Unveiling the geography of historical patents in the United States from 1836 to 1975 - Replication example

Sergio Petralia, Pierre-Alexandre Balland & David Rigby

June 16, 2016

Installing and loading Packages

```
install.packages('data.table')
install.packages('tm')
install.packages('stringi')
install.packages('plyr')
install.packages('RSNNS')
install.packages('neuralnet')
install.packages('kknn')
install.packages('plotly')
```

```
library(data.table)
library(tm)
library(stringi)
library(plyr)
library(RSNNS)
library(neuralnet)
library(kknn)
```

Load the data we will use in this tutorial

```
load("Example.RData")
```

```
ls()
```

```
 [1] "Assignees"                 "BTW"
 [3] "Cities2Search"             "City.Data"
 [5] "City.Matches"              "City.Names"
 [7] "City.State.Data"           "Cutoff"
 [9] "For"                       "Geography"
[11] "I"                         "ID"
[13] "Inventors"                 "Keywords"
[15] "KNN"                       "KNN.Results"
[17] "KNN.Table"                 "List"
[19] "NN"                        "NN.Results"
[21] "NN.Table"                  "OCR.Documents"
[23] "Patents"                   "Predicted.Geography"
[25] "Probit"                    "Probit.Results"
[27] "Probit.Table"              "Replications.NN"
```

```
[29] "Retrieved.States.Abbreviations" "Retrieved.States.Names"
[31] "Sample"                         "SampleTR"
[33] "SampleTS"                       "State.Data"
[35] "State.Matches"                  "State.Names"
[37] "States2Search"                  "StatesAbb2Search"
[39] "Threshold"                      "X0"
[41] "X0TR"                           "X0TS"
[43] "Y.Hat"                          "Y.Hat.KNN"
[45] "Y.Hat.NN"
```

Step 1 - Identifying candidate locations

The first step consists of identifying candidate locations within the text documents. Before that, it is convenient to clean up the documents. We will need to create variables describing the locations, and as these variables are created evaluating the candidate locations within the context of the document it is convinienent to reduce the size of the documents by trimming unnecesary words.

1.a Trimming Documents

The object "OCR.Documents" contain the documents we wil use throughout the example

```
length(OCR.Documents)
```

```
[1] 8481
```

```
# Let's take a look at the 10th entry
head (OCR.Documents[10])
```

```
[1] "PATENTED APR PROCESS SYNTHESIS DRYING OILS PET BONS ROLEUM HYDRO ..."
```

This is a short version of the original patent. The original document can be found here: https://www.google.com/patents/US2345754. Evaluating possible locations in the original document will require to match candidate names with all words in the text, however, if we take advantage of the fact that locations are usually quoted containing with at least a capital letter, then we could disregard all the words that do not contain one. If we want to keep important words that often occur close to geographical locations we can first capitalize them. For a great tutorial on text mining see: https://cran.r-project.org/web/packages/tm/vignettes/tm.pdf. Notice that we removed numbers and punctuation, and uppercased all words.

1.b Preparing the city and state names to look for

We need a list of US city names to look through the documents

```
head(City.Names)
```

```
          City State  FIPS   County
1       GREENE    OH 39155 TRUMBULL
2     CATSKILL    NY 36039   GREENE
```

```
3     NEW YORK    NY 36061 NEW YORK
4        KINGS    NY 36047    KINGS
5 KINGS COUNTY    NY 36047    KINGS
6        AKRON    OH 39153   SUMMIT
```

We create a character vector with a list of names to search over

```
Cities2Search=as.character(unique(City.Names$City))
Cities2Search=paste(' ',Cities2Search,' ',sep='')
```

We add a space after and before to make sure that we don't find words that are substrings of other words. We do the same for state names:

```
States2Search=as.character(unique(State.Names$Name))
States2Search=paste(' ',States2Search,' ',sep='')

StatesAbb2Search=as.character(unique(State.Names$Abbreviation))
```

Note: States are often quoted using the US postal code abbreviation of that time.

1.c Look for state names within the documents

```
Retrieved.States.Names=lapply(OCR.Documents,function(y)
  States2Search[stri_detect_fixed(y,States2Search)])
Retrieved.States.Abbreviations=lapply(OCR.Documents,function(y)
  StatesAbb2Search[stri_detect_fixed(y,StatesAbb2Search)])
```

Let's take a look

```
head(Retrieved.States.Names)
```

```
$US3693398
character(0)

$US3399321
character(0)

$US3413841
character(0)

$US1977249
character(0)

$US120714
[1] "MS"

$US1451635
[1] "WA"
```

You might want to create a Flag to differentiate whether the state was found with a complete name or just an abbreviation. Here we won't do that.

1.c Look for state names within the documents

```
Retrieved.States.Names = lapply(Retrieved.States.Names,
 function(y) as.character(State.Names$State.Code[match(gsub("^\\s+|\\s+$", "", y),
 State.Names$Name)]))
Retrieved.States.Abbreviations=lapply(Retrieved.States.Abbreviations,
 function(y) as.character(State.Names$State.Code[match(y,State.Names$Abbreviation)]))

State.Matches=mapply(c, Retrieved.States.Names, Retrieved.States.Abbreviations,
                     SIMPLIFY=FALSE)
```

```
head(State.Matches)
```

```
$US3693398
character(0)

$US3399321
character(0)

$US3413841
character(0)

$US1977249
character(0)

$US120714
[1] "MS"

$US1451635
[1] "WA"
```

Note: Dont' forget to take the spaces out

1.d Look for city names within the documents (It may take several minutes)

```
City.Matches=lapply(OCR.Documents,
 function(y) Cities2Search[stri_detect_fixed(y,Cities2Search,case_insensitive=F)] )
```

```
head(City.Matches)
```

```
$US3693398
[1] " DENMARK " " LARSON "   " METHOD "   " SIDES "    " SIGURD "

$US3399321
[1] " LAWRENCE "   " BUTLER "     " COLLINS "    " JAMES "      " HENDERSON "
[6] " METAL "      " JULY "       " EARTH "
```

```
$US3413841
 [1] " JOHNSON "     " PRESTON "     " HENSLEY "     " SWITZERLAND "
 [5] " SWISHER "     " STATION "     " WEBER "       " SIXTEEN "
 [9] " JEFFREY "     " MCCOY "       " METHOD "      " SHORTLY "
[13] " PLASTIC "     " WALLS "       " BLACK "       " SMALL "
[17] " NOLTON "

$US1977249
[1] " VIEW "   " ROBERT "

$US120714
[1] " MONROE "      " MISSISSIPPI " " ABERDEEN "    " WATER "
[5] " CLEMMONS "

$US1451635
[1] " WASHINGTON "   " AUGUST "       " ABERDEEN "     " THOMAS "
[5] " HARBOR "       " JOSEPH "       " GRAYS HARBOR "
```

