

СРСП "Изменения продовольственной безопасности в странах Центральной Азии с 2014 по 2017 год"

```
In [1]: import pyreadr
import os
import pandas
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import style
```

Global variables

```
In [2]: fAdMod_Sev = 0
fAdSev = 0
popAdSev = 0
popAdMod_Sev = 0
populationAd = 0

fChildMod_Sev, popChildMod_Sev = 0, 0
fChildSev, popChildSev, populationChild = 0, 0, 0
fTotalModSev, fTotalSev = 0, 0
popTotalModSev, popTotalSev = 0, 0

centralAsiaDataFramesArray = []
centralAsia = None

foodTotalModSevDict = {}
```

Formulas to calculate Adults food insecure.

```
In [3]: def popAdMod_SevFunc(dataframe, fAdultModSev):
    global popAdMod_Sev, popAdMod_Sev
    population = dataframe["N_adults"].astype("int").sum()
    popAdMod_Sev = fAdultModSev * population
    print(f"Pop AdultModSev = {popAdMod_Sev}")
    return popAdMod_Sev
```

```
In [4]: def popAdSevFunc(dataframe, fAdultSev):
    global populationAd
    populationAd = dataframe["N_adults"].astype("int").sum()
    popSev = fAdultSev * populationAd
    print(f"Pop AdultSev = {popSev}")
    return popSev
```

```
In [5]: def calculateAdultFoodAtModAndSev(dataframe):
    global fAdMod_Sev
    dataframe["Prob_Mod_Sev"].fillna(dataframe["Prob_Mod_Sev"].notnull().astype("float").mean(), inplace=True)
    dataframe["wt"].fillna(dataframe["wt"].notnull().astype("float").mean(), inplace=True)

    fAdMod_Sev = (dataframe["Prob_Mod_Sev"].astype("float") *
                  dataframe["wt"].astype("float").sum() / dataframe["wt"].astype("float").sum())
    print(f"F AdultModSev = {fAdMod_Sev}")
    popAdMod_SevFunc(dataframe, fAdMod_Sev)
    return dataframe
```

```
In [6]: def calculateAdultFoodAtSev(dataframe):
    global fAdSev
    dataframe["Prob_sev"].fillna(dataframe["Prob_sev"].notnull().astype("float").mean(), inplace=True)
    dataframe["wt"].fillna(dataframe["wt"].notnull().astype("float").mean(), inplace=True)

    fAdSev = (dataframe["Prob_sev"].astype("float") *
              dataframe["wt"].astype("float").sum() / dataframe["wt"].astype("float").sum())

    print(f"F AdultSev = {fAdSev}")
    pop = popAdSevFunc(dataframe, fAdSev)
    return dataframe
```

