496 and σ^2 = 3.723756. In this case, 46 out of 2500 tests fail to reject the null, which means the log normal distribution without the xmin parameter is not a good fit either. Note that the parameters μ and σ^2 for the log normal distribution without the xmin parameter are the same as that of the log normal distribution with the xmin parameter. This is because the xmin parameter calculated is 1, which means the entire data set were used to fit a power law distribution. Thus, both log normal distributions did lead to the same results.

Overall, since the power law distribution is the only distribution that fits the data, the power law must be a good fit to the data of population change in US counties.

DATA ON POPULATION CHANGE FOR METRO AREAS IN US:





POWER LAW:
The calculated parameters are α = 2.055709 and xmin = 61216. About 2417 of the 2500 KS tests (96.68% of the tests) fail to reject the null, which suggests the power law distribution to be a good fit to the data.

EXPONENTIAL W/ XMIN:
The calculated parameters are λ = 3.457706e-06 and xmin = 161058. About 119 of the 2500 KS tests failed to reject the null, which means the exponential distribution with xmin does not fit the data well.

REGULAR EXPONENTIAL:
The calculated parameter is λ = 1.377229e-05. None of the KS tests fail to reject, which implies that the exponential distribution is not a good fit.

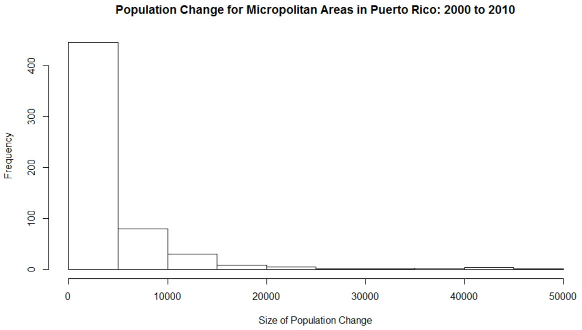
LOG NORMAL TEST W/ Xmin:
The calculated parameters are μ = 10.05736, σ^2 = 2.293322, and xmin = 218. About 2411 out of 2500 (96.44%) of the KS tests fail to reject the null, which means the log normal distribution with xmin may be a good fit to the data.

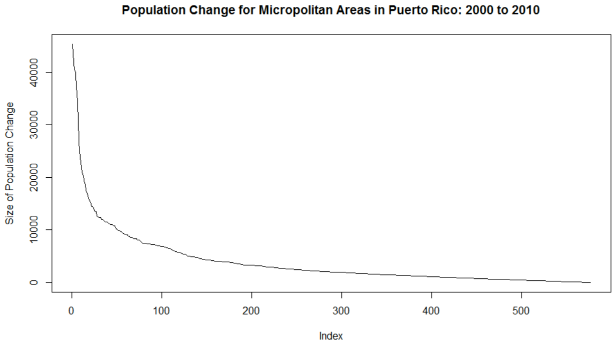
LOG NORMAL REGULAR:

The parameters are μ = 10.05736 and $\sigma^2 = 2.293322$. About 2407 out of 2500 (96.28%) of the KS tests fail to reject the null, which suggests that the log normal distribution without the xmin parameter is a good fit to the data.

Overall, we have that the power law distribution and the log normal distributions, with or without the xmin parameter, are good fits to the data. This means the power law distribution is not necessarily the best fit to the model, as there can be other models that is a good fit.

DATA ON POPULATION CHANGE IN MICROPOLITAN AREAS IN PUERTO RICO:





POWER LAW

The calculated parameters are α = 2.879117 and xmin = 6356. About 2439 out of 2500 (97.56%) of the tests fail to reject the null, which means the power law distribution may be a good fit.

EXPONENTIAL W/ XMIN:

The parameters are λ = 0.0001659792 and xmin = 4036. None of the tests fail to reject the null, which means the exponential with xmin is not a good fit to the data.

EXPONENTIAL REGULAR:

The parameter calculated is λ = 0.000256736. About 44 out of 2500 KS tests fail to reject the null, which means the exponential distribution is not a good fit.

