

Ultra-processed foods and recommended intake levels of nutrients linked to non-communicable diseases in Australia: evidence from a nationally representative cross-sectional study

Supplementary Materials

Appendix 1. Manual of the application of NOVA system to the 2011-13 AUSNUT database

A novel processed food classification system applied to disaggregated food codes in the Australian Food Nutrient Database 2011-13

-- Manual --

Presentation

This project is part of the thematic project "Consumption of ultra-processed foods, dietary nutrient profile and obesity in seven countries". The data source is the National Nutrition and Physical Activity Survey (NNPAS), conducted in 2011-12. This survey studied a random sample of the Australian population aged 2+ years (n=12,153).

Food consumption data were collected through two 24-h recalls. All foods and beverages purchased by households (after the exclusion of non-edible parts) were converted into energy and nutrients using data from the Australian Food Nutrient Database (AUSNUT) 2011-13 database.

Food items and/or the underlying ingredients were classified according to NOVA food classification system into the following four groups (and subgroups within these groups): Group 1 - Unprocessed or minimally processed foods (e.g. rice and other cereals, meat, fish, milk, eggs, fruit, roots and tubers, vegetables, nuts and seeds); Group 2 - Processed culinary ingredients (e.g. sugar, plant oils and butter); Group 3 - Processed foods (e.g. processed breads and cheese, canned fruit and fish, and salted and smoked meats); Group 4 - Ultra-processed foods (e.g. confectionaries, savoury snacks, fast food dishes, mass-produced packaged breads, frozen and ready meals and soft drinks). (Monteiro et al., 2018). The rationale underlying the classification is described elsewhere (Monteiro et al., 2016, 2018¹).

¹Monteiro, C. A.; Cannon, G.; Levy, R. B.; Moubarac, J-C.; Jaime, P.; Martins, A. P.; Canella, D.; Louzada, M. L.; Parra, D.; with Ricardo, C.; Calixto, G.; Machado, P.; Martins, C.; Martinez, E.; Baraldi, L.; Garzillo, J.; Sattamini, I. Nova. The star shines bright. [Food classification. Public health]. *World Nutrition*, v. 7, n. (1-3), p. 28-38, 2016. Monteiro, C. A.; Cannon, G.; Moubarac, J. C.; Levy, R. B.; Louzada, M. L. C.; Jaime, P. C. The UN Decade of Nutrition, the NOVA food classification and the trouble with ultra-processing. *Public Health Nutrition*, v. 21, n. 1, p. 5-17, 2018.

1 About AUSNUT 2011-13

AUSNUT 2011–13² is a set of files that enables food, dietary supplement and nutrient intake estimates to be made from the 2011–13 Australian Health Survey (AHS). It includes foods and dietary supplements consumed as part of the 2011–12 National Nutrition and Physical Activity Survey (NNPAS) and the 2012–13 National Aboriginal and Torres Strait Islander Nutrition and Physical Activity Survey (NATSINPAS) components of the AHS.

The complete AUSNUT 2011–13 contains the following 11 files:

Food details file – descriptive information on 5,740 foods and beverages to help users to identify where the nutrient data come from

Food composition database – 53 nutrient values for each of the 5,740 foods and beverages

Food recipe file – ingredient information on the 3,650 food and beverage recipes

Food retention factor file – information on the recipe factors

Food measures database – 16,152 measures for the 5,740 foods and beverages

Dietary supplement details file – descriptive information on 2,163 dietary supplements

Dietary supplement nutrient database – 35 nutrient values for each of the 2,163 dietary supplements

Dietary supplement recipe file – ingredient information on 18 dietary supplement recipes

Food and dietary supplement classification system

1995 NNS and 2011–13 AHS food classification concordance file

AUSNUT 2011–13 – AUSNUT 1999 Matching File

To apply NOVA system into the 2011-13 AUSNUT dataset, we used the following files: a) Food details file, b) Food composition database, c) Food recipe file.

²FSANZ. Food Standards Australia New Zealand, 2014. AUSNUT 2011-2013 - Food Composition Database. Available at: <<http://www.foodstandards.gov.au/science/monitoringnutrients/ausnut/pages/default.aspx>>.

2 Applying the NOVA system to the 2011-13 AUSNUT database

1st step: Identifying handmade recipes

We first combined the 'Food recipe file' with the 'Food composition database'. We also used the food name to identify handmade recipes. We identified 10 different sources of foods, which are presented below, including the decision about the disaggregation.

Food source		Examples	Decision
1	Homemade	13404004 Quiche, Lorraine, homemade	Apply recipe - disaggregate
2	Commercial	13101004 Biscuit, sweet, chocolate flavoured, commercial	Packaged foods do not disaggregate
		24904030 Salad, tabouleh, commercial	Potentially made from a commercial kitchen/ restaurant, disaggregate
3	Fast food	13505004 Chicken burger, white roll, chicken breast, with bacon, cheese, egg & sauce, fast food chain-style	Do not disaggregate
4	Takeaway	13503048 Kebab wrap, falafel, with salad & sauce, takeaway	Apply recipe - disaggregate
5	Purchased frozen	13502003 Pizza, cheese & tomato, purchased frozen, baked	Do not disaggregate
6	Ready to eat	13405008 Pastry, filled with spinach, commercial, ready to eat	Do not disaggregate
7	Cafe or restaurant	21602001 Soup, lentil or other legumes, from cafe or restaurant	Apply recipe - disaggregate
8	Dry (powder):	21201008 Soup, pea & ham, cup of soup, instant dry mix	Apply recipe – disaggregate (considering reconstitution)
9	Canned/ Preserved	24501006 Pea, green, canned in brine, cooked	Do not disaggregate
0	Unspecified	19401017 Cheese, haloumi	Packaged products do not disaggregate
		11202003 Coffee, cappuccino, from ground coffee beans, with regular fat cows milk	Potentially handmade, disaggregate

A total of 2,585 (45%) food codes were subject to disaggregation and this process was continued until all ingredients were single food items. For food codes with no recipe in the AUSNUT Food recipe file, we applied recipes of similar foods. The cases and the recipes applied to these foods is presented below.

