Estimating GHG emission based on the OECD-FAO Outlook projections - Clean file

Eduard Bukin

14 March 2017

Process

The purpose of this document is to explain the process of reproducing the GHG emissions data from the FAOSTAT using as the activity data the numbers of OECD-FAO Agricultural Outlook.

The process is structured around particular domains, data for which has to be reproduced.

For the domains GE, GM, GU, GP and GR emissions are reproduced based on the projected activity data, emissions related to other domains are treated separately and data is reproduced based on various assumptions:
* GB, GH and GA domains are projected as a constant share of total emissions assuming the share based on the 5 years average share in the last know historical period.
* GY domain emissions data are approximated based on the area and yields of crops relevant to the nitrogenous fertilizers consumption.
* GV domain data is kept at the constant level as it is assumed in the FAOSTAT. Alternatively, we test a situation, when emissions from the organic soils are changing with the same rate as the area utilized under the palm oil production.

Below, we elaborate more explicitly on the methodology of the GHG estimation for different domains.

Setup

Installing packages

```
packs <- c("plyr", "tidyverse", "dplyr", "tidyr","readxl", "stringr",
           "DT", "rmarkdown", "gridExtra", "grid", "ggplot2", "ggthemes",
           "scales", "devtools", "gridGraphics")
lapply(packs[!packs %in% installed.packages()[,1]], install.packages,
       dependencies = TRUE)
lapply(packs, require, character.only = TRUE)
```

Making sure that the number of digits displayed is large enough.

```
options(scipen=20)
```

Loading locally developed functions

```
l_ply(str_c("R/", list.files("R/", pattern="*.R")), source)
```

Loading data

Outlook data

First we load all outlook data. If there is no data save in the Rdata file we reload all data from the CSV file.

```
olRDFile <- "data/outlook.Rdata"
if(!file.exists(olRDFile)) {
  #olFile <- "C:/Users/Bukin/OneDrive - Food and Agriculture Organization/outlookGHG/data/base17.csv"
  olFile <- "C:/2017/Master/BaselineOutput.csv"
  if(!file.exists(olFile)) olFile <- "data/base17.csv"
  ol <- load_troll_csv(olFile, d.source = "") %>%
    select(AreaCode, ItemCode, ElementCode, Year, Value)
  save(ol, file = olRDFile)
} else {
  load(file = olRDFile)
}
```

FAOSTAT data

Next step is loading all FAOSTAT data. Since FAOSTAT data combines data from multiple domains, we load it all in on .Rdata file. In case if there is no such file, we reload all data from each domain specific file and save it in the R data file for further use.

```
fsRDFile <- "data/all_fs_emissions.Rdata"
if(!file.exists(fsRDFile)) {
  files <-
    c("data/Emissions_Agriculture_Agriculture_total_E_All_Data_(Norm).csv",
      "data/Emissions_Agriculture_Burning_crop_residues_E_All_Data_(Norm).csv",
      "data/Emissions_Agriculture_Burning_Savanna_E_All_Data_(Norm).csv",
      "data/Emissions_Agriculture_Crop_Residues_E_All_Data_(Norm).csv",
      "data/Emissions_Agriculture_Cultivated_Organic_Soils_E_All_Data_(Norm).csv",
      "data/Emissions_Agriculture_Enteric_Fermentation_E_All_Data_(Norm).csv",
      "data/Emissions_Agriculture_Manure_applied_to_soils_E_All_Data_(Norm).csv",
      "data/Emissions_Agriculture_Manure_left_on_pasture_E_All_Data_(Norm).csv",
      "data/Emissions_Agriculture_Manure_Management_E_All_Data_(Norm).csv",
      "data/Emissions_Agriculture_Rice_Cultivation_E_All_Data_(Norm).csv",
      "data/Emissions_Land_Use_Burning_Biomass_E_All_Data_(Norm).csv",
      "data/Emissions_Land_Use_Cropland_E_All_Data_(Norm).csv",
      "data/Emissions_Land_Use_Forest_Land_E_All_Data_(Norm).csv",
      "data/Emissions_Land_Use_Grassland_E_All_Data_(Norm).csv",
      "data/Emissions_Land_Use_Land_Use_Total_E_All_Data_(Norm).csv")
  domains <- c("GT", "GB", "GH", "GA", "GV", "GE", "GU",
               "GP", "GM", "GR", "GI", "GC", "GF", "GG", "GL")
  fs <-
    ddply(tibble(files, domains),
          .(files),
          function(x) {
            if(file.exists(x$files)) {
              read.fs.bulk(x$files) %>%
                mutate(Domain = as.character(x$domains))
            }
          }) %>% tbl_df()
  els <- fs %>%
    select(Domain, ElementCode, ElementName, Unit) %>%
    distinct()
  its <- fs %>%
    select(Domain, ItemCode, ItemName) %>%
    distinct()
```