There are many false positives detected. To evaluate candidate locations we will use information about the inventor/assignee names whenever it is available.

```
head(Inventors)
```

```
     Publication.number              Inventors.E                Inventors
6257          US3693398 JOHANNES SIGURD PEDERSEN PEDERSEN JOHANNES SIGURD
5588          US3399321              KAUER ERHARD             ERHARD KAUER
5626          US3413841                WEBER MAX                MAX WEBER
2381          US1977249        SPRAGINS ROBERT L        ROBERT L SPRAGINS
540            US120714                       NA       ALVEUS J CLEMMTOTS
1105          US1451635            THOMAS JOSEPH            JOSEPH THOMAS
```

```
head(Assignees)
```

```
  Publication.number               Original.Assignee.E
1          US3693398          AARHUS METALEMBALLAGE IND
2          US3399321 NORTH AMERICAN PHILIPS COMPANY INC
3          US3413841                         ELCALOR AG
4          US1977249                  SPRAGINS ROBERT L
5           US120714                                NA
6          US1451635                     JOSEPH THOMAS
          Original.Assignee
1 AARHUS METALEMBALLAGE IND
2              PHILIPS CORP
3                ELCALOR AG
4         ROBERT L SPRAGINS
5                     <NA>
6             JOSEPH THOMAS
```

This information is available at the Espacenet website, USPTO website, and on google patents. See: https://www.google.com/patents/US3693398, http://patft.uspto.gov/netacgi/nph-Parser?Sect2=PTO1&Sect2=HITOFF&p=1&u=/netahtml/PTO/search-bool.html&r=1&f=G&l=50&d=PALL&RefSrch=yes&Query=PN/3693398 and: http://worldwide.espacenet.com/publicationDetails/biblio?CC=US&NR=3693398A&KC=A&FT=D

1.e We now create a database with characteristics of each located state name

We idenitify the documents with positive entries

```
I=sapply(State.Matches,length)!=0
```

```
summary(I)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
      1     139     281     279     417     557     451
```

We create a list containing all the information for each documents (More appropriate in R)

```
List=mapply(function(x,y,r,z,t) list(x,y,r,z,t),OCR.Documents[I],State.Matches[I],
            paste(Assignees$Original.Assignee[I],Assignees$Original.Assignee.E[I],
            sep=','), paste(Inventors$Inventors[I],Inventors$Inventors.E[I],sep=','),
            Patents$Publication.number[I],SIMPLIFY = FALSE)
```

```
unlist(List[[1]])
```

```
[1] "DESCRIPTIONÂ Ã OCR ERRORS\nUNITED NSTATES ALVEUS CLEMMONS ABERDEEN MISSISSIPPI \nIMPROVEMENT WATERI
[2] "MS"
[3] "NA,NA"
[4] "ALVEUS J CLEMMTOTS,NA"
[5] "US120714"
```

Some useful Keywords to associate with the geographical locations

```
Keywords=c('COMPANY','CORPORATION','ASSIGNOR','INVENTOR')
Cutoff=c(' FIG ',' FIGURE ','REFERENCES CITED')
```