Food code	Food description	Recipe used to replace
31304004	Stock, liquid, fish, homemade from basic ingredients	<i>Food code:</i> 31304003 Stock, liquid, fish, commercial <i>Ingredients:</i> 11701005 Water, tap 15101118 Fish, white flesh, raw, not further defined 31301007 Salt, table, non-iodised 11301015 Juice, lemon, commercial 27101006 Sugar, white, granulated or lump 24802009 Onion, mature, brown skinned, peeled, raw 31302017 Pepper, ground, black or white 29202005 Wine, white, cooked
31304002	Stock, liquid, all flavours (except fish), homemade from basic ingredients	There is no similar recipe (only with fish as mentioned before. We applied the previous recipe: <i>Food code:</i> 31304003 Stock, liquid, fish, commercial.
21102004	Soup, minestrone, prepared, from cafe or restaurant	<i>Food code:</i> 21102008 Soup, mixed vegetable & legume, with pasta or grains, prepared with stock or water, homemade from basic ingredients <i>Ingredients:</i> 24802013 Onion, mature, peeled, fresh or frozen, baked, roasted, 24802002 Garlic, peeled or unpeeled, fresh or frozen, baked, roast 24101049 Potato, peeled, raw, not further defined 24701013 Pumpkin, peeled, fresh or frozen, raw 24301005 Carrot, mature, peeled or unpeeled, fresh or frozen, raw 24702007 Zucchini, green skin, fresh or frozen, peeled or unpeel 24502005 Bean, green, fresh or frozen, raw 24602006 Tomato, whole, canned in tomato juice, undrained 31304001 Stock, liquid, all flavours (except fish), prepared from c 24401013 Spinach, fresh, raw 24403006 Herbs, mixed, raw 14602003 Fat or oil, not further defined, for use in home cooked r 31302017 Pepper, ground, black or white 12101013 Grains & pasta, for homemade soup recipes 25101020 Legumes, mixed, for use in homemade soup recipes
18701014	Curry, commercial, beef, vindaloo sauce	<i>Food code:</i> 18701013 Curry, commercial, beef, tomato based sauce <i>Ingredients:</i> 18101025 Beef, casserole cuts, fully-trimmed, boiled, casseroled, 24802019 Onion, mature, peeled, fresh or frozen, boiled, microwa 24802003 Garlic, peeled or unpeeled, fresh or frozen, boiled, micr 24302021 Ginger, peeled, fresh or frozen, boiled, microwaved or 14403001 Oil, not further defined 24602008 Tomato, whole, canned in tomato juice, boiled or micro 23109002 Paste, curry, Indian style, commercial 31301007 Salt, table, non-iodised 14102001 Ghee, clarified butter 24803021 Mixed vegetables, cooked, for use in all other curry rec

Food code	Food description	Recipe used to replace
18705007	Curry, commercial, lamb, dairy based sauce	<p><i>Food code:</i> 18705009 Curry, commercial, lamb & vegetable, dairy based sauce</p> <p><i>Ingredients:</i></p> <p>24802019 Onion, mature, peeled, fresh or frozen, boiled, microwaved</p> <p>24802003 Garlic, peeled or unpeeled, fresh or frozen, boiled, microwaved</p> <p>24302021 Ginger, peeled, fresh or frozen, boiled, microwaved or microwaved</p> <p>14403001 Oil, not further defined</p> <p>24602008 Tomato, whole, canned in tomato juice, boiled or microwaved</p> <p>19201004 Yoghurt, natural, regular fat (~4%)</p> <p>22204006 Nut, almond, without skin, blanched, unsalted</p> <p>23109002 Paste, curry, Indian style, commercial</p> <p>14102001 Ghee, clarified butter</p> <p>18102013 Lamb, casserole cuts, semi-trimmed, boiled, casseroled</p> <p>31301007 Salt, table, non-iodised</p> <p>24803021 Mixed vegetables, cooked, for use in all other curry recipes</p>
18705020	Tandoori lamb, commercial	<p><i>Food code:</i> 18901031 Tandoori chicken, homemade, with commercial paste & yoghurt</p> <p><i>Ingredients:</i></p> <p>18301030 Chicken, fillet or kebab, flesh, raw</p> <p>23109002 Paste, curry, Indian style, commercial</p> <p>19201002 Yoghurt, Greek style (~8%), natural</p> <p>24802002 Garlic, peeled or unpeeled, fresh or frozen, baked, roasted</p> <p>11302004 Juice, lemon, home squeezed</p> <p>14403001 Oil, not further defined</p> <p>24802012 Onion, mature, peeled, fresh or frozen, raw, not further processed</p>

2st step: Applying NOVA system into food codes or underlying ingredients after full disaggregation

Two experts with Australian food and dietary intake knowledge applied the NOVA system to the AUSNUT 2011-13. All classifications were checked by another two independent food assessment experts and where classification discrepancies arose, these were discussed until consensus was reached amongst all researchers. We applied NOVA system in the ingredient fully disaggregated level, considering:

- The AUSNUT classification system that considers a major (two-digit), sub-major (three-digit), minor (five-digit) food group or the survey ID (eight-digit). For instance, the sub-major (three-digit) was used to classify all soft drinks (e.g. 115 - Soft drinks, and flavoured mineral waters) in the ultra-processed food group. The minor (five-digit) was used to classify preserved fruits (e.g. 16104 –Pears, commercially sterile) in the processed food group, while the survey ID (eight-digit) was used to identify exceptions (e.g. 16104011 Pear, canned in intense sweetened liquid, drained – classified as ultra-processed food)

- b) The source of food, for non-disaggregated food codes. All fast foods and meals purchased frozen were classified in the ultra-processed food groups, and most of the canned/preserved foods were classified in the processed food group
- c) Information present in the list of ingredients of food items sold in the Australian supermarkets websites (Coles and Woolworths). These supermarket chains hold a 70% of market share in Australia (Roy Morgan Research, 2017³)
- d) Previous publication applying the NOVA system into non-disaggregated AUSNUT 2011-13 (O'Halloran et al., 2017⁴)

When the classification of a food item was not clear (e.g. cake or cupcake, honey, commercial or homemade), the conservative alternative was chosen (homemade in this case, and thus disaggregated).

Breads, frequently consumed by the Australian population, may be either processed or ultra-processed foods. We first identified the processed breads, and these included: focaccia, ciabatta, baguette, pane di casa, sour dough, flats (naan, paratha, chapatti, roti, injera, and pita), pumpkin bread, corn bread and tortillas. The ultra-processed breads identified were: bagel, breadcrumbs, hot dog breads, fast food breads, pizza bases, all light breads and with addition of fibre, vitamins and minerals.

Of the 62 generic bread codes where the NOVA classification was not easily apparent, there were two generic bread codes that contributed the most to total bread energy intake (25% combined): i) Bread, from white flour, commercial, ii) Bread roll, from white flour, commercial. According to the AUSNUT 'Food details file', for the first bread code (i), proximates, vitamins, minerals, and fatty acids were derived from of *8 samples of Bakers Delight white bread purchased nationally in 2008 (2008 KFP) and a composite or individual analysis of 10 purchases of 6 brands (4x Mighty Soft, 2x Helgas, 1x Sunblest, 1x Foodland, 1x Super Soft, 1x Master White) of white bread purchased nationally in each state and territory capital city and two regional centres in NSW and QLD (2006 - KFP)*. As the samples were mostly of mass-produced branded breads, based on observation of bread types across a range of supermarkets websites and other retailers in Melbourne, this bread code was classified as ultra-processed. For the second bread code (ii), *the majority of nutrient data including minerals, moisture, fibre, protein, starch, B1, B3 and pantothenate were derived from individual analysis of 10 purchases of 5 brands of bread rolls (4x Coles, 2x Woolworths, 2x TipTop, 1x IGA, 1x local bakery) purchased nationally in each state and territory capital city and 2 regional centres in NSW and QLD (2006 - KFP)*. Fat, ash, sugars, tryptophan and vitamins were determined in a

³Roy Morgan Research, 2017. Aldi hits new high in supermarket wars. Available at: <http://www.roymorgan.com/findings/7234-woolworths-coles-aldi-iga-supermarket-market-shares-australia-march-2017-201705171406>

⁴O'Halloran, S.A.; Lacy, K.E.; Grimes, C.A.; Woods, J.; Campbell, K.J.; Nowson, C.A. A novel processed food classification system applied to Australian food composition databases. *Journal of Human Nutrition and Dietetics*, 2017.

composite of the 10 purchases in the same program. As the samples were mostly of simple unpackaged breads, based on observation of bread types across Coles and Woolworths supermarkets in Melbourne, this bread code was classified as processed bread.