2

```
  fs <-
    fs %>%
    select(Domain, AreaCode, ItemCode, ElementCode,
           Year, Value, Unit, ElementName, ItemName)
  save(fs, its, els, file = fsRDFile)
} else {
  load("data/all_fs_emissions.Rdata")
}
```

Mapping tables

Besides data from Outlook and FAOSTAT, we also need specific mapping tables which explain mappings
from FAOSTAT to Outlook areas and items.

```
itemsMTFile <- "mappingTables/fs_outlook_items_mt.csv"
itemsMT <- read_csv(itemsMTFile,
                    col_types = cols(
                      ItemCode = col_integer(),
                      OutlookItemCode = col_character(),
                      ItemCodeAggSign = col_character()
                    ))
```

Table `elementsMT` describes mapping and adjustment of elements from FAOSTAT to outlook.

```
elementsMTFile <- "mappingTables/fs_outlook_elements_mt.csv"
elementsMT <-
  read_csv(elementsMTFile,
           col_types = cols(
             Domain = col_character(),
             ItemCode = col_character(),
             ElementCode = col_integer(),
             OutlookElementCode = col_character(),
             OutlookAdjustment = col_double()
           ))
```

Table `emissionsMT` describes mapping and assumption behind projection of the implied emissions factor for
the years of projection.

```
emissionsMTFile <- "mappingTables/fs_outlook_emissions_mt.csv"
emissionsMT <-
  read_csv(emissionsMTFile,
           col_types = cols(
             Domain = col_character(),
             Emissions = col_integer(),
             OutlookEmissions = col_character(),
             ActivityElement = col_character(),
             EFLag = col_integer(),
             GHG = col_character()
           ))
```

Implementing the process

Reproducing GR, GE, GU, GP and GM

Domains discussed in this part are estimated based on the activity data, projected in the OECD-FAO Agricultural Outlook. There domains are:

- GR - Rice cultivation

- GE - Enteric fermentation
- GM - Manure Management
- GU - Manure applied to soils
- GP - Manure left of pastures

The overall process consist of several important steps. All steps are organised in the body of a function `outlook_emissions`. This function utilities faostat data, outlook data and previously loaded mapping tables for reproducing emissions for the pre-defined domain. The steps of reproduction are the following:

1. Mapping FAOSTAT Areas to the outlook regions re-estimating activity data and emissions respectively. Mapping the FAOSTAT activity data to the outlook activity data aggregating FAOSTAT items to the outlook items and re-estimating emissions and activity data according to aggregating. This is done with the `map_fs_data` function, which uses items and elements mapping tables and faostat filtered to one domain data. Thing the function uses `map_fs2ol` and `agg_ol_regions` which does the aggregation of the FAOSTAT data to the outlook structure. In the mapping process, some of the items and elements may be aggregated by subtracting one from another what is specified with the mapping tables.

2. Adjusting outlook activity data to the baseline level derived from the FAOSTAT historical data. This step is the part of the `outlook_emissions` function, where mapped faostat data is used for subset the outlook data to the items and elements relevant for one domain with the function `subset_outlook`. After sub-setting, we apply function `adjust_outlook_activity` in order to adjust activity data from the outlook to the levels of the FAOSTAT in the historical period.

3. At the next step we `reestimate_emissions` data based on the activity if such was prepared in the OUTLOOK data.

4. In some cases, for some items and elements outlook does not have any activity data. In such cases, we estimate the emissions for the missing items and elements combinations based on the constant share of these items and elements in the known and estimated emissions. Constant share is assumed based on the 5 years average share calculated on the last available. This step is made with the function `estimate_missing_emissions`.