The following function evaluates characteristics of each geographical location. It is an attempt to provide a fast but at the same time "readable" code. It could be made more efficient.

```
State.Data=lapply(List,function(z) {

  # Each (z) corresponds to a document, where the first element is the description, the
  # second the detected state, and so on.


  # How many times the state was found?
  Data=data.frame(count(stri_trans_toupper(z[[2]])),stringsAsFactors=F)
  colnames(Data)=c('State','Frequency')
  Data$State=as.character(Data$State)

  # Finding the state names
  temp=lapply(Data$State,function(y) State.Names[stri_detect_fixed
      (State.Names$State.Code,y),1:2])
```

```
# For each State we retrieve the position in the document, it can be
# found several times in the text
Location.S=lapply(temp,function(y) unlist(stri_locate_all_fixed(z[[1]],
  as.character(y),omit_no_match = T)))
Min.Location=sapply(Location.S,function(y) min(y,na.rm=T))
Max.Location=sapply(Location.S,function(y) max(y,na.rm=T))
Mean.Location=sapply(Location.S,function(y) mean(y,na.rm=T))
Median.Location=sapply(Location.S,function(y) median(y,na.rm=T))

# Find the Location of important words
Location=unlist(lapply(Keywords,function(y) unlist(stri_locate_all_fixed(z[[1]],
 y,omit_no_match = T))))
Min.Keywords=min(Location,na.rm=T)
Max.Keywords=max(Location,na.rm=T)
Mean.Keywords=mean(Location,na.rm=T)
Median.Keywords=median(Location,na.rm=T)

# Find the Location of important words
Location=unlist(lapply(Cutoff,function(y) unlist(stri_locate_all_fixed(z[[1]],
 y,omit_no_match = T))))
Min.Cutoff=min(Location,na.rm=T)

# Finding The Assignee location (first clean it )
Assig=sapply(stri_split_fixed(z[[3]],','),function (x) gsub("^\\s+|\\s+$", "",
 as.character(stripWhitespace(removeWords(x,c(stri_trans_toupper(letters)))))))
 Assig=stripWhitespace(stri_replace_all_fixed(Assig, "[[:punct:]]", ""))

Location=unlist(lapply(Assig,function(y) unlist(stri_locate_all_fixed(z[[1]],
 y,omit_no_match = T))))
Min.Assignees=min(Location,na.rm=T)
Max.Assignees=max(Location,na.rm=T)
Mean.Assignees=mean(Location,na.rm=T)
Median.Assignees=median(Location,na.rm=T)

# Finding The Inventor location
# First clean it

Inv=sapply(stri_split_fixed(z[[4]],','),function (x) gsub("^\\s+|\\s+$", "",
 as.character(stripWhitespace(removeWords(x,c(stri_trans_toupper(letters),letters))))))
Inv=stripWhitespace(stri_replace_all_fixed(Inv, "[[:punct:]]", ""))

Location=unlist(lapply(Inv,function(y) unlist(stri_locate_all_fixed(z[[1]],
  y,omit_no_match = T))))
Min.Inv=min(Location,na.rm=T)
Max.Inv=max(Location,na.rm=T)
Mean.Inv=mean(Location,na.rm=T)
Median.Inv=median(Location,na.rm=T)

# Now lets look for some important words, like County, resident, etc.
# And evaluate their distance  with respect to the located state name.
State.Locations=lapply(Location.S,function(y) outer(y,
 rowMeans(stri_locate_all_fixed(z[[1]],' STATE ',omit_no_match = T)[[1]]),FUN='-'))
County.Locations=lapply(Location.S,function(y) outer(y,
```

```
   rowMeans(stri_locate_all_fixed(z[[1]],' COUNTY ',omit_no_match = T)[[1]]),FUN='-'))
  City.Locations=lapply(Location.S,function(y) outer(y,
   rowMeans(stri_locate_all_fixed(z[[1]],' CITY ',omit_no_match = T)[[1]]),FUN='-'))
  Subject.Locations=lapply(Location.S,function(y) outer(y,
   rowMeans(stri_locate_all_fixed(z[[1]],'SUBJECT',omit_no_match = T)[[1]]),FUN='-'))
  Resident.Locations=lapply(Location.S,function(y) outer(y,
   rowMeans(stri_locate_all_fixed(z[[1]],c('RESIDENT','RESIDING'),
   omit_no_match = T)[[1]]),FUN='-'))
  BIKT.Locations=lapply(Location.S,function(y) outer(y,
   rowMeans(stri_locate_all_fixed(z[[1]],'BE IT KNOWN THAT',
   omit_no_match = T)[[1]]),FUN='-'))

  R.of.State=unlist(lapply(State.Locations,function(y) min(y[y>0])))
  L.of.State=unlist(lapply(State.Locations,function(y) max(y[y<0])))
  R.of.County=unlist(lapply(County.Locations,function(y) min(y[y>0])))
  L.of.County=unlist(lapply(County.Locations,function(y) max(y[y<0])))
  R.of.City=unlist(lapply(City.Locations,function(y) min(y[y>0])))
  L.of.City=unlist(lapply(City.Locations,function(y) max(y[y<0])))
  R.of.Res=unlist(lapply(Resident.Locations,function(y) min(y[y>0])))
  L.of.Res=unlist(lapply(Resident.Locations,function(y) max(y[y<0])))
  R.of.BIKT=unlist(lapply(BIKT.Locations,function(y) min(y[y>0])))
  L.of.BIKT=unlist(lapply(BIKT.Locations,function(y) max(y[y<0])))

  # Compile
  Data=data.frame(Data,Publication.number=z[[5]],Min.Cutoff,
                  L.of.BIKT,R.of.BIKT,R.of.State,L.of.State,
                  R.of.County,L.of.County,R.of.City,L.of.City,
                  L.of.Res,R.of.Res,Min.Location,Max.Location,
                  Mean.Location,Median.Location,Min.Keywords,
                  Max.Keywords,Mean.Keywords,Median.Keywords,
                  Min.Inv,Max.Inv,Mean.Inv,Median.Inv,
                  Min.Assignees,Max.Assignees,Mean.Assignees,
                  Median.Assignees)

  return(Data)

})

State.Data=rbindlist(State.Data)
```

```
head(State.Data)
```

```
  State Frequency.State Publication.number Min.Cutoff.State
1    MS               1           US120714              226
2    WA               1          US1451635              424
3    KS               1           US957500              497
4    WA               1           US957500              497
5    KS               1           US189636              289
6    WA               1           US189636              289
  L.of.BIKT.State R.of.BIKT.State R.of.State.State L.of.State.State
1            -Inf             Inf               Inf             -Inf
2            -Inf             Inf               Inf             -126
3            -Inf             Inf                 4             -161
```

```
4            -Inf             Inf              Inf             -276
5            -Inf             Inf                4             -112
6            -Inf             Inf              Inf             -168
  R.of.County.State L.of.County.State R.of.City.State L.of.City.State
1               4.5            -109.5            26.5           -87.5
2               Inf            -106.5             Inf            -Inf
3              20.5            -144.5             Inf            -Inf
4               Inf            -259.5             Inf            -Inf
5              20.5             -95.5             Inf            -Inf
6               Inf            -151.5             Inf            -Inf
  L.of.Res.State R.of.Res.State Min.Location.State Max.Location.State
1           -Inf            Inf                 67                201
2          -89.5            Inf                223                232
3           -Inf            Inf                246                421
4           -Inf            Inf                127                136
5           -Inf            Inf                144                270
6           -Inf            Inf                 84                 93
  Mean.Location.State Median.Location.State Min.Keywords.State
1               134.0                 134.0                Inf
2               227.5                 227.5                414
3               333.5                 333.5                474
4               131.5                 131.5                474
5               207.0                 207.0                Inf
6                88.5                  88.5                Inf
  Max.Keywords.State Mean.Keywords.State Median.Keywords.State
1               -Inf                 NaN                    NA
2                420               417.0                 417.0
3                493               483.5                 483.5
4                493               483.5                 483.5
5               -Inf                 NaN                    NA
6               -Inf                 NaN                    NA
  Min.Inv.State Max.Inv.State Mean.Inv.State Median.Inv.State
1           Inf          -Inf            NaN               NA
2           200           604            402              402
3           219           639            429              429
4           219           639            429              429
5           Inf          -Inf            NaN               NA
6           Inf          -Inf            NaN               NA
  Min.Assignees.State Max.Assignees.State Mean.Assignees.State
1                 Inf                -Inf                  NaN
2                 200                 604                  402
3                 219                 639                  429
4                 219                 639                  429
5                 Inf                -Inf                  NaN
6                 Inf                -Inf                  NaN
  Median.Assignees.State
1                     NA
2                    402
3                    429
4                    429
5                     NA
6                     NA
```

1.f Do the same for city names

```
head(City.Data)
```

```
     City Freq Length Publication.number Min.Cutoff Min.Countries
1    DATA    1    894          US3693398        423           Inf
2  LARSON    1    894          US3693398        423           Inf
3  METHOD    2    894          US3693398        423           Inf
4   SIDES    2    894          US3693398        423           Inf
5  SIGURD    1    894          US3693398        423           Inf
6 DENMARK    2    894          US3693398        423           Inf
  Country.Name Discard Detected.Name Date Month R.of.Corporation
1            0       0             0    0     0              Inf
2            0       0             0    0     0              Inf
3            0       0             0    0     0              Inf
4            0       0             0    0     0              Inf
5            0       1             1    0     0              Inf
6            1       0             0    0     0              Inf
  L.of.STREET L.of.BIKT R.of.BIKT R.of.State L.of.State R.of.County
1       -44.5      -Inf       Inf        Inf       -Inf         Inf
2       -14.5      -Inf       Inf        Inf       -Inf         Inf
3       -49.5      -Inf       Inf        Inf       -Inf         Inf
4       -19.5      -Inf       Inf        Inf       -Inf         Inf
5       -63.5      -Inf       Inf        Inf       -Inf         Inf
6       -37.5      -Inf       Inf        Inf       -Inf         Inf
  L.of.County R.of.City L.of.City L.of.Res R.of.Res R.of.Columbia
1        -Inf       Inf      -Inf     -Inf      Inf           Inf
2        -Inf       Inf      -Inf     -Inf      Inf           Inf
3        -Inf       Inf      -Inf     -Inf      Inf           Inf
4        -Inf       Inf      -Inf     -Inf      Inf           Inf
5        -Inf       Inf      -Inf     -Inf      Inf           Inf
6        -Inf       Inf      -Inf     -Inf      Inf           Inf
  L.of.Columbia R.of.Washington L.of.Washington R.of.Planograph
1          -Inf             Inf            -Inf             Inf
2          -Inf             Inf            -Inf             Inf
3          -Inf             Inf            -Inf             Inf
4          -Inf             Inf            -Inf             Inf
5          -Inf             Inf            -Inf             Inf
6          -Inf             Inf            -Inf             Inf
  L.of.Planograph Name.Similarity Min.Location Max.Location Mean.Location
1            -Inf               4          251          254         252.5
2            -Inf               4          279          284         281.5
3            -Inf               5           62          444         253.0
4            -Inf               4           93          474         283.5
5            -Inf               0          118          123         120.5
6            -Inf               6          143          205         174.0
  Median.Location Min.Keywords Max.Keywords Mean.Keywords Median.Keywords
1           252.5          Inf         -Inf           NaN              NA
2           281.5          Inf         -Inf           NaN              NA
3           253.0          Inf         -Inf           NaN              NA
4           283.5          Inf         -Inf           NaN              NA
5           120.5          Inf         -Inf           NaN              NA
```

```
6           174.0          Inf         -Inf           NaN              NA
  Min.Inv Max.Inv Mean.Inv Median.Inv Min.Assignees Max.Assignees
1     109     132    120.5      120.5           160           184
2     109     132    120.5      120.5           160           184
3     109     132    120.5      120.5           160           184
4     109     132    120.5      120.5           160           184
5     109     132    120.5      120.5           160           184
6     109     132    120.5      120.5           160           184
  Mean.Assignees Median.Assignees
1            172              172
2            172              172
3            172              172
4            172              172
5            172              172
6            172              172
```