Formulas to calculate Child food insecure.

```
In [7]: def calculateChildrenWeight(dataframe):
    dataframe["N_child"] = np.where((dataframe.N_child == "10+"), 10, dataframe.N_child)
    dataframe["N_adults"] = np.where((dataframe.N_adults == "10+"), 10, dataframe.N_adults)

    dataframe["N_child"] = np.where((dataframe.N_child == ""), 0, dataframe.N_child)
    dataframe["N_adults"] = np.where((dataframe.N_adults == ""), 0, dataframe.N_adults)

    dataframe["N_child"].fillna(dataframe["N_child"].notnull().astype("float").mean(), inplace=True)
    dataframe["N_adults"].fillna(dataframe["N_adults"].notnull().astype("float").mean(), inplace=True)
```

```
dataframe["Wt_child"] = (dataframe["wt"] /
                        dataframe["N_adults"].astype("float")) * dataframe["N_child"].astype("float")
return dataframe
```

```
In [8]: def popChildMod_SevFunc(dataframe, fChildModSev):
        global populationChild, popChildMod_Sev
        populationChild = dataframe["N_child"].astype("int").sum()
        popChildMod_Sev = fChildModSev * populationChild
        print(f"Pop ChildModSev = {popChildMod_Sev}")
        return popChildMod_Sev
```

```
In [9]: def popChildSevFunc(dataframe, fChildSev):
        global popChildSev
        population = dataframe["N_child"].astype("int").sum()
        popChildSev = fChildSev * population
        print(f"Pop ChildModSev = {popChildSev}")
        return popChildSev
```

```
In [10]: def calculateChildFoodAtMod_Sev(dataframe):
        global fChildMod_Sev
        dataframe["Prob_Mod_Sev"].fillna(dataframe["Prob_Mod_Sev"].notnull().astype("float").mean(), inplace=True)
        dataframe["Wt_child"].fillna(dataframe["Wt_child"].notnull().astype("float").mean(), inplace=True)

        fChildMod_Sev = (dataframe["Prob_Mod_Sev"].astype("float") *
                        dataframe["Wt_child"].astype("float")).sum() / dataframe["Wt_child"].astype("float").sum()
        print(f"F childModSev = {fChildMod_Sev}")
        popChildMod_SevFunc(dataframe, fChildMod_Sev)
        return dataframe
```

```
In [11]: def calculateChildFoodAtSev(dataframe):
        global fChildSev
        dataframe["Prob_sev"].fillna(dataframe["Prob_sev"].notnull().astype("float").mean(), inplace=True)
        dataframe["Wt_child"].fillna(dataframe["Wt_child"].notnull().astype("float").mean(), inplace=True)

        fChildSev = (dataframe["Prob_sev"].astype("float") *
                    dataframe["Wt_child"].astype("float")).sum() / dataframe["Wt_child"].astype("float").sum()

        print(f"F childSev = {fChildSev}")
        popChildSevFunc(dataframe, fChildSev)
        return dataframe
```

Total food insecure.

```
In [12]: def totalMod_SevFunc(popAdMod_Sev, popChildMod_Sev, popAdult, popChild):
        total = (popAdMod_Sev + popChildMod_Sev) / (popAdult + popChild)
        print(f"Total Mod_Sev = {total}")
        return total
```

```
In [13]: def totalSevFunc(popAdSev, popChildSev, popAdult, popChild):
        total = (popAdSev + popChildSev) / (popAdult + popChild)
        print(f"Total Sev = {total}")
        return total
```

```
In [14]: def popTotalMod_SevFunc(fToal, popTotal):
        total = fToal * popTotal
        print(f'Pop Total Mod_SEV = {total}')
        return total
```

```
In [15]: def popTotalSevFunc(fToal, popTotal):
        total = fToal * popTotal
        print(f'Pop Total SEV = {total}')
        return total
```

Add year and country column.

```
In [16]: def addYears(dataframe, dir):
        year = dir.split("_")[-1]
        country = dir.replace('/', '_').split('_')[-2]

        dataframe["year"] = year
        dataframe["country"] = country
        if country not in foodTotalModSevDict:
            foodTotalModSevDict[country] = {}
        return dataframe, country, year
```

Main cycle.

```
In [17]: path = 'Lab5'
        for dir, folder, files in os.walk(path):
            for file in files:
```

```

if file[-1:-6:-1][:-1] == "RData":
    dataFrame = None
    try:
        result = pyreadr.read_r(f'{dir}/{file}')['data']
        dataFrame = pandas.DataFrame(result)
    except pyreadr.custom_errors.LibrdDataError as e:

        file_path = f'{dir}/{file}'.split(".")
        dataFrame = pandas.read_stata(file_path[0]+".dta")

print("")
print(dir)

dataFrame, country, year = addYears(dataFrame, dir)

dataFrame = calculateChildrenWeight(dataFrame)

dataFrame = calculateAdultFoodAtModAndSev(dataFrame)
dataFrame = calculateAdultFoodAtSev(dataFrame)

dataFrame = calculateChildFoodAtMod_Sev(dataFrame)
dataFrame = calculateChildFoodAtSev(dataFrame)

fTotalModSev = totalMod_SevFunc(popAdMod_Sev, popChildMod_Sev, populationAd, populationChild)
fTotalSev = totalSevFunc(popAdSev, popChildSev, populationAd, populationChild)

popTotalModSev = popTotalMod_SevFunc(fTotalModSev, populationAd+populationChild)
popTotalSev = popTotalSevFunc(fTotalSev, populationAd+populationChild)

centralAsiaDataFramesArray.append(dataFrame)

foodTotalModSevDict[country].update({int(year): fTotalModSev})

with pandas.ExcelWriter(f'{dir}/{file[0:-6]}.xlsx') as writer:
    dataFrame.to_excel(writer)