5. At the next step we convert all GHG to the GHG expressed in the CO2 equivalent with the function `convert_ghg`.

6. After the numbers are re-estimated in the steps 1-4, we aggregate regions relevant to the outlook such as "Big five" region, Cosimo and Aglink regions and the World total. The regional aggregating is made using the function `agg_ol_regions`.

We preform all above explained calculations for one domain at the time. That allows us to apply the same functions and approaches to every domain maintaining methodological consistency.

Reproducing data.

```r
gm <- outlook_emissions(fs, ol, DomainName = "GM")
ge <- outlook_emissions(fs, ol, DomainName = "GE")
gu <- outlook_emissions(fs, ol, DomainName = "GU")
gp <- outlook_emissions(fs, ol, DomainName = "GP")
gr <- outlook_emissions(fs, ol, DomainName = "GR")
```

Reproducing GV

For the GV - Cultivating Organic Soils domain we repeat the last know values.

```
gv_fs <-
  fs %>%
  filter(Year %in% c(2000:2016), Domain == "GT") %>%
  map_fs_data(., fsYears = c(2000:2016)) %>%
  filter(ItemCode == "GV") %>%
  filter(AreaCode %in% get_ol_countries())
gv <-
  gv_fs %>%
  filter(Year %in% (max(Year))) %>%
  mutate(Year = max(Year)) %>%
  group_by_(.dots = names(.)[!names(.) %in% c("Value")]) %>%
  summarise(Value = mean(Value)) %>%
  ungroup()
# Expanding projected emissions for the projected period
gv <-
  ldply((max(gv$Year) + 1):2030, function(x) {
    gv %>%
      mutate(Year = x)
  }) %>%
  tbl_df() %>%
  bind_rows(gv_fs) %>%
  mutate(d.source = "Outlook")
gv <-
  gv %>%
  bind_rows(gv_fs) %>%
  bind_rows(gv %>% filter(d.source ==  "Outlook") %>% mutate(d.source = "no adj. Outlook"))%>%
  arrange(Domain, AreaCode, ItemCode, ElementCode, Year) %>%
  agg_all_ol_regions()
```

Reproducing GB, GH and GA

Reproducing emissions for the domains Burning crop residues, Burning Savanna and crop residues. To reproduce emissions for these domains such we use the constant share of the emissions from this domains in the estimable emissions from agriculture and continue this trend to future.

Projecting of these domains is made based on the total aggregates of all estimated domains and Agriculture total domain.

```
gtpart <-
  bind_rows(list(gm, ge, gu, gp, gr)) %>%
  agg_ghg_domains %>%
  agg_total_emissions
gt <-
  outlook_emissions(fs,
                    gtpart %>% filter(d.source == "Outlook"),
                    DomainName = "GT", useActivity = FALSE) %>%
  filter(!ItemCode %in% c("GM", "GE", "GU", "GP", "GR", "GV")) %>%
  bind_rows(gtpart, gv) %>%
  join_names()
```

```
## Parsed with column specification:
```

```
## cols(
##   ItemCode = col_character(),
##   ItemName = col_character()
## )

## Parsed with column specification:
## cols(
##   ElementCode = col_character(),
##   ElementName = col_character(),
##   Unit = col_character()
## )
```

Reproducing GI, GC, GG and GF domains

When reproducing data for the GI - Burning Biomass, GC - Cropland and GG - Grassland domains we assume that the values of emissions remains constant at the levels of the last 5 years average. Blow we reproduce that.

```
# Number of years lag for average projections
nYears <- max(5 - 1, 0)
lastYear = 2030
# Reproducing emissions for the GI, GC, GG
ol_lu_fs <-
  fs %>%
  filter(Year %in% c(2000:2016), Domain == "GL") %>%
  map_fs_data(., fsYears = c(2000:2016)) %>%
  filter(AreaCode %in% get_ol_countries())
ol_lu <-
  ol_lu_fs %>%
  filter(Year %in% (max(Year) - nYears + 1):max(Year)) %>%
  mutate(Year = max(Year)) %>%
  group_by_(.dots = names(.)[!names(.) %in% c("Value")]) %>%
  summarise(Value = mean(Value)) %>%
  ungroup()
# Expanding projected emissions for the projected period
ol_lu <-
  ldply((max(ol_lu$Year) + 1):lastYear, function(x) {
    ol_lu %>%
      mutate(Year = x)
  }) %>%
  tbl_df() %>%
  bind_rows(ol_lu_fs) %>%
  mutate(d.source = "Outlook") %>%
  arrange(Domain, AreaCode, ItemCode, ElementCode, Year) %>%
  filter(ItemCode != "GF")
ol_lu <-
  ol_lu %>%
  mutate(d.source = "no adj. Outlook") %>%
  bind_rows(ol_lu)%>%
  bind_rows(ol_lu_fs)
```