All the remaining breads were classified as processed breads as the conservative approach. Sensitivity analyses were carried out to account for exposure misclassification of breads and are presented in the previous section (Appendix 1).

The final application of NOVA system to the groups and subgroups of the 2011-13 AUSNUT dataset is presented further in the document *Food Group Code Classified* (Appendix 3).

The groups and subgroups of NOVA food classification system applied to the AUSNUT 2011-13 is presented below. For the analyses, we aggregated the following subgroups:

Red meat and poultry: Red meat (subgroup 4) + Poultry (5)

Fruits: Fruits (2) + Freshly squeezed juices (47)

Frozen and shelf stable ready meals: Packaged ready meals (26) + Industrial French fries (33) + Frozen pizza (38) + Instant and canned soups (40)

Pastries, buns, and cakes: Pastries, buns, and cakes (32) + Industrial desserts (39)

Other ultra-processed foods: Other ultra-processed foods (45) + Ultra-processed cheese (44)

Groups and subgroups of NOVA food classification system applied to the AUSNUT 2011-13/ NNPAS 2011-12

Group	Subgroup	
1		Unprocessed or minimally processed foods
	1	Milk and plain yoghurt
	2	Fruits
	3	Potatoes and other tubers and roots
	4	Red meat
	5	Poultry
	6	Cereals
	7	Pasta
	8	Vegetables
	9	Eggs
	10	Nuts and seeds
	11	Fish
	12	Legumes
	13	Other unprocessed or minimally processed foods ^a
47	Freshly squeezed juices	
2		Processed culinary ingredients
	14	Salt
	15	Table sugar
	16	Animal fats
	17	Plant oil
18	Other processed culinary ingredients ^b	
3		Processed foods
	19	Beer and wine
	20	Cheese
	21	Vegetables and other plant foods preserved in brine
	22	Processed breads
	23	Bacon and other salted, smoked or canned meat or fish
24	Other processed foods ^c	
4		Ultra-processed foods
	25	Mass-produced packaged breads
	26	Packaged ready meals
	27	Breakfast cereals
	28	Sausage and other reconstituted meat products
	29	Confectionary
	30	Ice cream, ice pops and frozen yogurts
	31	Biscuits
	32	Pastries, buns, and cakes
	33	Industrial French fries
	34	Margarine and other spreads
	35	Milk-based drinks
36	Sauces, dressing and gravies	
37	Packaged salty snacks	
38	Frozen pizza	
39	Industrial desserts	
40	Instant and canned soups	
41	Soft drinks, carbonated	
42	Fruit drinks and iced teas	
43	Alcoholic distilled drinks	
44	Ultra-processed cheese	
45	Other ultra-processed foods ^d	
46	Fast food dishes	

a Including non presweetened, non-flavored coffee and tea; coconut water; b Including vinegar; baking powder and baking soda; c Including salted or sugared nuts and seeds; peanut, sesame, cashew; d Including soya products such as meatless patties and fish sticks; baby food and baby formula.

Appendix 2. Routine of the application of NOVA system into the disaggregated items of the 2011-13 AUSNUT database (for use in Stata software)

tostring ingcode, replace

gen digcode3_ing = substr(ingcode, 1, 3)

gen digcode5_ing = substr(ingcode, 1, 5)

destring ingcode digcode3_ing digcode5_ing, replace

gen nova_group = .

gen subgroups = .