```

Lab5/KGZ_2015

F AdultModSev = 0.25450719076333794
 Pop AdultModSev = 751.5597343241369
 F AdultSev = 0.08826532543116555
 Pop AdultSev = 260.6475059982319
 F childModSev = 0.28286003567985013
 Pop ChildModSev = 481.71064076278475
 F childSev = 0.1007322374659121
 Pop ChildModSev = 171.5470004044483
 Total Mod_Sev = 0.26487765787949347
 Total Sev = 0.036844287028446804
 Pop Total Mod_SEV = 1233.2703750869216
 Pop Total SEV = 171.5470004044483

Lab5/KGZ_2014

F AdultModSev = 0.2686160855334666
 Pop AdultModSev = 754.5425842635077
 F AdultSev = 0.12309877508300224
 Pop AdultSev = 345.7844592081533
 F childModSev = 0.2848776580675904
 Pop ChildModSev = 435.29306152727816
 F childSev = 0.11989084946798746
 Pop ChildModSev = 183.19321798708484
 Total Mod_Sev = 0.27434531837463355
 Total Sev = 0.042239616782818734
 Pop Total Mod_SEV = 1189.8356457907857
 Pop Total SEV = 183.19321798708484

Lab5/TAIJ_2017

F AdultModSev = 0.42703399329423214
 Pop AdultModSev = 1277.6857079363426
 F AdultSev = 0.2668375185164021
 Pop AdultSev = 798.3778554010751
 F childModSev = 0.44119904372841023
 Pop ChildModSev = 1097.2620217525562
 F childSev = 0.2800630154268552
 Pop ChildModSev = 696.5167193665889
 Total Mod_Sev = 0.43346372142524164
 Total Sev = 0.12712478907950153
 Pop Total Mod_SEV = 2374.947729688899
 Pop Total SEV = 696.5167193665889

Lab5/TAIJ_2016

F AdultModSev = 0.37962178816609393
 Pop AdultModSev = 1545.4402996241683
 F AdultSev = 0.25001094310937333
 Pop AdultSev = 1017.7945493982588
 F childModSev = 0.4042986872532754
 Pop ChildModSev = 949.6976163579438
 F childSev = 0.2634944208620596
 Pop ChildModSev = 618.948394604978
 Total Mod_Sev = 0.38865076572930096
 Total Sev = 0.09640940725934237

Pop Total Mod_SEV = 2495.137915982112
Pop Total SEV = 618.948394604978

Lab5/UZB_2014

F AdultModSev = 0.11862667915550518
Pop AdultModSev = 453.5097944114963
F AdultSev = 0.03792579065023992
Pop AdultSev = 144.9902976558672
F childModSev = 0.13379523835501128
Pop ChildModSev = 210.72750040914278
F childSev = 0.0488327810415117
Pop ChildModSev = 76.91163014038094
Total Mod_Sev = 0.12305248144139293
Total Sev = 0.014248171571022775
Pop Total Mod_SEV = 664.237294820639
Pop Total SEV = 76.91163014038094