For the domain GF - Forestland we continue the last know value to the future.

```
# Reproducing emissions for the GF
gf_sf <-
```

```
  fs %>%
    filter(Year %in% c(2000:2016), Domain == "GL") %>%
    map_fs_data(., fsYears = c(2000:2016)) %>%
    filter(ItemCode == "GF")

gf <-
  gf_sf %>%
  filter(AreaCode %in% get_ol_countries()) %>%
  filter(Year %in% (max(Year))) %>%
  mutate(Year = max(Year)) %>%
  group_by_(.dots = names(.)[!names(.) %in% c("Value")]) %>%
  summarise(Value = mean(Value)) %>%
  ungroup()
# Expanding projected emissions for the projected period
gf <-
  ldply((max(gf$Year) + 1):lastYear, function(x) {
    gf %>%
      mutate(Year = x)
  }) %>%
  tbl_df() %>%
  bind_rows(gf_sf) %>%
  mutate(d.source = "Outlook") %>%
  arrange(Domain, AreaCode, ItemCode, ElementCode, Year)
gf <-
  gf %>%
  mutate(d.source = "no adj. Outlook") %>%
  bind_rows(gf) %>%
  bind_rows(gf_sf)
```

Combining Land use total emissions

```
lu <-
  bind_rows(gf, ol_lu)  %>%
  agg_all_ol_regions() %>%
  join_names()
```

```
## Parsed with column specification:
## cols(
##   ItemCode = col_character(),
##   ItemName = col_character()
## )

## Parsed with column specification:
## cols(
##   ElementCode = col_character(),
##   ElementName = col_character(),
##   Unit = col_character()
## )
```

Exporting all results into file

Exporting data for the domain GT - Agriculture Total as in the FAOSTAT:

```
agTotal <- gt
```

Exporting data for the GL - Land Use Total

```
landuseTotal <- lu
```

Exporting the activity based data - data that was calculated based on the mapped activity data. It is data for four domains: gm, ge, gu, gp and gr.

```
activityBasedData <-
  bind_rows(list(gm, ge, gu, gp, gr)) %>%
  agg_total_emissions %>%
  join_names()
```

```
## Parsed with column specification:
## cols(
##   ItemCode = col_character(),
##   ItemName = col_character()
## )
```

```
## Parsed with column specification:
## cols(
##   ElementCode = col_character(),
##   ElementName = col_character(),
##   Unit = col_character()
## )
```

Structure of the exported data is similar:

```
str(activityBasedData)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    718798 obs. of  10 variables:
##  $ Domain     : chr  "GE" "GE" "GE" "GE" ...
##  $ AreaCode   : chr  "AF1" "AF1" "AF1" "AF1" ...
##  $ Year       : num  2000 2000 2000 2000 2000 2000 2000 2000 2000 2000 ...
##  $ ElementCode: chr  "Emissions_CO2Eq" "Emissions_CO2Eq" "Emissions_CO2Eq" "Emissions_CO2Eq" ...
##  $ d.source   : chr  "Faostat" "Faostat" "Faostat" "Faostat" ...
##  $ ItemCode   : chr  "BV" "EQ" "MK" "PK" ...
##  $ Value      : num  3281.87 251.03 678.78 5.79 775.72 ...
##  $ ItemName   : chr  "Cattle, non-dairy incl. Buffaloes" "Asses, Camels, Horses, Llamas and Mules" "(
##  $ ElementName: chr  "Emissions of all GHG in CO2 Equivalent" "Emissions of all GHG in CO2 Equivalent
##  $ Unit       : chr  "Gigagram" "Gigagram" "Gigagram" "Gigagram" ...
```

Important to notice that in any of these data frames, variable `d.source` is a categorical variable that describes the sources of data.