* 11 Non-alcoholic beverages

* 111 Tea

* 11101 Tea, regular, caffeinated, prepared with water

replace nova_group = 1 if digcode5_ing == 11101

replace subgroups = 13 if digcode5_ing == 11101

* 11102 Tea, regular, caffeinated, prepared with milk or milk substitute

replace nova_group = 1 if digcode5_ing == 11102

replace subgroups = 13 if digcode5_ing == 11102

* 11103 Tea, regular, decaffeinated, prepared with water or milk

replace nova_group = 1 if digcode5_ing == 11103

replace subgroups = 13 if digcode5_ing == 11103

* 11104 Tea mixed with other foods

replace nova_group = 4 if digcode5_ing == 11104

replace subgroups = 42 if digcode5_ing == 11104

* 11105 Herbal tea

replace nova_group = 1 if digcode5_ing == 11105

replace subgroups = 13 if digcode5_ing == 11105

* 11106 Tea powders and bases

replace nova_group = 4 if digcode5_ing == 11106

replace subgroups = 42 if digcode5_ing == 11106

* 112 Coffee and coffee substitutes

* 11201 Coffee beverage, prepared with water

replace nova_group = 1 if digcode5_ing == 11201

replace subgroups = 13 if digcode5_ing == 11201

* 11202 Coffee beverage, prepared with milk or milk substitute

replace nova_group = 1 if digcode5_ing == 11202

replace subgroups = 13 if digcode5_ing == 11202

* 11203 Coffee beverage, decaffeinated, prepared with water

replace nova_group = 1 if digcode5_ing == 11203

replace subgroups = 13 if digcode5_ing == 11203

* 11204 Coffee beverage, decaffeinated, prepared with milk or milk substitute

replace nova_group = 1 if digcode5_ing == 11204

replace subgroups = 13 if digcode5_ing == 11204

* 11205 Dry coffee powder, caffeinated or decaffeinated

```
replace nova_group = 1 if digcode5_ing == 11205
replace subgroups = 13 if digcode5_ing == 11205
*    11206 Coffee substitutes, beverage
replace nova_group = 4 if digcode5_ing == 11206
replace subgroups = 42 if digcode5_ing == 11206
*    11207 Coffee substitutes, powders and bases
replace nova_group = 4 if digcode5_ing == 11207
replace subgroups = 42 if digcode5_ing == 11207
*    11208 Coffee-based mixes, beverage
replace nova_group = 4 if digcode5_ing == 11208
replace subgroups = 42 if digcode5_ing == 11208
*    11209 Dry or concentrate coffee-based mixes
replace nova_group = 4 if digcode5_ing == 11209
replace subgroups = 42 if digcode5_ing == 11209

*    113 Fruit and vegetable juices, and drinks
*    11301 Fruit juices, commercially prepared
replace nova_group = 4 if digcode5_ing == 11301
replace subgroups = 42 if digcode5_ing == 11301
*    11302 Fruit juices, freshly-squeezed
replace nova_group = 1 if digcode5_ing == 11302
replace subgroups = 47 if digcode5_ing == 11302
*    11303 Fruit juices, fortified
replace nova_group = 4 if digcode5_ing == 11303
replace subgroups = 42 if digcode5_ing == 11303
*    11304 Vegetable juices
replace nova_group = 4 if digcode5_ing == 11304
replace subgroups = 42 if digcode5_ing == 11304
*Exception (aloe vera, carrot and celery)
replace nova_group = 1 if (ingcode == 11304001 | ingcode == 11304002 | ingcode == 11304003)
replace subgroups = 47 if (ingcode == 11304001 | ingcode == 11304002 | ingcode == 11304003)
*    11305 Vegetable juices, freshly squeezed
replace nova_group = 1 if digcode5_ing == 11305
replace subgroups = 47 if digcode5_ing == 11305
*    11306 Fruit and vegetable juice blends
replace nova_group = 4 if digcode5_ing == 11306
replace subgroups = 42 if digcode5_ing == 11306
*Exception (home squeezed)
replace nova_group = 1 if ingcode == 11306003
replace subgroups = 47 if ingcode == 11306003
*    11307 Fruit drinks (ready to drink or made from concentrate)
replace nova_group = 4 if digcode5_ing == 11307
replace subgroups = 42 if digcode5_ing == 11307
*    11308 Vegetable drinks
replace nova_group = 4 if digcode5_ing == 11308
replace subgroups = 42 if digcode5_ing == 11308
*    11309 Fruit drink, prepared from dry powder
replace nova_group = 4 if digcode5_ing == 11309
replace subgroups = 42 if digcode5_ing == 11309

*    114 Cordials
replace nova_group = 4 if digcode3_ing == 114
replace subgroups = 42 if digcode3_ing == 114
```

```

*      11401 Cordials, made from concentrate
*      11402 Cordials, made from concentrate, intense sweetened
*      11403 Cordial concentrate
*      11404 Cordial concentrate, intense sweetened

*      115      Soft drinks, and flavoured mineral waters
replace nova_group = 4 if digcode3_ing == 115
replace subgroups = 41 if digcode3_ing == 115
*      11501 Soft drinks, non-cola
*      11502 Soft drinks, non-cola, intense sweetened
*      11503 Soft drinks, cola
*      11504 Soft drinks, cola, intense sweetened
*      11505 Flavoured mineral waters
*      11506 Flavoured mineral waters, intense sweetened

*      116      Electrolyte, energy and fortified drinks
replace nova_group = 4 if digcode3_ing == 116
replace subgroups = 42 if digcode3_ing == 116
*      11601 Electrolyte drinks (sports drinks)
*      11602 Electrolyte drink bases (sport drink bases)
*      11603 Energy drinks
*      11604 Energy drinks, intense sweetened

*      117      Waters, municipal and bottled, unflavoured
*      11701 Domestic water (including tap, tank/rain water)
replace nova_group = 1 if digcode5_ing == 11701
replace subgroups = 13 if digcode5_ing == 11701
*      11702 Purchased packaged water including mineral water
replace nova_group = 1 if digcode5_ing == 11702
replace subgroups = 13 if digcode5_ing == 11702
*      11703 Purchased packaged water, fortified
replace nova_group = 4 if digcode5_ing == 11703
replace subgroups = 42 if digcode5_ing == 11703

*      118      Other beverage flavourings and prepared beverages
replace nova_group = 4 if digcode3_ing == 118
replace subgroups = 42 if digcode3_ing == 118
*      11801 Fortified beverage flavourings prepared with water or milk
*      11802 Fortified dry beverage flavourings
*      11803 Unfortified beverage flavourings prepared with water or milk
*      11804 Unfortified dry beverage flavourings
*      11805 Breakfast cereal beverages
*      11806 Other beverages

*****
*      12      Cereals and cereal products
*****

*      121      Flours and other cereal grains and starches
*      12101 Grains (other than rice) and grain fractions
replace nova_group = 1 if digcode5_ing == 12101
replace subgroups = 6 if digcode5_ing == 12101
*      12102 Rice and rice grain fractions
replace nova_group = 1 if digcode5_ing == 12102

```

```

replace subgroups = 6 if digcode5_ing == 12102
  *Exception (rice, flavoured, from dry mix)
replace nova_group = 4 if (ingcode == 12102010 | ingcode == 12102011)
replace subgroups = 40 if (ingcode == 12102010 | ingcode == 12102011)
*   12103 Cereal flours and starches
replace nova_group = 1 if digcode5_ing == 12103
replace subgroups = 6 if digcode5_ing == 12103
  *Exception (custard powder and rice paper)
replace nova_group = 4 if ingcode == 12103004
replace subgroups = 40 if ingcode == 12103004
replace nova_group = 3 if ingcode == 12103017
replace subgroups = 24 if ingcode == 12103017
*   12104 Cereal flours and starches, fortified
replace nova_group = 1 if digcode5_ing == 12104
replace subgroups = 6 if digcode5_ing == 12104

*   122 Regular breads, and bread rolls (plain/unfilled/untopped varieties)
*   12201 Breads, and bread rolls, white, mandatorily fortified
*   12202 Breads, and bread rolls, white, additional voluntary fortification
*   12203 Breads, and bread rolls, white, not stated as to fortification
*   12204 Breads, and bread rolls, mixed grain, mandatorily fortified
*   12205 Breads, and bread rolls, mixed grain, additional voluntary fortification
*   12206 Breads, and bread rolls, mixed grain, not stated as to fortification
*   12207 Breads, and bread rolls, wholemeal and brown, mandatorily fortified
*   12208 Breads, and bread rolls, wholemeal and brown, additional voluntary fortification
*   12209 Breads, and bread rolls, wholemeal, not stated as to fortification
*   12210 Breads, and bread rolls, rye, mandatorily fortified
*   12211 Breads, and bread rolls, rye, additional voluntary fortification
*   12212 Breads, and bread rolls, rye, not stated as to fortification
*   12213 Breads, and bread rolls, gluten free
*   12214 Breads, and bread rolls, not stated as to major flour or fortification

*Processed breads: focaccia, ciabatta, baguette, damper, pane di casa, sour dough, flats (naan, paratha,
chapatti, roti, injera, pita), pumpkin bread, corn bread and tortillas
replace nova_group = 3 if (ingcode == 12201005 | ingcode == 12201006 | ingcode == 12201007 |
ingcode == 12201008 | ingcode == 12201017 | ingcode == 12201018 | ingcode == 12201019 |
ingcode == 12201020 | ingcode == 12201021 | ingcode == 12201022 | ingcode == 12201023 |
ingcode == 12201024 | ingcode == 12201028 | ingcode == 12201029 | ingcode == 12203002 |
ingcode == 12203009 | ingcode == 12203010 | ingcode == 12203015 | ingcode == 12203017 |
ingcode == 12203022 | ingcode == 12203023 | ingcode == 12204007 | ingcode == 12204008 |
ingcode == 12204009 | ingcode == 12204010 | ingcode == 12204011 | ingcode == 12204012 |
ingcode == 12207001 | ingcode == 12207002 | ingcode == 12207017 | ingcode == 12207018 |
ingcode == 12209004 | ingcode == 12209005 | ingcode == 12210005 | ingcode == 12210006 |
ingcode == 12210009 | ingcode == 12210010 | ingcode == 12214001 | ingcode == 12214002 |
ingcode == 12214003 | ingcode == 12214006 | ingcode == 12214009 | ingcode == 12214010 |
ingcode == 12214011)
replace subgroups = 22 if (ingcode == 12201005 | ingcode == 12201006 | ingcode == 12201007 |
ingcode == 12201008 | ingcode == 12201017 | ingcode == 12201018 | ingcode == 12201019 |
ingcode == 12201020 | ingcode == 12201021 | ingcode == 12201022 | ingcode == 12201023 |
ingcode == 12201024 | ingcode == 12201028 | ingcode == 12201029 | ingcode == 12203002 |
ingcode == 12203009 | ingcode == 12203010 | ingcode == 12203015 | ingcode == 12203017 |
ingcode == 12203022 | ingcode == 12203023 | ingcode == 12204007 | ingcode == 12204008 |
ingcode == 12204009 | ingcode == 12204010 | ingcode == 12204011 | ingcode == 12204012 |
ingcode == 12207001 | ingcode == 12207002 | ingcode == 12207017 | ingcode == 12207018 |