LOG NORMAL W/ XMIN:

The estimated parameters are μ = 7.625002, $\sigma^2 = 1.444097$ and xmin = 93. About 2456 out of 2500 KS tests (98.24% of the tests) fail to reject the null, which means the log normal distribution with the xmin parameter may be a good fit.

LOG NORMAL REGULAR:

The estimated parameters are μ = 7.505166 and $\sigma^2 = 1.939343$. About 1600 out of 2500 KS tests failed to reject the null, which is 64% of the tests. This means that the log normal distribution without the xmin parameter is not such a good fit to the data.

Overall, the only distribution that seems to fit the data well are the power law and the log normal with xmin. This suggests that the power law is not necessarily a good fit to the data since another distribution could fit just as well to the data.

Appendix: R Code

```
#DATA ON FAMILY NAMES:
topfamilynames = read.csv('1000familynames.csv',stringsAsFactors=FALSE)
namecount = topfamilynames[-1,1:3]
names(namecount) = c('Surnames','Rank','Count')
famnamecount = as.numeric(namecount$Count)
data = famnamecount
data = sort(data,decreasing=TRUE)
hist(data,xlab='Number of Family Surnames',main='Histogram of Family Surnames')
plot(data,type='l',ylab='Frequency of Family Surnames',main='Plot of Family Surnames')


#DATA ON POPULATION CHANGE OF CITIES WITH 50K POPULATION OR MORE
changepop = read.csv('pop change 1.csv',stringsAsFactors=FALSE)
changepop1 = changepop[6:784,5]
popchange = as.numeric(gsub(',','',changepop1))
popchange1 = popchange[!is.na(popchange)]
data = abs(popchange1)
data = sort(data,decreasing=TRUE)
hist(data,main='Population Change for Places with population of 50K or more',xlab='Size of
Population Change')
plot(data,type='l',main='Population Change for Places with population of 50K or
more',ylab='Size of Population Change')


#DATA ON SINGLE GRADE ON ENROLLMENT AND HIGH SCHOOL GRADUATION
FOR PEOPLE 3 YEARS AND OLDER: OCTOBER 2006
enroll = read.csv('enrollment.csv',stringsAsFactors=FALSE)
enroll1 = enroll[9:39,3]
enroll2 = as.numeric(gsub(',','',enroll1))
data = enroll1
data = sort(data,decreasing=TRUE)
hist(data,main='Single Grade',xlab='Size of Population Change')
plot(data,type='l',main='Population Change for Places with population of 50K or
more',ylab='Size of Population Change')


#DATA ON INCOME OF HOUSEHOLDS
census = readLines('http://www.census.gov/hhes/www/cpstables/032011/hhinc/new06_000.htm')
pattt = '^([0-9 /,\\*]+)$'
cen1 = grep(pattt,census,value=TRUE)
cen2 = cen1[2:259]
cen3 = matrix(cen2,ncol=6,byrow=TRUE)
cen4 = cen3[-1,1]
cen5 = as.numeric(gsub(',','',cen4))
data = sort(cen5,decreasing=TRUE)
```

```
hist(data,main='Income Distribution to $250,000 or More for Households: 2010',xlab='Size of Household for Each Income Bracket')
plot(data,type='l',main='Income Distribution to $250,000 or More for Households: 2010',ylab='Size of Household for Each Income Bracket')

#DATA ON CHANGE OF POP IN COUNTIES
#http://www.census.gov/population/www/cen2010/cph-t/cph-t-1.html
popmun = read.csv('pop change municipios.csv',stringsAsFactors=FALSE)
popmun1 = popmun[7:3229,5]
popmun2 = as.numeric(gsub(',','',popmun1))
popchange1 = popmun2[!is.na(popmun2)]
data = abs(popchange1)
data = sort(data,decreasing=TRUE)
hist(data,main='Population Change for US Counties: 2000 to 2010',xlab='Size of Population Change')
plot(data,type='l',main='Population Change for US Counties: 2000 to 2010',ylab='Size of Population Change')