- Faostat - stands for the FAOSTAT data
- Outlook - stands for the outlook based data with the adjusted activity data, whenever it was not properly matching with the FAOSTAT
- no adj. Outlook - stands for the outlook based data with the not-adjusted activity data. # Annexes

Funcitons

To see all functions used in this work, please go to the folder "R" and see documentation in the source code. ## Mapping tabels from FAOSTAT countries to Outlook countries and regions

AreaCode	AreaName	OutlookAreaCode	OutlookAreaName

AreaCode	AreaName	OutlookAreaCode	OutlookAreaName
1	Armenia	ARM	Armenia
2	Afghanistan	AFG	Afghanistan
3	Albania	ALB	Albania
4	Algeria	DZA	Algeria
5	American Samoa	ASM	American Samoa
6	Andorra	AND	Andorra
7	Angola	AGO	Angola
8	Antigua and Barbuda	ATG	Antigua and Barbuda
9	Argentina	ARG	Argentina
10	Australia	AUS	Australia
11	Austria	E15	EU-15
12	Bahamas	BHS	Bahamas
13	Bahrain	BHR	Bahrain
14	Barbados	BRB	Barbados
15	Belgium-Luxembourg	NA	NA
16	Bangladesh	BGD	Bangladesh
17	Bermuda	NA	NA
18	Bhutan	BTN	Bhutan
19	Bolivia (Plurinational State of)	BOL	Bolivia
20	Botswana	BWA	Botswana
21	Brazil	BRA	Brazil
22	Aruba	ABW	Aruba
23	Belize	BLZ	Belize
24	British Indian Ocean Territory	NA	NA
25	Solomon Islands	SLB	Solomon Islands
26	Brunei Darussalam	BRN	Brunei Darussalam
27	Bulgaria	NMS	New Member States
28	Myanmar	MMR	Myanmar
29	Burundi	BDI	Burundi
30	Antarctica	NA	NA
32	Cameroon	CMR	Cameroon
33	Canada	CAN	Canada
35	Cabo Verde	CPV	Cape Verde
36	Cayman Islands	CYM	Cayman Islands
37	Central African Republic	CAF	Central African Republic
38	Sri Lanka	LKA	Sri Lanka
39	Chad	TCD	Chad
40	Chile	CHL	Chile
41	China, mainland	CHN	China
42	Christmas Island	CXR	Christmas Island
43	Cocos (Keeling) Islands	CCK	Cocos (Keeling) Islands
44	Colombia	COL	Colombia
45	Comoros	COM	Comoros
46	Congo	COG	Congo
47	Cook Islands	COK	Cook Islands
48	Costa Rica	CRI	Costa Rica
49	Cuba	CUB	Cuba
50	Cyprus	NMS	New Member States
51	Czechoslovakia	NA	NA
52	Azerbaijan	AZE	Azerbaijan, Republic of

AreaCode	AreaName	OutlookAreaCode	OutlookAreaName
53	Benin	BEN	Benin
54	Denmark	E15	EU-15
55	Dominica	DMA	Dominica
56	Dominican Republic	DOM	Dominican Republic
57	Belarus	BLR	Belarus
58	Ecuador	ECU	Ecuador
59	Egypt	EGY	Egypt
60	El Salvador	SLV	El Salvador
61	Equatorial Guinea	GNQ	Equatorial Guinea
62	Ethiopia PDR	NA	NA
63	Estonia	NMS	New Member States
64	Faroe Islands	FRO	Faeroe Islands
65	Falkland Islands (Malvinas)	FLK	Falkland Islands (Malvinas)
66	Fiji	FJI	Fiji Islands
67	Finland	E15	EU-15
68	France	E15	EU-15
69	French Guiana	GUF	French Guiana
70	French Polynesia	PYF	French Polynesia
71	French Southern and Antarctic Territories	NA	NA
72	Djibouti	DJI	Djibouti
73	Georgia	GEO	Georgia
74	Gabon	GAB	Gabon
75	Gambia	GMB	Gambia
79	Germany	E15	EU-15
80	Bosnia and Herzegovina	BIH	Bosnia and Herzegovina
81	Ghana	GHA	Ghana
82	Gibraltar	GIB	Gibraltar
83	Kiribati	KIR	Kiribati
84	Greece	E15	EU-15
85	Greenland	NA	NA
86	Grenada	GRD	Grenada
87	Guadeloupe	GLP	Guadeloupe
88	Guam	GUM	Guam
89	Guatemala	GTM	Guatemala
90	Guinea	GIN	Guinea
91	Guyana	GUY	Guyana
93	Haiti	HTI	Haiti
94	Holy See	VAT	Holy See
95	Honduras	HND	Honduras
96	China, Hong Kong SAR	HKG	Hong Kong SAR
97	Hungary	NMS	New Member States
98	Croatia	NMS	New Member States
99	Iceland	ISL	Iceland
100	India	IND	India
101	Indonesia	IDN	Indonesia
102	Iran (Islamic Republic of)	IRN	Iran (Islamic Republic of)
103	Iraq	IRQ	Iraq
104	Ireland	E15	EU-15
105	Israel	ISR	Israel
106	Italy	E15	EU-15
107	Côte d'Ivoire	CIV	Côte d'Ivoire
108	Kazakhstan	KAZ	Kazakhstan