```

```

ingcode == 12209004 | ingcode == 12209005 | ingcode == 12210005 | ingcode == 12210006 |
ingcode == 12210009 | ingcode == 12210010 | ingcode == 12214001 | ingcode == 12214002 |
ingcode == 12214003 | ingcode == 12214006 | ingcode == 12214009 | ingcode == 12214010 |
ingcode == 12214011)

```

*Ultra-processed breads: bagel, breadcrumbs, hot dog breads, fast food breads, pizza bases, all light breads and with addition of fibre, vitamins and minerals.

```

replace nova_group = 4 if (ingcode == 12207003 | ingcode == 12207004 | ingcode == 12201011 |
ingcode == 12201012 | ingcode == 12201001 | ingcode == 12201002 | ingcode == 12201003 |
ingcode == 12201027 | ingcode == 12202001 | ingcode == 12202002 | ingcode == 12202003 |
ingcode == 12202004 | ingcode == 12202005 | ingcode == 12202006 | ingcode == 12202007 |
ingcode == 12202008 | ingcode == 12202009 | ingcode == 12202010 | ingcode == 12202012 |
ingcode == 12202013 | ingcode == 12203018 | ingcode == 12203019 | ingcode == 12203020 |
ingcode == 12208001 | ingcode == 12208002 | ingcode == 12208003 | ingcode == 12208004 |
ingcode == 12208005 | ingcode == 12208006 | ingcode == 12208007 | ingcode == 12208008 |
ingcode == 12210003 | ingcode == 12210004 | ingcode == 12214007 | ingcode == 12214008)
replace subgroups = 25 if (ingcode == 12207003 | ingcode == 12207004 | ingcode == 12201011 |
ingcode == 12201012 | ingcode == 12201001 | ingcode == 12201002 | ingcode == 12201003 |
ingcode == 12201027 | ingcode == 12202001 | ingcode == 12202002 | ingcode == 12202003 |
ingcode == 12202004 | ingcode == 12202005 | ingcode == 12202006 | ingcode == 12202007 |
ingcode == 12202008 | ingcode == 12202009 | ingcode == 12202010 | ingcode == 12202012 |
ingcode == 12202013 | ingcode == 12203018 | ingcode == 12203019 | ingcode == 12203020 |
ingcode == 12208001 | ingcode == 12208002 | ingcode == 12208003 | ingcode == 12208004 |
ingcode == 12208005 | ingcode == 12208006 | ingcode == 12208007 | ingcode == 12208008 |
ingcode == 12210003 | ingcode == 12210004 | ingcode == 12214007 | ingcode == 12214008)

```

*Unclear (default G3)

```

replace nova_group = 3 if (ingcode == 12201009 | ingcode == 12201010 | ingcode == 12201013 |
ingcode == 12201014 | ingcode == 12201015 | ingcode == 12201016 | ingcode == 12201025 | ingcode
== 12201026 | ingcode == 12203003 | ingcode == 12203004 | ingcode == 12203011 | ingcode ==
12203012 | ingcode == 12203013 | ingcode == 12203014 | ingcode == 12203025 | ingcode ==
12203026 | ingcode == 12204001 | ingcode == 12204002 | ingcode == 12204003 | ingcode ==
12204004 | ingcode == 12204005 | ingcode == 12204006 | ingcode == 12204013 | ingcode ==
12204014 | ingcode == 12204015 | ingcode == 12204016 | ingcode == 12204017 | ingcode ==
12204018 | ingcode == 12206001 | ingcode == 12206002 | ingcode == 12206005 | ingcode ==
12206006 | ingcode == 12207005 | ingcode == 12207006 | ingcode == 12207007 | ingcode ==
12207008 | ingcode == 12207009 | ingcode == 12207010 | ingcode == 12207011 | ingcode ==
12207012 | ingcode == 12207013 | ingcode == 12207014 | ingcode == 12207015 | ingcode ==
12207016 | ingcode == 12207019 | ingcode == 12207020 | ingcode == 12207021 | ingcode ==
12207022 | ingcode == 12209002 | ingcode == 12209003 | ingcode == 12209008 | ingcode ==
12209009 | ingcode == 12210001 | ingcode == 12210002 | ingcode == 12210007 | ingcode ==
12210008 | ingcode == 12212001 | ingcode == 12212002 | ingcode == 12213001 | ingcode ==
12213002 | ingcode == 12213003 | ingcode == 12213004)
replace subgroups = 22 if (ingcode == 12201009 | ingcode == 12201010 | ingcode == 12201013 |
ingcode == 12201014 | ingcode == 12201015 | ingcode == 12201016 | ingcode == 12201025 | ingcode
== 12201026 | ingcode == 12203003 | ingcode == 12203004 | ingcode == 12203011 | ingcode ==
12203012 | ingcode == 12203013 | ingcode == 12203014 | ingcode == 12203025 | ingcode ==
12203026 | ingcode == 12204001 | ingcode == 12204002 | ingcode == 12204003 | ingcode ==
12204004 | ingcode == 12204005 | ingcode == 12204006 | ingcode == 12204013 | ingcode ==
12204014 | ingcode == 12204015 | ingcode == 12204016 | ingcode == 12204017 | ingcode ==
12204018 | ingcode == 12206001 | ingcode == 12206002 | ingcode == 12206005 | ingcode ==
12206006 | ingcode == 12207005 | ingcode == 12207006 | ingcode == 12207007 | ingcode ==
12207008 | ingcode == 12207009 | ingcode == 12207010 | ingcode == 12207011 | ingcode ==
12207012 | ingcode == 12207013 | ingcode == 12207014 | ingcode == 12207015 | ingcode ==
12207016 | ingcode == 12207019 | ingcode == 12207020 | ingcode == 12207021 | ingcode ==

```

```

12207022 | ingcode == 12209002 | ingcode == 12209003 | ingcode == 12209008 | ingcode ==
12209009 | ingcode == 12210001 | ingcode == 12210002 | ingcode == 12210007 | ingcode ==
12210008 | ingcode == 12212001 | ingcode == 12212002 | ingcode == 12213001 | ingcode ==
12213002 | ingcode == 12213003 | ingcode == 12213004)