1.g Now merge the data

```
City.State.Data=merge(data.frame(City.Data),
                      data.frame(unique(City.Names[,c(1:2)])),
                      by='City',allow.cartesian=TRUE)
colnames(State.Data)[-c(1,3)]=paste(colnames(State.Data)[-c(1,3)],'State',sep = '.')
City.State.Data=merge(City.State.Data,State.Data,by=c('Publication.number','State'),
  all.x=T)
```

```
head(City.State.Data)
```

```
  Publication.number State    City Freq Length Min.Cutoff Min.Countries
1          US3693398    CA  LARSON    1    894        423           Inf
2          US3693398    CA   SIDES    2    894        423           Inf
3          US3693398    IA DENMARK    2    894        423           Inf
4          US3693398    IL DENMARK    2    894        423           Inf
5          US3693398    KS DENMARK    2    894        423           Inf
6          US3693398    ME DENMARK    2    894        423           Inf
  Country.Name Discard Detected.Name Date Month R.of.Corporation
1            0       0             0    0     0              Inf
2            0       0             0    0     0              Inf
3            1       0             0    0     0              Inf
4            1       0             0    0     0              Inf
5            1       0             0    0     0              Inf
6            1       0             0    0     0              Inf
  L.of.STREET L.of.BIKT R.of.BIKT R.of.State L.of.State R.of.County
1       -14.5      -Inf       Inf        Inf       -Inf         Inf
2       -19.5      -Inf       Inf        Inf       -Inf         Inf
3       -37.5      -Inf       Inf        Inf       -Inf         Inf
4       -37.5      -Inf       Inf        Inf       -Inf         Inf
5       -37.5      -Inf       Inf        Inf       -Inf         Inf
6       -37.5      -Inf       Inf        Inf       -Inf         Inf
  L.of.County R.of.City L.of.City L.of.Res R.of.Res R.of.Columbia
1        -Inf       Inf      -Inf     -Inf      Inf           Inf
2        -Inf       Inf      -Inf     -Inf      Inf           Inf
```

```
3        -Inf       Inf      -Inf     -Inf      Inf           Inf
4        -Inf       Inf      -Inf     -Inf      Inf           Inf
5        -Inf       Inf      -Inf     -Inf      Inf           Inf
6        -Inf       Inf      -Inf     -Inf      Inf           Inf
  L.of.Columbia R.of.Washington L.of.Washington R.of.Planograph
1          -Inf             Inf            -Inf             Inf
2          -Inf             Inf            -Inf             Inf
3          -Inf             Inf            -Inf             Inf
4          -Inf             Inf            -Inf             Inf
5          -Inf             Inf            -Inf             Inf
6          -Inf             Inf            -Inf             Inf
  L.of.Planograph Name.Similarity Min.Location Max.Location Mean.Location
1            -Inf               4          279          284         281.5
2            -Inf               4           93          474         283.5
3            -Inf               6          143          205         174.0
4            -Inf               6          143          205         174.0
5            -Inf               6          143          205         174.0
6            -Inf               6          143          205         174.0
  Median.Location Min.Keywords Max.Keywords Mean.Keywords Median.Keywords
1           281.5          Inf         -Inf          NaN             NA
2           283.5          Inf         -Inf          NaN             NA
3           174.0          Inf         -Inf          NaN             NA
4           174.0          Inf         -Inf          NaN             NA
5           174.0          Inf         -Inf          NaN             NA
6           174.0          Inf         -Inf          NaN             NA
  Min.Inv Max.Inv Mean.Inv Median.Inv Min.Assignees Max.Assignees
1     109     132    120.5      120.5           160           184
2     109     132    120.5      120.5           160           184
3     109     132    120.5      120.5           160           184
4     109     132    120.5      120.5           160           184
5     109     132    120.5      120.5           160           184
6     109     132    120.5      120.5           160           184
  Mean.Assignees Median.Assignees Frequency.State Min.Cutoff.State
1            172              172              NA               NA
2            172              172              NA               NA
3            172              172              NA               NA
4            172              172              NA               NA
5            172              172              NA               NA
6            172              172              NA               NA
  L.of.BIKT.State R.of.BIKT.State R.of.State.State L.of.State.State
1              NA              NA               NA               NA
2              NA              NA               NA               NA
3              NA              NA               NA               NA
4              NA              NA               NA               NA
5              NA              NA               NA               NA
6              NA              NA               NA               NA
  R.of.County.State L.of.County.State R.of.City.State L.of.City.State
1                NA                NA              NA              NA
2                NA                NA              NA              NA
3                NA                NA              NA              NA
4                NA                NA              NA              NA
5                NA                NA              NA              NA
6                NA                NA              NA              NA
  L.of.Res.State R.of.Res.State Min.Location.State Max.Location.State
```

```
1             NA             NA                 NA                 NA
2             NA             NA                 NA                 NA
3             NA             NA                 NA                 NA
4             NA             NA                 NA                 NA
5             NA             NA                 NA                 NA
6             NA             NA                 NA                 NA
  Mean.Location.State Median.Location.State Min.Keywords.State
1                  NA                    NA                 NA
2                  NA                    NA                 NA
3                  NA                    NA                 NA
4                  NA                    NA                 NA
5                  NA                    NA                 NA
6                  NA                    NA                 NA
  Max.Keywords.State Mean.Keywords.State Median.Keywords.State
1                 NA                  NA                    NA
2                 NA                  NA                    NA
3                 NA                  NA                    NA
4                 NA                  NA                    NA
5                 NA                  NA                    NA
6                 NA                  NA                    NA
  Min.Inv.State Max.Inv.State Mean.Inv.State Median.Inv.State
1            NA            NA             NA               NA
2            NA            NA             NA               NA
3            NA            NA             NA               NA
4            NA            NA             NA               NA
5            NA            NA             NA               NA
6            NA            NA             NA               NA
  Min.Assignees.State Max.Assignees.State Mean.Assignees.State
1                  NA                  NA                   NA
2                  NA                  NA                   NA
3                  NA                  NA                   NA
4                  NA                  NA                   NA
5                  NA                  NA                   NA
6                  NA                  NA                   NA
  Median.Assignees.State
1                     NA
2                     NA
3                     NA
4                     NA
5                     NA
6                     NA
```