AreaCode	AreaName	OutlookAreaCode	OutlookAreaName
109	Jamaica	JAM	Jamaica
110	Japan	JPN	Japan
112	Jordan	JOR	Jordan
113	Kyrgyzstan	KGZ	Kyrgyzstan
114	Kenya	KEN	Kenya
115	Cambodia	KHM	Cambodia
116	Democratic People's Republic of Korea	PRK	Democratic People's Republic of Korea
117	Republic of Korea	KOR	South Korea
118	Kuwait	KWT	Kuwait
119	Latvia	NMS	New Member States
120	Lao People's Democratic Republic	LAO	Lao People's Democratic Republic
121	Lebanon	LBN	Lebanon
122	Lesotho	LSO	Lesotho
123	Liberia	LBR	Liberia
124	Libya	LBY	Libyan Arab Jamahiriya
125	Liechtenstein	NA	NA
126	Lithuania	NMS	New Member States
127	Marshall Islands	MHL	Marshall Islands
128	China, Macao SAR	MAC	Macau SAR
129	Madagascar	MDG	Madagascar
130	Malawi	MWI	Malawi
131	Malaysia	MYS	Malaysia
132	Maldives	MDV	Maldives
133	Mali	MLI	Mali
134	Malta	NMS	New Member States
135	Martinique	MTQ	Martinique
136	Mauritania	MRT	Mauritania
137	Mauritius	MUS	Mauritius
138	Mexico	MEX	Mexico
140	Monaco	MCO	Monaco
141	Mongolia	MNG	Mongolia
142	Montserrat	MSR	Montserrat
143	Morocco	MAR	Morocco
144	Mozambique	MOZ	Mozambique
145	Micronesia (Federated States of)	FSM	Micronesia (Federated States of)
146	Republic of Moldova	MDA	Moldova, Republic of
147	Namibia	NAM	Namibia
148	Nauru	NRU	Nauru
149	Nepal	NPL	Nepal
150	Netherlands	E15	EU-15
151	Netherlands Antilles (former)	ANT	Netherlands Antilles
153	New Caledonia	NCL	New Caledonia
154	The former Yugoslav Republic of Macedonia	MKD	The former Yugoslav Republic of Macedo
155	Vanuatu	VUT	Vanuatu
156	New Zealand	NZL	New Zealand
157	Nicaragua	NIC	Nicaragua
158	Niger	NER	Niger
159	Nigeria	NGA	Nigeria
160	Niue	NIU	Niue
161	Norfolk Island	NFK	Norfolk Island
162	Norway	NOR	Norway
163	Northern Mariana Islands	MNP	Northern Mariana Islands