```

```

*      123    English-style muffins, flat breads, and savoury and sweet breads

```

```

*      12301  English-style muffins

```

```

replace nova_group = 4 if digcode5_ing == 12301

```

```

replace subgroups = 25 if digcode5_ing == 12301

```

```

*      12302  Flat breads (e.g. Pita bread), wheat based

```

```

replace nova_group = 3 if digcode5_ing == 12302

```

```

replace subgroups = 22 if digcode5_ing == 12302

```

```

*      12303  Flat breads (e.g. Pita bread), other cereal flours

```

```

replace nova_group = 3 if digcode5_ing == 12303

```

```

replace subgroups = 22 if digcode5_ing == 12303

```

```

*      12304  Savoury filled or topped breads and bread rolls

```

```

* Disaggregated

```

```

*      12305  Sweet breads, buns and scrolls, uniced, unfilled

```

```

replace nova_group = 4 if digcode5_ing == 12305

```

```

replace subgroups = 25 if digcode5_ing == 12305

```

```

*      12306  Sweet breads, buns and scrolls, iced and/or filled

```

```

replace nova_group = 4 if digcode5_ing == 12306

```

```

replace subgroups = 25 if digcode5_ing == 12306

```

```

*      12307  Fried bread products and garlic breads

```

```

* Disaggregated, except:

```

```

replace nova_group = 4 if ingcode == 12307001

```

```

replace subgroups = 25 if ingcode == 12307001

```

```

*      124    Pasta and pasta products (without sauce)

```

```

*      12401  Pasta and noodles, wheat based, other than instant noodles

```

```

replace nova_group = 1 if digcode5_ing == 12401

```

```

replace subgroups = 7 if digcode5_ing == 12401

```

```

*      12402  Instant noodles and noodle products, wheat based

```

```

replace nova_group = 4 if digcode5_ing == 12402

```

```

replace subgroups = 40 if digcode5_ing == 12402

```

```

*      12403  Pasta and noodles, not wheat based

```

```

replace nova_group = 1 if digcode5_ing == 12403

```

```

replace subgroups = 7 if digcode5_ing == 12403

```

```

*      12404  Filled pasta

```

```

replace nova_group = 4 if digcode5_ing == 12404

```

```

replace subgroups = 26 if digcode5_ing == 12404

```

```

*      125    Breakfast cereals, ready to eat

```

```

replace nova_group = 4 if digcode3_ing == 125

```

```

replace subgroups = 27 if digcode3_ing == 125

```

```

*Exception (muesli natural style)

```

```

replace nova_group = 1 if (ingcode == 12514016 | ingcode == 12514017 | ingcode == 12514018 |
ingcode == 12514019 )

```

```

replace subgroups = 6 if (ingcode == 12514016 | ingcode == 12514017 | ingcode == 12514018 |
ingcode == 12514019 )

```

```

*      12501  Breakfast cereal, corn based

```

```

*      12502  Breakfast cereal, corn based, fortified

```

```

*      12503  Breakfast cereal, rice based

```

* 12504 Breakfast cereal, rice based, fortified
 * 12505 Breakfast cereal, wheat based
 * 12506 Breakfast cereal, wheat based, fortified, sugars ≤ 20 g/100g
 * 12507 Breakfast cereal, wheat based, fortified, sugars > 20 g/100g
 * 12508 Breakfast cereal, wheat based, with fruit and/or nuts, unfortified
 * 12509 Breakfast cereal, wheat based, with fruit and/or nuts, fortified, sugars ≤ 25 g/100g
 * 12510 Breakfast cereal, wheat based, with fruit and/or nuts, fortified, sugars > 25 g/100g
 * 12511 Breakfast cereal, mixed grain
 * 12512 Breakfast cereal, mixed grain, fortified, sugars ≤ 20 g/100g
 * 12513 Breakfast cereal, mixed grain, fortified, sugars > 20 g/100g
 * 12514 Breakfast cereal, mixed grain, with fruit and/or nuts
 * 12515 Breakfast cereal, mixed grain, with fruit and/or nuts, fortified
 * 12516 Breakfast cereal, other

* 126 Breakfast cereals, hot porridge style
 * 12601 Porridge style, oat based
 * 12602 Porridge style, other cereals

*All disaggregated, except:

replace nova_group = 4 if (ingcode == 12601017 | ingcode == 12601018)

replace subgroups = 27 if (ingcode == 12601017 | ingcode == 12601018)

* 13 Cereal based products and dishes

* 131 Sweet biscuits

replace nova_group = 4 if digcode3_ing == 131

replace subgroups = 31 if digcode3_ing == 131

* 13101 Sweet biscuits, plain or flavoured including short bread varieties
 * 13102 Sweet biscuits, plain with fruit or nuts
 * 13103 Sweet biscuits, with jam, marshmallow or other sugar-based filling
 * 13104 Sweet biscuits, cream-filled
 * 13105 Sweet biscuits, chocolate-coated, chocolate chip
 * 13106 Sweet biscuits, chocolate-coated, chocolate or cream filled
 * 13107 Sweet biscuits, other toppings

* 132 Savoury biscuits

replace nova_group = 4 if digcode3_ing == 132

replace subgroups = 31 if digcode3_ing == 132

* 13201 Savoury biscuits, wheat based, plain, energy ≤ 1800 kJ per 100 g
 * 13202 Savoury biscuits, wheat based, plain, energy > 1800 kJ per 100 g
 * 13203 Savoury biscuits, rye based
 * 13204 Savoury biscuits, rice based (includes rice cakes)
 * 13205 Savoury biscuits, corn based