#http://www.census.gov/population/www/cen2010/cph-t/cph-t-2.html
#DATA ON POP CHANGE IN METROPOLITAN
popmetro = read.csv('pop change metro.csv',stringsAsFactors=FALSE)
popmetro1 = popmetro[7:372,4]
popmetro2 = as.numeric(gsub(',','',popmetro1))
data = abs(popmetro2)
data = sort(data,decreasing=TRUE)
hist(data,main='Population Change for Metropolitan Areas in the US: 2000 to 2010',xlab='Size of Population Change')
plot(data,type='l',main='Population Change for Metropolitan Areas in the US: 2000 to 2010',ylab='Size of Population Change')


#DATA ON POP CHANGE IN MICROPOLITAN
popmetro = read.csv('pop change metro.csv',stringsAsFactors=FALSE)
popmicro = popmetro[375:950,4]
popmicro2 = as.numeric(gsub(',','',popmicro))
data = abs(popmicro2)
data = sort(data,decreasing=TRUE)
hist(data,main='Population Change for Micropolitan Areas in Puerto Rico: 2000 to 2010',xlab='Size of Population Change')
plot(data,type='l',main='Population Change for Micropolitan Areas in Puerto Rico: 2000 to 2010',ylab='Size of Population Change')


xmins = unique(data) # search over all unique values of data
```

```
dat = numeric(length(xmins))
z = sort(data)

for (i in 1:length(xmins)){
   xmin = xmins[i]                # choose next xmin candidate
   z1 = z[z>=xmin]                  # truncate data below this xmin value
   n = length(z1)
   a = 1+ n*(sum(log(z1/xmin)))^-1     # estimate alpha using direct MLE
   cx = (n:1)/n                   # construct the empirical CDF
   cf = (z1/xmin)^(-a+1)              # construct the fitted theoretical CDF
   dat[i] = max(abs(cf-cx))    # compute the KS statistic
   }

D = min(dat[dat>0],na.rm=TRUE)                     # find smallest D value
xmin = xmins[which(dat==D)] # find corresponding xmin value
z = data[data>=xmin]
z = sort(z)
n = length(z)
alpha = 1 + n*(sum(log(z/xmin)))^-1 # get corresponding alpha estimate

library(gsl)
library(numDeriv)

# the following code, up to the rpowerlaw, came from this website:
http://www.rickwash.com/papers/cscw08-appendix/powerlaw.R

dpowerlaw <- function(x, alpha=2, xmin=1, log=F) {
  if (log)
    log(alpha-1) - log(xmin) - alpha * log(x / xmin)
  else
    ((alpha - 1) / xmin) * ((x / xmin) ^ (-alpha))
}

ppowerlaw <- function(q, alpha=2, xmin=1, lower.tail=T, log.p = F) {
  p <- (q / xmin) ^ (- alpha + 1)
  if (lower.tail)
    p <- 1-p
  if (log.p)
    p <- log(p)
  p
}

qpowerlaw <- function(p, alpha=2, xmin=1, lower.tail=T, log.p = F) {
  if (!lower.tail)
    p <- 1-p
  if (log.p)
```

```
    p <- exp(p)
  xmin * ((1 - p) ^ (-1 / (alpha - 1)))
}

rpowerlaw <- function(n, alpha=2, xmin=1) {
  qpowerlaw(runif(n, 0, 1), alpha, xmin)
}

testresult = numeric(2500)
for (i in 1:2500){
    power = rpowerlaw(length(z),alpha,xmin)   #randomly generate power law data using the
parameters we found
    w = ks.test(z,power)     #using KS test to see how good the fit is
    if (w$p.value > 0.10){
       testresult[i] = 1}
    if (w$p.value <= 0.10){
       testresult[i] = 0}
    }
sum(testresult)

#FITTING AN EXPONENTIAL:
dat2 = numeric(length(xmins))
z = sort(data)

for (i in 1:length(xmins)){
   xmin = xmins[i]              # choose next xmin candidate
   z2 = z[z>=xmin]                # truncate data below this xmin value
   n = length(z2)
   lambda = 1/(mean(z2)-xmin)     # estimate lambda using direct MLE
   cx = (1:n)/n                 # construct the empirical CDF