Step 2 - Identify the correct locations

2.a Match the data with the collected sample

We can now match the data with the collected sample and create a flag identifying the correct locations

```
head(Sample)
```

```
  Publication.number    City State Freq Length Min.Cutoff Min.Countries
```

```
1          US3693398 DENMARK    IA    2    894        423           Inf
2          US3693398 DENMARK    IL    2    894        423           Inf
3          US3693398 DENMARK    KS    2    894        423           Inf
4          US3693398 DENMARK    ME    2    894        423           Inf
5          US3693398 DENMARK    NY    2    894        423           Inf
6          US3693398 DENMARK    SC    2    894        423           Inf
  Country.Name Discard Detected.Name Date Month R.of.Corporation
1            1       0             0    0     0              Inf
2            1       0             0    0     0              Inf
3            1       0             0    0     0              Inf
4            1       0             0    0     0              Inf
5            1       0             0    0     0              Inf
6            1       0             0    0     0              Inf
  L.of.STREET L.of.BIKT R.of.BIKT R.of.State L.of.State R.of.County
1       -37.5      -Inf       Inf        Inf       -Inf         Inf
2       -37.5      -Inf       Inf        Inf       -Inf         Inf
3       -37.5      -Inf       Inf        Inf       -Inf         Inf
4       -37.5      -Inf       Inf        Inf       -Inf         Inf
5       -37.5      -Inf       Inf        Inf       -Inf         Inf
6       -37.5      -Inf       Inf        Inf       -Inf         Inf
  L.of.County R.of.City L.of.City L.of.Res R.of.Res R.of.Columbia
1        -Inf       Inf      -Inf     -Inf      Inf           Inf
2        -Inf       Inf      -Inf     -Inf      Inf           Inf
3        -Inf       Inf      -Inf     -Inf      Inf           Inf
4        -Inf       Inf      -Inf     -Inf      Inf           Inf
5        -Inf       Inf      -Inf     -Inf      Inf           Inf
6        -Inf       Inf      -Inf     -Inf      Inf           Inf
  L.of.Columbia R.of.Washington L.of.Washington R.of.Planograph
1          -Inf             Inf            -Inf             Inf
2          -Inf             Inf            -Inf             Inf
3          -Inf             Inf            -Inf             Inf
4          -Inf             Inf            -Inf             Inf
5          -Inf             Inf            -Inf             Inf
6          -Inf             Inf            -Inf             Inf
  L.of.Planograph Name.Similarity Min.Location Max.Location Mean.Location
1            -Inf               6          143          205           174
2            -Inf               6          143          205           174
3            -Inf               6          143          205           174
4            -Inf               6          143          205           174
5            -Inf               6          143          205           174
6            -Inf               6          143          205           174
  Median.Location Min.Keywords Max.Keywords Mean.Keywords Median.Keywords
1             174          Inf         -Inf          NaN             NA
2             174          Inf         -Inf          NaN             NA
3             174          Inf         -Inf          NaN             NA
4             174          Inf         -Inf          NaN             NA
5             174          Inf         -Inf          NaN             NA
6             174          Inf         -Inf          NaN             NA
  Min.Inv Max.Inv Mean.Inv Median.Inv Min.Assignees Max.Assignees
1     109     132    120.5      120.5           160           184
2     109     132    120.5      120.5           160           184
3     109     132    120.5      120.5           160           184
4     109     132    120.5      120.5           160           184
5     109     132    120.5      120.5           160           184
```

```
6     109     132    120.5      120.5           160           184
  Mean.Assignees Median.Assignees Frequency.State Min.Cutoff.State
1            172              172              NA               NA
2            172              172              NA               NA
3            172              172              NA               NA
4            172              172              NA               NA
5            172              172              NA               NA
6            172              172              NA               NA
  L.of.BIKT.State R.of.BIKT.State R.of.State.State L.of.State.State
1              NA              NA               NA               NA
2              NA              NA               NA               NA
3              NA              NA               NA               NA
4              NA              NA               NA               NA
5              NA              NA               NA               NA
6              NA              NA               NA               NA
  R.of.County.State L.of.County.State R.of.City.State L.of.City.State
1                NA                NA              NA               NA
2                NA                NA              NA               NA
3                NA                NA              NA               NA
4                NA                NA              NA               NA
5                NA                NA              NA               NA
6                NA                NA              NA               NA
  L.of.Res.State R.of.Res.State Min.Location.State Max.Location.State
1             NA             NA                 NA                 NA
2             NA             NA                 NA                 NA
3             NA             NA                 NA                 NA
4             NA             NA                 NA                 NA
5             NA             NA                 NA                 NA
6             NA             NA                 NA                 NA
  Mean.Location.State Median.Location.State Min.Keywords.State
1                  NA                    NA                 NA
2                  NA                    NA                 NA
3                  NA                    NA                 NA
4                  NA                    NA                 NA
5                  NA                    NA                 NA
6                  NA                    NA                 NA
  Max.Keywords.State Mean.Keywords.State Median.Keywords.State
1                 NA                  NA                    NA
2                 NA                  NA                    NA
3                 NA                  NA                    NA
4                 NA                  NA                    NA
5                 NA                  NA                    NA
6                 NA                  NA                    NA
  Min.Inv.State Max.Inv.State Mean.Inv.State Median.Inv.State
1            NA            NA             NA               NA
2            NA            NA             NA               NA
3            NA            NA             NA               NA
4            NA            NA             NA               NA
5            NA            NA             NA               NA
6            NA            NA             NA               NA
  Min.Assignees.State Max.Assignees.State Mean.Assignees.State
1                  NA                  NA                   NA
2                  NA                  NA                   NA
3                  NA                  NA                   NA
```

```
4                   NA                   NA                   NA
5                   NA                   NA                   NA
6                   NA                   NA                   NA
  Median.Assignees.State Correct Year SUBSTRING IS.WASH.ALSO IS.COL.ALSO
1                     NA       0 1972         0            0           0
2                     NA       0 1972         0            0           0
3                     NA       0 1972         0            0           0
4                     NA       0 1972         0            0           0
5                     NA       0 1972         0            0           0
6                     NA       0 1972         0            0           0
  COC
1   0
2   0
3   0
4   0
5   0
6   0
```

At this stage we added some other useful variables. For instance, the variable COC identifies whenever a particular location name corresponded to the same county for another location within the same patent. Therefore it measures the co-ocurrence of a county by two different location names in a patent.

2.b Create a function to use in the estimation

We create a function to use in the estimation. It will be handy so we dont have to create new variables when estimating different specifications

```
BTW=function(x,l,u) {y=1*I(as.numeric(x)>l & as.numeric(x)<u)
y[is.na(y)]=0
return(y)
}
```

It returns a number 1, for any vector x, whenever the value of an entry is between a lower value (l) and an unpper value (u), and zero otherwise.