* 133 Cakes, muffins, scones, cake-type desserts

replace nova_group = 4 if digcode3_ing == 133

replace subgroups = 32 if digcode3_ing == 133

* 13301 Cakes and cake mixes, chocolate
 * 13302 Cakes and cake mixes, sponge
 * 13303 Cakes and cake mixes, other types
 * 13304 Muffins, cake type, and muffin mixes
 * 13305 Cake-type desserts
 * 13306 Slices, biscuit and cake-type

```

*      13307 Scones and rock cakes, plain or with added fruit or vegetables only
*      13308 Scones and rock cakes, with added cheese, chocolate or similar
*      13309 Other desserts containing cereal

*      134    Pastries
*      13401 Pastry, plain/unfilled, all types
replace nova_group = 4 if digcode5_ing == 13401
replace subgroups = 32 if digcode5_ing == 13401
*      13402 Sweet pastry products, fruit and/or nut fillings
replace nova_group = 4 if digcode5_ing == 13402
replace subgroups = 32 if digcode5_ing == 13402
      *Exception (apple pie from fast food outlet)
replace subgroups = 46 if ingcode == 13402003
*      13403 Sweet pastry products, egg or dairy based fillings
replace nova_group = 4 if digcode5_ing == 13403
replace subgroups = 32 if digcode5_ing == 13403
*      13404 Savoury pastry products, quiches and flans
*Disaggregated, except (quiche commercial):
replace nova_group = 4 if ingcode == 13404003
replace subgroups = 32 if ingcode == 13404003
*      13405 Savoury pastry products, pies, rolls and envelopes
*Disaggregated, except (pies ready to heat/eat):
replace nova_group = 4 if (ingcode == 13405001 | ingcode == 13405008 | ingcode == 13405010 |
ingcode == 13405011 | ingcode == 13405012 | ingcode == 13405014 | ingcode == 13405016 | ingcode
== 13405018 | ingcode == 13405019 | ingcode == 13405020 | ingcode == 13405021 | ingcode ==
13405022 | ingcode == 13405024 | ingcode == 13405026 | ingcode == 13405030 | ingcode ==
13405032 | ingcode == 13405034 | ingcode == 13405035 | ingcode == 13405036)
replace subgroups = 26 if (ingcode == 13405001 | ingcode == 13405008 | ingcode == 13405010 |
ingcode == 13405011 | ingcode == 13405012 | ingcode == 13405014 | ingcode == 13405016 | ingcode
== 13405018 | ingcode == 13405019 | ingcode == 13405020 | ingcode == 13405021 | ingcode ==
13405022 | ingcode == 13405024 | ingcode == 13405026 | ingcode == 13405030 | ingcode ==
13405032 | ingcode == 13405034 | ingcode == 13405035 | ingcode == 13405036)
*      13406 Savoury pastry products, pies, rolls and envelopes, fried
replace nova_group = 4 if digcode5_ing == 13406
replace subgroups = 26 if digcode5_ing == 13406
replace subgroups = 46 if (ingcode == 13406001 | ingcode == 13405008 | ingcode == 13406007)
*      135    Mixed dishes where cereal is the major ingredient
*      13501 Pizza, saturated fat ≤5 g/100 g
replace nova_group = 4 if digcode5_ing == 13501
replace subgroups = 38 if (ingcode == 13501003 | ingcode == 13501004 | ingcode == 13501010 |
ingcode == 13501013 | ingcode == 13501019 | ingcode == 13501026 | ingcode == 13501035)
replace subgroups = 46 if (ingcode == 13501001 | ingcode == 13501002 | ingcode == 13501007 |
ingcode == 13501008 | ingcode == 13501016 | ingcode == 13501017 | ingcode == 13501021 | ingcode
== 13501023 | ingcode == 13501029 | ingcode == 13501030 | ingcode == 13501032 | ingcode ==
13501033 | ingcode == 13501038 | ingcode == 13501039)
*      13502 Pizza, saturated fat >5 g/100 g
replace nova_group = 4 if digcode5_ing == 13502
replace subgroups = 38 if (ingcode == 13502001 | ingcode == 13502003 | ingcode == 13502009)
replace subgroups = 46 if (ingcode == 13502006 | ingcode == 13502007 | ingcode == 13502010 |
ingcode == 13502011 | ingcode == 13502016 | ingcode == 13502017 | ingcode == 13502018)
*      13503 Sandwiches and filled rolls, saturated fat ≤5 g/100 g
* Disaggregated
*      13504 Sandwiches and filled rolls, saturated fat >5 g/100 g
* Disaggregated

```


* 13505 Burgers, saturated fat ≤ 5 g/100 g
 replace nova_group = 4 if digcode5_ing == 13505
 replace subgroups = 46 if digcode5_ing == 13505

* 13506 Burgers, saturated fat > 5 g/100 g
 replace nova_group = 4 if digcode5_ing == 13506
 replace subgroups = 46 if digcode5_ing == 13506

* 13507 Taco and tortilla-based dishes, saturated fat ≤ 5 g/100 g
 * Disaggregated, except:
 replace nova_group = 4 if (ingcode == 13507001 | ingcode == 13507002 | ingcode == 13507003 | ingcode == 13507004)
 replace subgroups = 46 if (ingcode == 13507001 | ingcode == 13507002 | ingcode == 13507003)
 replace subgroups = 26 if ingcode == 13507004

* 13508 Taco and tortilla-based dishes, saturated fat > 5 g/100 g
 * Disaggregated

* 13509 Savoury pasta/noodle and sauce dishes, saturated fat ≤ 5 g/100 g
 * Disaggregated, except (frozen, canned and dry mix pasta dish):
 replace nova_group = 4 if (ingcode == 13509004 | ingcode == 13509005 | ingcode == 13509006 | ingcode == 13509068 | ingcode == 13509070 | ingcode == 13509072 | ingcode == 13509073 | ingcode == 13509074)
 replace subgroups = 26 if (ingcode == 13509004 | ingcode == 13509005 | ingcode == 13509006)
 replace subgroups = 40 if (ingcode == 13509068 | ingcode == 13509070 | ingcode == 13509072 | ingcode == 13509073 | ingcode == 13509074)

* 13510 Savoury pasta/noodle and sauce dishes, saturated fat > 5 g/100 g
 * Disaggregated

* 13511 Savoury rice-based dishes, saturated fat ≤ 5 g/100 g
 * Disaggregated

* 13512 Savoury rice-based dishes, saturated fat > 5 g/100 g
 * Disaggregated

* 13513 Savoury dumplings
 * Disaggregated, except (gnocchi):
 replace nova_group = 4 if ingcode == 13513001
 replace subgroups = 26 if ingcode == 13513001

* 13514 Sushi, all types
 * Disaggregated, except (sushi comercial):
 replace nova_group = 3 if ingcode == 13514003
 replace subgroups = 24 if ingcode == 13514003

* 13515 Other savoury grain dishes
 * Disaggregated, except (breadcrumbs, commercial):
 replace nova_group = 4 if ingcode == 13515001
 replace subgroups = 25 if ingcode == 13515001

* 136 Batter-based products

* 13601 Pancakes, crepes and dishes
 * Disaggregated, except (dry mix, commercial, fast food chain):
 replace nova_group = 4 if (ingcode == 13601007 | ingcode == 13601008 | ingcode == 13601009 | ingcode == 13601015)
 replace subgroups = 32 if (ingcode == 13601007 | ingcode == 13601008 | ingcode == 13601009)
 replace subgroups = 46 if ingcode == 13601015