AreaCode	AreaName	OutlookAreaCode	OutlookAreaName
164	Pacific Islands Trust Territory	PIT	Pacific Islands
165	Pakistan	PAK	Pakistan
166	Panama	PAN	Panama
167	Czech Republic	NMS	New Member States
168	Papua New Guinea	PNG	Papua New Guinea
169	Paraguay	PRY	Paraguay
170	Peru	PER	Peru
171	Philippines	PHL	Philippines
172	Pitcairn Islands	PCN	Pitcairn Islands
173	Poland	NMS	New Member States
174	Portugal	E15	EU-15
175	Guinea-Bissau	GNB	Guinea-Bissau
176	Timor-Leste	TLS	Timor-Leste
177	Puerto Rico	PRI	Puerto Rico
178	Eritrea	ERI	Eritrea
179	Qatar	QAT	Qatar
180	Palau	PLW	Palau
181	Zimbabwe	ZWE	Zimbabwe
182	Réunion	REU	Réunion
183	Romania	NMS	New Member States
184	Rwanda	RWA	Rwanda
185	Russian Federation	RUS	Russia
186	Serbia and Montenegro	SRM	Serbia+Montenegro
187	Saint Helena, Ascension and Tristan da Cunha	SHN	Saint Helena
188	Saint Kitts and Nevis	KNA	Saint Kitts and Nevis
189	Saint Lucia	LCA	Saint Lucia
190	Saint Pierre and Miquelon	NA	NA
191	Saint Vincent and the Grenadines	VCT	Saint Vincent and the Grenadines
192	San Marino	SMR	San Marino
193	Sao Tome and Principe	STP	Sao Tome and Principe
194	Saudi Arabia	SAU	Saudi Arabia
195	Senegal	SEN	Senegal
196	Seychelles	SYC	Seychelles
197	Sierra Leone	SLE	Sierra Leone
198	Slovenia	NMS	New Member States
199	Slovakia	NMS	New Member States
200	Singapore	SGP	Singapore
201	Somalia	SOM	Somalia
202	South Africa	ZAF	South Africa
203	Spain	E15	EU-15
205	Western Sahara	ESH	Western Sahara
206	Sudan (former)	SDN	Sudan
207	Suriname	SUR	Suriname
208	Tajikistan	TJK	Tajikistan
209	Swaziland	SWZ	Swaziland
210	Sweden	E15	EU-15
211	Switzerland	CHE	Switzerland
212	Syrian Arab Republic	SYR	Syrian Arab Republic
213	Turkmenistan	TKM	Turkmenistan
214	China, Taiwan Province of	TWN	Taiwan Province of China
215	United Republic of Tanzania	TZA	United Republic of Tanzania
216	Thailand	THA	Thailand

AreaCode	AreaName	OutlookAreaCode	OutlookAreaName
217	Togo	TGO	Togo
218	Tokelau	TKL	Tokelau
219	Tonga	TON	Tonga
220	Trinidad and Tobago	TTO	Trinidad and Tobago
221	Oman	OMN	Oman
222	Tunisia	TUN	Tunisia
223	Turkey	TUR	Turkey
224	Turks and Caicos Islands	TCA	Turks and Caicos Islands
225	United Arab Emirates	ARE	United Arab Emirates
226	Uganda	UGA	Uganda
227	Tuvalu	TUV	Tuvalu
228	USSR	NA	NA
229	United Kingdom	E15	EU-15
230	Ukraine	UKR	Ukraine
231	United States of America	USA	United States
233	Burkina Faso	BFA	Burkina Faso
234	Uruguay	URY	Uruguay
235	Uzbekistan	UZB	Uzbekistan
236	Venezuela (Bolivarian Republic of)	VEN	Venezuela
237	Viet Nam	VNM	Viet Nam
238	Ethiopia	ETH	Ethiopia
239	British Virgin Islands	VGB	British Virgin Islands
240	United States Virgin Islands	VIR	United States Virgin Islands
242	Wake Island	NA	NA
243	Wallis and Futuna Islands	WLF	Wallis and Futuna Islands
244	Samoa	WSM	Samoa
248	Yugoslav SFR	NA	NA
249	Yemen	YEM	Yemen
250	Democratic Republic of the Congo	COD	Democratic Republic of the Congo
251	Zambia	ZMB	Zambia
252	Unspecified	NA	NA
255	Belgium	E15	EU-15
256	Luxembourg	E15	EU-15
258	Anguilla	AIA	Anguilla
259	Channel Islands	NA	NA
260	Svalbard and Jan Mayen Islands	NA	NA
264	Isle of Man	NA	NA
270	Mayotte	MYT	Mayotte
272	Serbia	SRB	Serbia
273	Montenegro	MNE	Montenegro
276	Sudan	NA	NA
277	South Sudan	NA	NA
279	Curaçao	NA	NA
280	Sint Maarten (Dutch Part)	NA	NA
281	Saint-Martin (French Part)	NA	NA
299	Occupied Palestinian Territory	PSE	Palestine, Occupied Tr.
351	China	NA	NA
357	China (exc. Hong Kong & Macao)	NA	NA
5000	World	WLD	World