2.c Divide the sample into training and an evaluation set, by patent number

```
set.seed(1)
ID <- sample(1:2, length(unique(Sample$Publication.number)), replace = TRUE)
ID=ID[match(Sample$Publication.number,unique(Sample$Publication.number))]

SampleTR=Sample[ID==1,]
SampleTS=Sample[ID==2,]
```

2.d Estimating a Probit Model

```
Probit=glm(Correct~poly(Min.Location,5,raw=T)+BTW(L.of.STREET,-2,0)
           +I(BTW(Min.Location-Min.Location.State,50,10000) |
            BTW(Min.Location.State-Min.Location,50,10000))+I(Min.Countries!='Inf')
           +BTW(Min.Countries-Min.Location,0,15)
           +I(BTW(Min.Location.State-Min.Cutoff,0,10000)==1 &
            I(Min.Location.State/Length)>0.5)
           +I(BTW(Min.Location-Min.Cutoff,0,10000)==1 & I(Min.Location/Length)>0.5)
           +COC+BTW(R.of.Corporation,0,10)+BTW(R.of.State,0,10)
           + Country.Name+SUBSTRING+factor(nchar(City))
           +BTW(Min.Location.State-Min.Location,0,10)
           +BTW(Min.Location-Min.Location.State,0,50)
           +I(BTW(Min.Location.State-Min.Location,0,10) & I(Min.Location/Length)>0.5)
           +I(BTW(Min.Location-Min.Location.State,0,50) & I(Min.Location/Length)>0.5)
           +Detected.Name+Discard+is.na(Min.Location.State)
           +poly(Min.Location/Length,5,raw=T)
           +as.numeric(Max.Location/Length)+BTW(R.of.City,0,15)
           +BTW(R.of.County,0,15)+I(BTW(R.of.City,0,15) | BTW(R.of.County,0,15))
           + BTW(Min.Location-Min.Keywords,-40,40)
           + I(I((is.na(Min.Location.State)))
           & BTW(Min.Location-Min.Keywords,-40,40)) ,data=SampleTR,
           family = binomial(link = "probit"))
Y.Hat= predict(Probit,newdata=SampleTS,type='response')
```

It proves to be useful to include covariates of competing candidate locations within a patent documents (as in a spatial econometrics framework).

```
# Assesment
Probit.Table=table(SampleTS$Correct,1*(Y.Hat>0.5))
colnames(Probit.Table)=c('Predicted as Incorrect (Y.Hat<0.5)',
                         'Predicted as Correct (Y.Hat>0.5)')
row.names(Probit.Table)=c('Incorrect','Correct')
```

```
Probit.Table
```

```
           Predicted as Incorrect (Y.Hat<0.5)
  Incorrect                             175572
  Correct                                  862

           Predicted as Correct (Y.Hat>0.5)
  Incorrect                               654
  Correct                                3904
```

2.d Lets normalize the data for the KNN and NN procedures (It may take several minutes)

```
X0=model.frame(Probit$formula,SampleTR)
X0=as.matrix(X0)
X0[X0=="FALSE"| X0==" FALSE"]='0'
X0[X0=="TRUE" | X0==" TRUE"]='1'
X0=apply(X0,2,function(y) as.numeric(y))
X0=normalizeData(X0, type = "0_1")
X0TR=data.frame(X0)

X0=model.frame(Probit$formula,SampleTS)
X0=as.matrix(X0)
X0[X0=="FALSE"| X0==" FALSE"]='0'
X0[X0=="TRUE" | X0==" TRUE"]='1'
X0=apply(X0,2,function(y) as.numeric(y))
X0=normalizeData(X0, type = "0_1")
X0TS=data.frame(X0)
```

2.e Estimating a KNN Model (It may take several minutes)

```
# Note: the results are already in the environment
For=paste(colnames(X0TR[,-1]),collapse='+')
KNN <- kknn(paste('X1~',For,collapse=''),train=X0TR,test=X0TS,
            k=10,scale=F,kernel='epanechnikov')
Y.Hat.KNN=KNN[[1]]


# Assesment
KNN.Table=table(SampleTS$Correct,1*(Y.Hat.KNN>0.5))
colnames(KNN.Table)=c('Predicted as Incorrect (Y.Hat<0.5)',
                      'Predicted as Correct (Y.Hat>0.5)')
row.names(KNN.Table)=c('Incorrect','Correct')
```

```
KNN.Table
```

```
           Predicted as Incorrect (Y.Hat<0.5)
  Incorrect                             175465
  Correct                                  844

           Predicted as Correct (Y.Hat>0.5)
  Incorrect                              761
  Correct                               3922
```

2.f Estimating a NN Model (It may take several hours)

```
For=paste(colnames(X0TR[,-1]),collapse='+')

NN <- neuralnet(paste('X1~',For,collapse=''),data=X0TR,hidden=15,
```

```
                linear.output = F)
# Note: the results are already in the environment

Y.Hat.NN=compute(NN,covariate=X0TS[,-1])$net.result

# Assesment
NN.Table=table(SampleTS$Correct,1*(Y.Hat.NN>0.5))
colnames(NN.Table)=c('Predicted as Incorrect (Y.Hat<0.5)',
                     'Predicted as Correct (Y.Hat>0.5)')
row.names(NN.Table)=c('Incorrect','Correct')
```

```
NN.Table
```

```
           Predicted as Incorrect (Y.Hat<0.5)
  Incorrect                             175616
  Correct                                  971

           Predicted as Correct (Y.Hat>0.5)
  Incorrect                                610
  Correct                                 3795
```

Step 3 - Graphic Analysis

The results will slightly differ compared to the paper, as we are using a simplified specification.

3.a Create measures of Alpha and Coverage for each procedure

```
Threshold=seq(0.01,0.99,0.01)

# Probit Model
Probit.Results=lapply(Threshold,function(y){
  temp=table(SampleTS$Correct,1*(Y.Hat>y))
  Alpha=temp[1,2]/sum(temp[,2])
  Coverage=temp[2,2]/sum(temp[2,])
  return(data.frame(Threshold=y,Alpha=Alpha,Coverage=Coverage))
})
Probit.Results=rbindlist(Probit.Results)


# KNN Model
KNN.Results=lapply(Threshold,function(y){
  temp=table(SampleTS$Correct,1*(Y.Hat.KNN>y))
  Alpha=temp[1,2]/sum(temp[,2])
  Coverage=temp[2,2]/sum(temp[2,])
  return(data.frame(Threshold=y,Alpha=Alpha,Coverage=Coverage))
})
KNN.Results=rbindlist(KNN.Results)
```

```
# NN Model
NN.Results=lapply(Threshold,function(y){
  temp=table(SampleTS$Correct,1*(Y.Hat.NN>y))
  Alpha=temp[1,2]/sum(temp[,2])
  Coverage=temp[2,2]/sum(temp[2,])
  return(data.frame(Threshold=y,Alpha=Alpha,Coverage=Coverage))
})
NN.Results=rbindlist(NN.Results)
```