Lab5/Kazakhstan_2015

F AdultModSev = 0.10833418531131406
Pop AdultModSev = 305.1774000219717
F AdultSev = 0.06827019942087646
Pop AdultSev = 192.317151768609
F childModSev = 0.10430277536504172
Pop ChildModSev = 108.37058360427835
F childSev = 0.06214505906661247
Pop ChildModSev = 64.56871637021035
Total Mod_Sev = 0.10724792106489887
Total Sev = 0.01674499905866451
Pop Total Mod_SEV = 413.5479836262501
Pop Total SEV = 64.56871637021035

Lab5/UZB_2015

F AdultModSev = 0.1337281899189894
Pop AdultModSev = 482.22385284787583
F AdultSev = 0.03076289200578743
Pop AdultSev = 110.93098857286947
F childModSev = 0.13048726764079263
Pop ChildModSev = 202.2552648432286
F childSev = 0.024254078068145793
Pop ChildModSev = 37.593821005625976
Total Mod_Sev = 0.13275390180199853
Total Sev = 0.007291276378127614
Pop Total Mod_SEV = 684.4791176911044
Pop Total SEV = 37.593821005625976

Lab5/Kazakhstan_2014

F AdultModSev = 0.16231480450213137
Pop AdultModSev = 437.7630277422483
F AdultSev = 0.09504546768907865
Pop AdultSev = 256.3376263574451
F childModSev = 0.15934214475558897
Pop ChildModSev = 148.9849053464757
F childSev = 0.08759092586899962
Pop ChildModSev = 81.89751568751464
Total Mod_Sev = 0.1615495410486575
Total Sev = 0.022548875464624075
Pop Total Mod_SEV = 586.747933088724
Pop Total SEV = 81.89751568751464

Lab5/KGZ_2016

F AdultModSev = 0.21143802037433193
Pop AdultModSev = 569.1911508477016
F AdultSev = 0.05375027799738904
Pop AdultSev = 144.6957483689713
F childModSev = 0.2320711634283463
Pop ChildModSev = 427.4750830350139
F childSev = 0.054706093060772454
Pop ChildModSev = 100.76862341794286
Total Mod_Sev = 0.21982051916248688
Total Sev = 0.022225104415073416
Pop Total Mod_SEV = 996.6662338827155
Pop Total SEV = 100.76862341794286

Lab5/KGZ_2017

F AdultModSev = 0.2212277935899953
Pop AdultModSev = 555.5029897044782
F AdultSev = 0.06535282976090591
Pop AdultSev = 164.10095552963475
F childModSev = 0.2626158322357305
Pop ChildModSev = 476.6477355078508
F childSev = 0.06737363256216217
Pop ChildModSev = 122.28314310032434
Total Mod_Sev = 0.23859240065009918
Total Sev = 0.028267023370393975
Pop Total Mod_SEV = 1032.150725212329
Pop Total SEV = 122.28314310032434

Lab5/TAIJ_2014

F AdultModSev = 0.2803618188130671

```

Pop AdultModSev = 1157.053226241528
F AdultSev = 0.18062460475394948
Pop AdultSev = 745.4377438195495
F childModSev = 0.27874314709113673
Pop ChildModSev = 654.4889093699891
F childSev = 0.1718089812341738
Pop ChildModSev = 403.4074879378401
Total Mod_Sev = 0.2797748471986899
Total Sev = 0.06230231473943476
Pop Total Mod_SEV = 1811.542135611517
Pop Total SEV = 403.4074879378401

```

```

Lab5/TAIJ_2015
F AdultModSev = 0.2422633391058232
Pop AdultModSev = 960.8164028936949
F AdultSev = 0.15159237179200596
Pop AdultSev = 601.2153465270957
F childModSev = 0.2549562960185807
Pop ChildModSev = 584.1048741785685
F childSev = 0.16118814458960184
Pop ChildModSev = 369.2820392547778
Total Mod_Sev = 0.24691086416369878
Total Sev = 0.059019024972794924
Pop Total Mod_SEV = 1544.9212770722634
Pop Total SEV = 369.2820392547778