Mapping tabel for mapping FAOSTAT items to the Outlook

ItemCode	ItemName	OutlookItemCode	ItemCodeAggSign	Exist
27	Rice	RI	+	TRUE
1051	Swine, breeding	PK	+	TRUE
1049	Swine, market	PK	+	TRUE
976	Sheep	SH	+	TRUE
1016	Goats	SH	+	TRUE
961	Cattle, non-dairy	BV	+	TRUE
946	Buffaloes	BV	+	TRUE
960	Cattle, dairy	MK	+	TRUE
1107	Asses	EQ	+	FALSE
1126	Camels	EQ	+	FALSE
1096	Horses	EQ	+	FALSE
1177	Llamas	EQ	+	FALSE
1110	Mules	EQ	+	FALSE
1053	Chickens, broilers	PT	+	TRUE
1052	Chickens, layers	PT	+	TRUE
1068	Ducks	PT	+	TRUE
1079	Turkeys	PT	+	TRUE
5058	Enteric Fermentation	GE	+	TRUE
5059	Manure Management	GM	+	TRUE
5060	Rice Cultivation	GR	+	TRUE
5062	Manure applied to Soils	GU	+	TRUE
5063	Manure left on Pasture	GP	+	TRUE
5064	Crop Residues	GA	+	FALSE
6759	Cultivation of Organic Soils	GV	+	FALSE
5066	Burning - Crop residues	GB	+	FALSE
5067	Burning - Savanna	GH	+	FALSE
5061	Synthetic Fertilizers	GY	+	FALSE
6798	Burning - all categories	BA	+	TRUE
5070	Cropland	GC	+	TRUE
6794	Grassland	GG	+	TRUE
5069	Burning Biomass	GI	+	TRUE
5065	Forest land	GF	+	TRUE
6661	Forest	FO	+	TRUE
6750	Net Forest conversion	FC	+	TRUE

Mapping tabel for mapping FAOSTAT elements to the Outlook

Domain	ItemCode	ElementCode	OutlookElementCode	OutlookAdjustment
GR	RI	5312	AH	0.001
GR	NA	72255	EM_CH4	1.000
GR	NA	72315	EM_CH4Eq	1.000
GE	MK	5111	CI	0.001
GE	BV	5111	LI	0.001
GE	SH	5111	LI	0.001
GE	PK	5111	LI	0.001
GE	NA	72254	EM_CH4	1.000
GE	NA	72314	EM_CH4Eq	1.000
GM	MK	5111	CI	0.001

Domain	ItemCode	ElementCode	OutlookElementCode	OutlookAdjustment
GM	BV	5111	LI	0.001
GM	SH	5111	LI	0.001
GM	PK	5111	LI	0.001
GM	PT	5111	LI	0.001
GM	NA	72256	EM_CH4	1.000
GM	NA	72446	EM_CH4Eq	1.000
GM	NA	72306	EM_N2O	1.000
GM	NA	72436	EM_N2OEq	1.000
GU	MK	5111	CI	0.001
GU	BV	5111	LI	0.001
GU	SH	5111	LI	0.001
GU	PK	5111	LI	0.001
GU	PT	5111	LI	0.001
GU	NA	72301	EM_N2O	1.000
GU	NA	72311	EM_N2OEq	1.000
GP	MK	5111	CI	0.001
GP	BV	5111	LI	0.001
GP	SH	5111	LI	0.001
GP	PK	5111	LI	0.001
GP	PT	5111	LI	0.001
GP	NA	72300	EM_N2O	1.000
GP	NA	72310	EM_N2OEq	1.000
GT	NA	7231	Emissions_CO2Eq	1.000
GT	NA	7244	EM_CH4Eq	1.000
GT	NA	7243	EM_N2OEq	1.000
GL	NA	7233	EM_CO2	1.000
GL	NA	7217	Emissions_CO2Eq	1.000
GL	NA	7243	EM_N2OEq	1.000
GL	NA	7244	EM_CH4Eq	1.000
GF	NA	72332	EM_CO2	1.000
GF	NA	72172	Emissions_CO2Eq	1.000
GF	NA	5110	Area	1.000