3.b Performance of Models

```
par(mfrow=c(1,3))

plot(Probit.Results$Threshold,Probit.Results$Coverage,
     main='Probit Model | Coverage and Alpha',ylab='',
     xlab='Threshold',type='l',lwd=3,bty='l',
     col='dodgerblue4',cex.main=1.5,cex.axis=1,cex.lab=1.5,ylim=c(0,1))
abline(h=0.05,col='black')
lines(Probit.Results$Threshold,Probit.Results$Alpha,
      ylab='',type='l',lwd=3,bty='l',col='red')
legend('topright',legend=c('Coverage','Alpha'),bty='n',lwd = 2,
       col = c('dodgerblue4','red'))

plot(KNN.Results$Threshold,KNN.Results$Coverage,
     main='KNN Model | Coverage and Alpha',ylab='',xlab='Threshold',
     type='l',lwd=3,bty='l',
     col='dodgerblue4',cex.main=1.5,cex.axis=1,cex.lab=1.5,ylim=c(0,1))
abline(h=0.05,col='black')
lines(KNN.Results$Threshold,KNN.Results$Alpha,ylab='',type='l',lwd=3,
      bty='l',col='red')
legend('topright',legend=c('Coverage','Alpha'),bty='n',lwd = 2,
       col = c('dodgerblue4','red'))

plot(NN.Results$Threshold,NN.Results$Coverage,
     main='NN Model | Coverage and Alpha',ylab='',xlab='Threshold',type='l',lwd=3,bty='l',
     col='dodgerblue4',
     cex.main=1.5,cex.axis=1,cex.lab=1.5,ylim=c(0,1))
abline(h=0.05,col='black')
lines(NN.Results$Threshold,NN.Results$Alpha,ylab='',type='l',lwd=3,
      bty='l',col='red')
legend('topright',legend=c('Coverage','Alpha'),bty='n',lwd = 2,
       col = c('dodgerblue4','red'))
```

3.c Performance comparison across models

```
par(mfrow=c(1,1))
plot(Probit.Results$Alpha,Probit.Results$Coverage,
     main='Comparison Across Models',ylab='Coverage',
```

```
     xlab='Alpha', type='l',lwd=3,bty='l',col='dodgerblue4',
     cex.main=1.5,cex.axis=1,cex.lab=1.5,ylim=c(0,1))
lines(KNN.Results$Alpha,KNN.Results$Coverage,ylab='',type='l',
      lwd=3,bty='l',col='red')
lines(NN.Results$Alpha,NN.Results$Coverage,ylab='',type='l',
      lwd=3,bty='l',col='dark green')
legend('bottomright',legend=c('Probit','KNN','NN'),bty='n',
       lwd = 3,
       col = c('dodgerblue4','red','dark green'))
```

3.d Geographical Distribution

```
library('plotly')

Geography=unique(subset(Sample,Correct==1)[,1:3])
Geography=merge(Geography,City.Names,by=c('City','State'))
Geography$Place=paste(Geography$County,Geography$State,sep = ' | ')
Geography=count(Geography,'Place')
Geography=Geography[order(Geography$freq,decreasing = T),]

Predicted.Geography=unique(subset(SampleTS,Y.Hat>0.75)[,1:3])
Predicted.Geography=merge(Predicted.Geography,City.Names,
                          by=c('City','State'))
Predicted.Geography$Place=paste(Predicted.Geography$County,
                                Predicted.Geography$State,sep = ' | ')
Predicted.Geography=count(Predicted.Geography,'Place')
I=match(Geography$Place,Predicted.Geography$Place)
Predicted.Geography=Predicted.Geography[I,]
Predicted.Geography$freq[is.na(Predicted.Geography$freq)]=0
Predicted.Geography$Place=Geography$Place
```

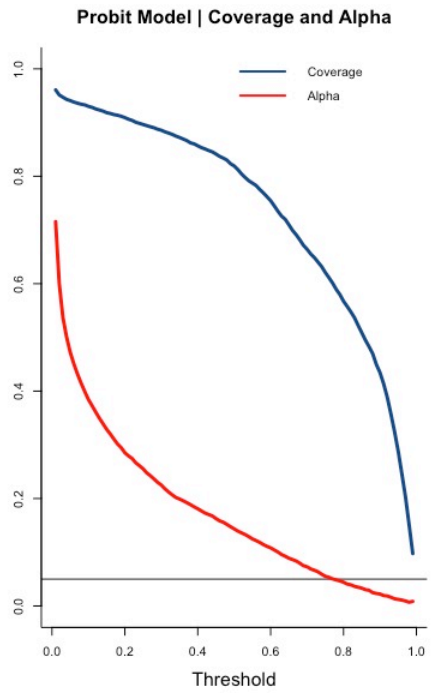
Plot with all the Values

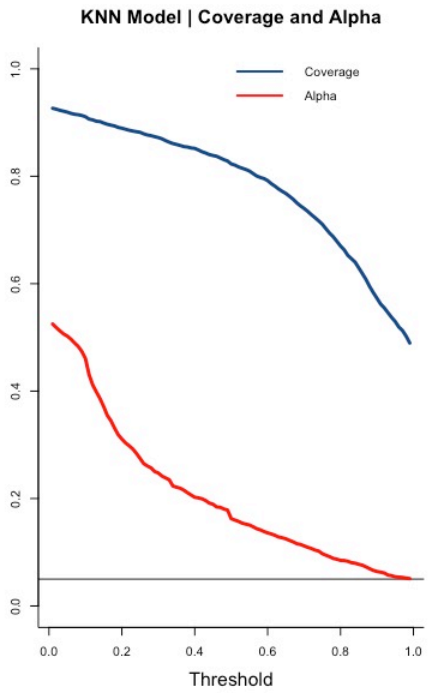
```
plot_ly(x=Geography$Place , y = Geography$freq/sum(Geography$freq),
        type = "bar",opacity = 0.9,name='Sample')  %>%
  add_trace(x=Predicted.Geography$Place,
            y = Predicted.Geography$freq/sum(Predicted.Geography$freq),
            type = "bar",opacity = 0.6,name='Predicted') %>%
  layout(barmode='overlay', showlegend = T,xaxis = list(title=' ',
      autotick = T,showticklabels = FALSE,tickcolor = toRGB("white"))
      , yaxis = list(title='Distribution'))
```

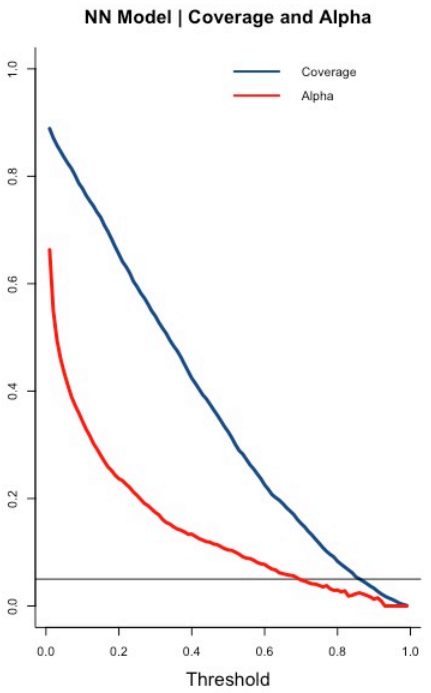
Note that NY is highly under-represented in the final sample. In the original code we included specific variables for cities which their name coincide with the name of the state. This improved the accuracy drastically.