```

```

Lab5/UZB_2017
F AdultModSev = 0.18567370758094195
Pop AdultModSev = 650.4149976560396
F AdultSev = 0.04866023188724257
Pop AdultSev = 170.45679230101072
F childModSev = 0.19478420541438476
Pop ChildModSev = 319.44609687959104
F childSev = 0.04581924800176155
Pop ChildModSev = 75.14356672288893
Total Mod_Sev = 0.18857886341350003
Total Sev = 0.014610843228249841
Pop Total Mod_SEV = 969.8610945356306
Pop Total SEV = 75.14356672288893

```

```

Lab5/Kazakhstan_2016
F AdultModSev = 0.1376915809414728
Pop AdultModSev = 310.7698981849041
F AdultSev = 0.0827059274197974
Pop AdultSev = 186.66727818648275
F childModSev = 0.13153291584874596
Pop ChildModSev = 116.80122927368642
F childSev = 0.06711565427399618
Pop ChildModSev = 59.59870099530861
Total Mod_Sev = 0.1359526637388205
Total Sev = 0.018950302383245982
Pop Total Mod_SEV = 427.57112745859047
Pop Total SEV = 59.59870099530861

```

```

Lab5/UZB_2016
F AdultModSev = 0.11683565660925159
Pop AdultModSev = 425.2817900576758
F AdultSev = 0.029608512398456254
Pop AdultSev = 107.77498513038077
F childModSev = 0.119741782139756
Pop ChildModSev = 186.67743835587962
F childSev = 0.026247810189346625
Pop ChildModSev = 40.92033608519139
Total Mod_Sev = 0.11770710298394987
Total Sev = 0.007870809018117212
Pop Total Mod_SEV = 611.9592284135554
Pop Total SEV = 40.92033608519139

```

```

Lab5/Kazakhstan_2017
F AdultModSev = 0.11964925266592337
Pop AdultModSev = 281.29539301758587
F AdultSev = 0.046029235560190236
Pop AdultSev = 108.21473280200725
F childModSev = 0.17667620861714553
Pop ChildModSev = 191.34033393236862
F childSev = 0.06807505966671266
Pop ChildModSev = 73.72528961904982
Total Mod_Sev = 0.137634166263819
Total Sev = 0.021469216546024992
Pop Total Mod_SEV = 472.63572694995446
Pop Total SEV = 73.72528961904982

```

Merge all countries to one excel file.

```

In [18]: centralAsia = pandas.concat(centralAsiaDataFramesArray)
centralAsia.reset_index(drop=True, inplace=True)

```

```

with pandas.ExcelWriter(f'CentralAsia.xlsx') as writer:
    centralAsia.to_excel(writer)

for i in foodTotalModSevDict:
    values = []
    years = [2014, 2015, 2016, 2017]
    for z in years:
        values.append(foodTotalModSevDict[i][z]*100)
    foodTotalModSevDict[i] = values

```

Graphics of food insecure for each Central Asia country.

```

In [19]: # Countries
for i in foodTotalModSevDict:
    style.use('ggplot')
    x = [2014, 2015, 2016, 2017]
    y = foodTotalModSevDict[i]
    plt.plot(x, y, 'r', label=i, linewidth=5)
    plt.title(i)
    plt.savefig(f'{i}.png', bbox_inches='tight')
    plt.close()

```

Central Asia food insecure graphic.

```

In [20]: # Central Asia
dfArray = pandas.DataFrame(foodTotalModSevDict).T
_2014 = dfArray[0].mean()
_2015 = dfArray[1].mean()
_2016 = dfArray[2].mean()
_2017 = dfArray[3].mean()

style.use('ggplot')
x = [2014, 2015, 2016, 2017]
y = [_2014, _2015, _2016, _2017]
plt.plot(x, y, 'g', label="Central Asia", linewidth=5)
plt.title("Central Asia")
plt.savefig('Central Asia.png', bbox_inches='tight')
plt.show()
plt.close()

```