   cf = 1 - exp(lambda*(xmin-z2))             # construct the fitted theoretical CDF
   dat2[i] = max(abs(cf-cx))    # compute the KS statistic
   }

D = min(dat2[dat2>0],na.rm=TRUE)                    # find smallest D value
xmin = xmins[which(dat2==D)]                     # find corresponding xmin value
z = data[data>=xmin]
z = sort(z)
n = length(z)
lambda = 1/(mean(z)-xmin)

testresult2 = numeric(2500)
for (i in 1:2500){
    expfit = rexp(length(z),lambda)   #randomly generate exponential data using the parameters
we found
    w1 = ks.test(expfit,z)     #using KS test to see how good the fit is
    if (w1$p.value > 0.10){
      testresult2[i] = 1}
    if (w1$p.value <= 0.10){
      testresult2[i] = 0}
    }
sum(testresult2)

#REGULAR EXPONENTIAL TEST:
lambda2 = 1/mean(data)
testresult3 = numeric(length(data))
for (i in 1:2500){
    expfit = rexp(length(data),lambda2)   #randomly generate exponential data using the
parameters we found
    w2 = ks.test(expfit,data)     #using KS test to see how good the fit is
    if (w2$p.value > 0.10){
      testresult3[i] = 1}
    if (w2$p.value <= 0.10){
      testresult3[i] = 0}
    }
sum(testresult3)

LOG NORMAL TEST W/ Xmin:
dat3 = numeric(length(xmins))
z = sort(data)

for (i in 1:length(xmins)){
   xmin = xmins[i]               # choose next xmin candidate
   z3 = z[z>=xmin]                 # truncate data below this xmin value
   n = length(z3)
   mu = sum(log(z3))/length(z3)
   sigmasq = sum((log(z3)-mu)^2)/length(z3)     # estimate lamda using direct MLE
   cx = (1:n)/n                    # construct the empirical CDF
   cf = pnorm((log(z3)-mu)/sqrt(sigmasq))           # construct the fitted theoretical CDF
   dat3[i] = max(abs(cf-cx))    # compute the KS statistic
   }

D = min(dat3[dat3>0],na.rm=TRUE)                     # find smallest D value
xmin = xmins[which(dat3==D)]                     # find corresponding xmin value
z = data[data>=xmin]
z = sort(z)
n = length(z)
mu = sum(log(z))/length(z)
sigmasq = sum((log(z)-mu)^2)/length(z)


testresult4 = numeric(2500)
```

```
for (i in 1:2500){
   lognfit = rlnorm(length(data),mean=mu,sd=sqrt(sigmasq))   #randomly generate exponential
data using the parameters we found
   w3 = ks.test(lognfit,data)     #using KS test to see how good the fit is
   if (w3$p.value > 0.10){
     testresult4[i] = 1}
   if (w3$p.value <= 0.10){
     testresult4[i] = 0}
   }
sum(testresult4)
```

REGULAR LOG NORMAL TEST:

```
mu2 = sum(log(data[data>0]))/length(data[data>0])
sigmasq2 = sum((log(data[data>0])-mu2)^2)/length(data[data>0])
testresult5 = numeric(2500)
for (i in 1:2500){
   lognfit = rlnorm(length(data),mean=mu2,sd=sqrt(sigmasq2))   #randomly generate exponential
data using the parameters we found
   w3 = ks.test(lognfit,data)     #using KS test to see how good the fit is
   if (w3$p.value > 0.10){
     testresult5[i] = 1}
   if (w3$p.value <= 0.10){
     testresult5[i] = 0}
   }
sum(testresult5)
```

REFERENCES

- POWER-LAW DISTRIBUTIONS IN EMPIRICAL DATA, written by Aaron Clauset, Cosma Rohilla Shalizi, and M. E. J. Newman
- www.census.gov,:US Census Bureau, source of the data that were analyzed in this paper
- Helped with R code in random generator from the power law: Rick Wash, http://www.rickwash.com/papers/cscw08-appendix/powerlaw.R
- Helped design the R code procedures to fit the power law: http://tuvalu.santafe.edu/~aaronc/courses/7000/csci7000-001_2011_L3.pdf