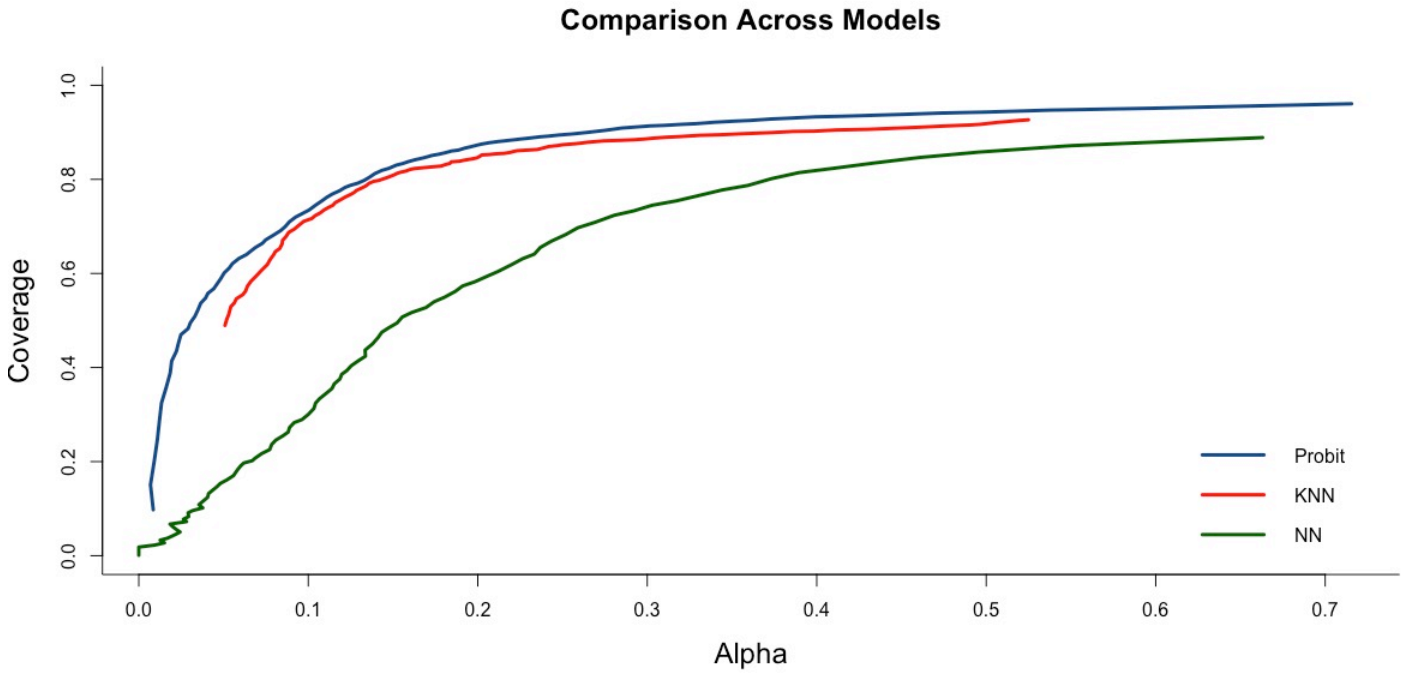
Plot with only positive Values

```
plot_ly(x=Geography$Place[Predicted.Geography$freq>0] ,
        y = Geography$freq[Predicted.Geography$freq>0]/sum(Geography$freq),
        type = "bar",opacity = 0.9,name='Sample')  %>%
  add_trace(x=Predicted.Geography$Place[Predicted.Geography$freq>0],
            y = Predicted.Geography$freq[Predicted.Geography$freq>0]/
              sum(Predicted.Geography$freq),
            type = "bar",opacity = 0.6,name='Predicted') %>%
  layout(barmode='overlay', showlegend = T,xaxis = list(title=' ',
      autotick = T,showticklabels = FALSE,tickcolor = toRGB("white"))
      , yaxis = list(title='Distribution'))
```

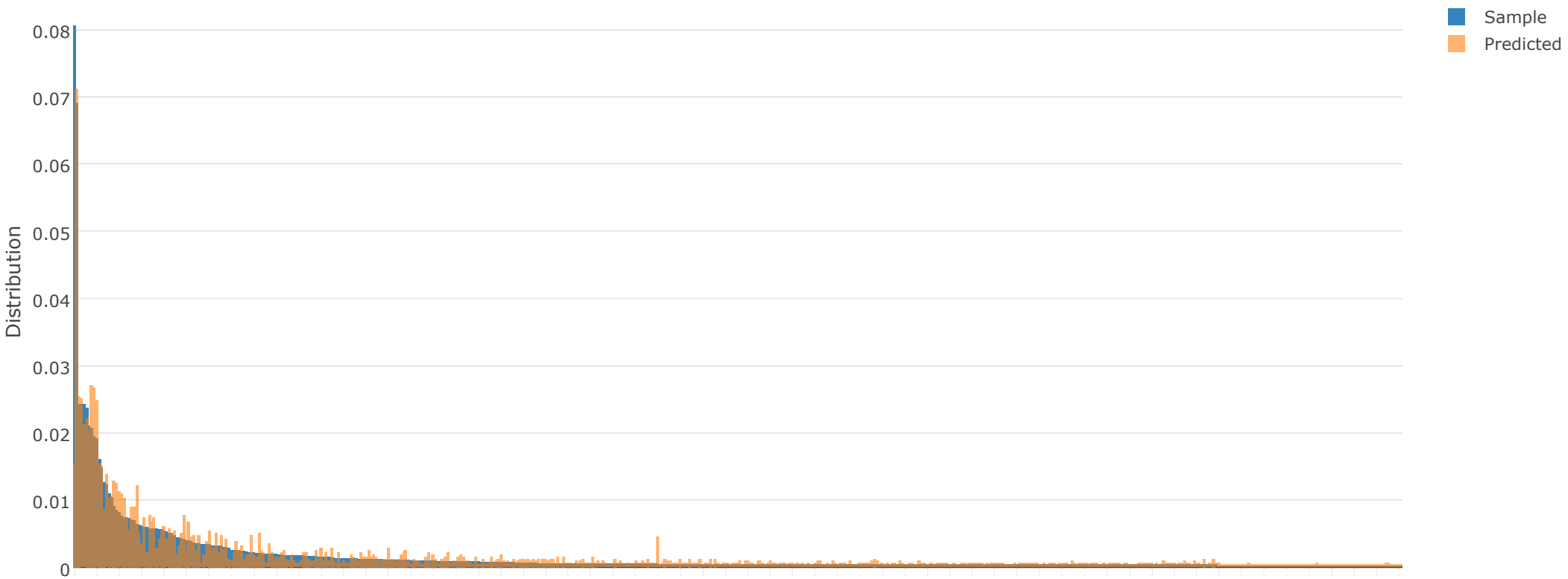

Probit Model | Coverage and Alpha
Coverage
Alpha
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Threshold


KNN Model | Coverage and Alpha
Coverage
Alpha
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Threshold


NN Model | Coverage and Alpha
Coverage
Alpha
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Threshold


Comparison Across Models
Coverage
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Alpha
Probit
KNN
NN

Sample
Predicted
Distribution
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0


Sample
Predicted
Distribution
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0