

```
#####
# Calculates shortest on-road distance between each sub-Saharan African capital
# by using a Google Maps Python API
#####

import csv
import numpy as np
import googlemaps

try:
    wddata='/Users/lillianpetersen/iiasa/data/supply_chain/'
    wdfigs='/Users/lillianpetersen/iiasa/figs/'
    wdvars='/Users/lillianpetersen/iiasa/saved_vars/'
    f=open(wddata+'population/CAPITALVERSIONcasenumbers.csv','r')
except:
    wddata='C:/Users/garyk/Documents/code/riskAssessmentFromPovertyEstimations/supply_chain/data/'
    wdfigs='C:/Users/garyk/Documents/code/riskAssessmentFromPovertyEstimations/supply_chain/figs/'
    wdvars='C:/Users/garyk/Documents/code/riskAssessmentFromPovertyEstimations/supply_chain/vars/'

# listof50k = open('C:/Users/garyk/Documents/python_code/riskAssessmentFromPovertyEstimations/data/popula
#
# citynames=["" for x in range(629)]
# countrynames=["" for x in range(629)]
# previouscity=1
# firstline=True
# index = 0
# for line in listof50k:
#     if firstline:
#         firstline = False
#         continue
#     tmp=line.split(',')
#     marketname=tmp[0]
#     countrycode=tmp[2]
#     if(np.amax([marketname==citynames[i] for i in range(len(citynames))])==0):
#         citynames[index]=marketname
#         countrynames[index]=countrycode
#         index=index+1
#     previouscity=tmp[0]

###cost per country
countrycosted=[]
capitalcosted=[]
rutfprice=[]
rusfprice=[]
scplusprice=[]
f=open(wddata+'foodstuffs/pricesCorrected_intl.csv')
code=np.zeros(shape=(247),dtype=int)
i=-1
for line in f:
    i+=1
    tmp=line.split(',')
    countrycosted.append(tmp[0])
    rutfprice.append(tmp[1])
    rusfprice.append(tmp[2])
    scplusprice.append(tmp[3])
    capitalcosted.append(tmp[4][:-1])

countrycosted[0]='Angola'
countrycosted[5]="Cote d'Ivoire"

###capitalonly
subsaharancountry=[]
subsaharancapital=[]
indexedwasting=np.zeros(shape=43)
```

```

indexedSAM=np.zeros(shape=43)
indexedstunting=np.zeros(shape=43)
indexedMAM=np.zeros(shape=43)
f=open(wddata+'population/CAPITALVERSIONcasenumbers.csv','r')
i=-1
for line in f:
    i+=1
    tmp=line.split(',')
    subsaharancountry.append(tmp[0])
    indexedwasting[i]=float(tmp[1])
    indexedSAM[i]=float(tmp[2])
    indexedMAM[i]=float(tmp[3])
    indexedstunting[i]=float(tmp[4])
    subsaharancapital.append(tmp[5][:-1])

#####
# mapping between
#####
gmaps = googlemaps.Client(key='AIzaSyCGTiyRUR08brCATN_p4gskQQim4m6G5Tk')
distanceArray=np.zeros(shape=(33,43))
distanceDictionary={}
counter=0
matchcountries=subsaharancountry
listofcities=subsaharancapital
for i in range(len(capitalcosted)):
    distanceDictionary[capitalcosted[i]]=[]
    for j in range(len(listofcities)):
        if capitalcosted[i]==listofcities[j]:
            distanceDictionary[capitalcosted[i]].append(0)
            distanceArray[i,j]=0
        else:
            gmapreturn=(gmaps.distance_matrix(capitalcosted[i]+countrycosted[i],listofcities[j]+matchcountries[j]))
            if(gmapreturn=={u'status': u'ZERO_RESULTS'} or gmapreturn=={u'status': u'NOT_FOUND'}):
                print(listofcities[i] + matchcountries[i])
                distanceDictionary[listofcities[i]].append(99999999)
                distanceArray[i,j]=99999999
            else:
                distanceDictionary[capitalcosted[i]].append(gmapreturn['distance']['value'])
                distanceArray[i,j]=gmapreturn['distance']['value']

np.savetxt(wddata + "travel_time/INTLcapitaldistanceArray.csv", distanceArray, delimiter=",")

with open(wddata+'optiarrays/rutfdictionary.json', 'w') as fp:
    json.dump(rutfdictionary, fp, sort_keys=True)

```