* 13602 Drop scones, pikelets
 * Disaggregated, except (commercial):
 replace nova_group = 4 if ingcode == 13602001
 replace subgroups = 32 if ingcode == 13602001

```

*      13603  Waffles
* Disaggregated, except (frozen):
replace nova_group = 4 if ingcode == 13603001
replace subgroups = 32 if ingcode == 13603001

*      13604  Batters and batter puddings
* Disaggregated, except (commercial):
replace nova_group = 4 if (ingcode == 13604001 | ingcode == 13604004 | ingcode == 13604009 )
replace subgroups = 32 if (ingcode == 13604001 | ingcode == 13604004 | ingcode == 13604009 )

*      13605  Doughnuts
replace nova_group = 4 if digcode5_ing == 13605
replace subgroups = 32 if digcode5_ing == 13605

*      13606  Crumpets
replace nova_group = 4 if digcode5_ing == 13606
replace subgroups = 32 if digcode5_ing == 13606

*****
*      14      Fats and oils
*****

*      141      Butters
replace nova_group = 2 if digcode3_ing == 141
replace subgroups = 16 if digcode3_ing == 141
*      14101  Butter
*      14102  Butter products

*      142      Dairy blends
replace nova_group = 4 if digcode3_ing == 142
replace subgroups = 34 if digcode3_ing == 142
*      14201  Dairy blend, regular, fat content ≥ 65g/100g
*      14202  Dairy blend, reduced fat, fat content <65 g/100g
*      14203  Dairy fats, unspecified type

*      143      Margarine and table spreads
replace nova_group = 4 if digcode3_ing == 143
replace subgroups = 34 if digcode3_ing == 143
*      14301  Polyunsaturated margarine spreads, fat content ≥ 65g/100g
*      14302  Polyunsaturated margarine spreads, fat content <65 g/100g
*      14303  Monounsaturated margarine spreads, fat content ≥65 g/100g
*      14304  Monounsaturated margarine spreads,fat content <65 g/100g
*      14305  Cooking margarine
*      14306  Margarine spreads with added phytosterols
*      14307  Unspecified margarine spread

*      144      Plant oils
replace nova_group = 2 if digcode3_ing == 144
replace subgroups = 17 if digcode3_ing == 144
*      14401  Polyunsaturated oils
*      14402  Monounsaturated oils
*      14403  Unspecified vegetable/nut oils

*      145      Other fats

```

```

replace nova_group = 2 if digcode3_ing == 145
replace subgroups = 16 if digcode3_ing == 145
  *Exception (shortening)
replace nova_group = 4 if (ingcode == 14501004 | ingcode == 14502004 | ingcode == 14502005 |
ingcode == 14502006 | ingcode == 14503002)
replace subgroups = 34 if (ingcode == 14501004 | ingcode == 14502004 | ingcode == 14502005 |
ingcode == 14502006 | ingcode == 14503002)
*   14501  Animal-based solid fats
*   14502  Vegetable-based solid fats
*   14503  Other fats or oils

*   146    Unspecified fats
*   14601  Unspecified dairy-based fat or margarine used as a spread
replace nova_group = 4 if digcode5_ing == 14601
replace subgroups = 34 if digcode5_ing == 14601
*   14602  Unspecified fats or oil used in cooking
replace nova_group = 2 if digcode5_ing == 14602
replace subgroups = 16 if digcode5_ing == 14602

*****
*   15    Fish and seafood products and dishes
*****

*   151    Fin fish (excluding commercially sterile)
*   15101  Fin fish, fresh, frozen
replace nova_group = 1 if digcode5_ing == 15101
replace subgroups = 11 if digcode5_ing == 15101
*   15102  Smoked fish
replace nova_group = 3 if digcode5_ing == 15102
replace subgroups = 23 if digcode5_ing == 15102
*   152    Crustacea and molluscs (excluding commercially sterile)
replace nova_group = 1 if digcode3_ing == 152
replace subgroups = 11 if digcode3_ing == 152
*   15201  Crustacea, fresh, frozen
*   15202  Molluscs, fresh, frozen
*   153    Other sea and freshwater foods
replace nova_group = 1 if digcode3_ing == 153
replace subgroups = 11 if digcode3_ing == 153
*   15301  Fish roe
*   15302  Eel
*   15303  Mixtures of finfish and/or shellfish

*   154    Packed (commercially sterile) fish and seafood
replace nova_group = 3 if digcode3_ing == 154
replace subgroups = 23 if digcode3_ing == 154
*   15401  Packed fin fish
*   15402  Packed crustacea and molluscs

*   155    Fish and seafood products (homemade and takeaway)
*   15501  Fin fish, battered or crumbed
*Disaggregated, except (frozen meals and snapper no add):

```

```

replace nova_group = 4 if (ingcode == 15501005 | ingcode == 15501009 | ingcode == 15501025 |
ingcode == 15501033 | ingcode == 15501036 | ingcode == 15501038 | ingcode == 15501041 | ingcode
== 15501042 | ingcode == 15501043)
replace subgroups = 26 if (ingcode == 15501005 | ingcode == 15501009 | ingcode == 15501025 |
ingcode == 15501033 | ingcode == 15501036 | ingcode == 15501038 | ingcode == 15501041 | ingcode
== 15501042 | ingcode == 15501043)
replace nova_group = 1 if ingcode == 15501027
replace subgroups = 11 if ingcode == 15501027
*      15502  Crustacea, battered or crumbed
*Disaggregated, except (frozen meals and takeaway):
replace nova_group = 4 if (ingcode == 15502003 | ingcode == 15502004)
replace subgroups = 26 if ingcode == 15502003
replace subgroups = 46 if ingcode == 15502004
*      15503  Molluscs, battered or crumbed
*Disaggregated, except (frozen meal):
replace nova_group = 4 if (ingcode == 15503008 | ingcode == 15503006)
replace subgroups = 26 if ingcode == 15503008
replace subgroups = 46 if ingcode == 15503006
*      15504  Fish and seafood products
*Disaggregated, except (canned, frozen meals, paste, takeaway):
replace nova_group = 3 if ingcode == 15504001
replace subgroups = 23 if ingcode == 15504001
replace nova_group = 4 if (ingcode == 15504002 | ingcode == 15504003 | ingcode == 15504006 |
ingcode == 15504013 | ingcode == 15504014 | ingcode == 15504015)
replace subgroups = 26 if (ingcode == 15504002 | ingcode == 15504003 | ingcode == 15504006 |
ingcode == 15504013 | ingcode == 15504014 | ingcode == 15504015)
replace nova_group = 4 if (ingcode == 15504008 | ingcode == 15504011)
replace subgroups = 36 if (ingcode == 15504008 | ingcode == 15504011)
replace nova_group = 4 if (ingcode == 15504007 | ingcode == 15504016)
replace subgroups = 46 if (ingcode == 15504007 | ingcode == 15504016)

*      156    Mixed dishes with fish or seafood as the major component
*      15601  Mixed dishes with fish as the major component
*Disaggregated
*      15602  Mixed dishes with fish as the major component, with rice, pasta or noodles
*Disaggregated, except (frozen and salty snack):
replace nova_group = 4 if (ingcode == 15602006 | ingcode == 15602008)
replace subgroups = 26 if ingcode == 15602006
replace subgroups = 37 if ingcode == 15602008
*      15603  Mixed seafood dishes with crustacea, molluscs or other seafood products as the major
component
*Disaggregated
*      15604  Mixed seafood dishes with crustacea, molluscs or other seafood products as the major
component, with rice, pasta or noodles
replace nova_group = 4 if ingcode == 15604003
replace subgroups = 26 if ingcode == 15604003
*      157    Wild caught fish and seafood (NATSINPAS)
replace nova_group = 1 if digcode3_ing == 157
replace subgroups = 11 if digcode3_ing == 157
*      15701  Wild caught finfish
*      15702  Wild caught crustacea and molluscs

*****
*      16    Fruit products and dishes

```

```
*****  
  
*      160    Wild harvested fruits (NATSINPAS)  
*      16001  Wild harvested fruits  
replace nova_group = 1 if digcode5_ing == 16001  
replace subgroups = 2 if digcode5_ing == 16001  
  
*      161    Pome fruit  
*      16101  Apples  
replace nova_group = 1 if digcode5_ing == 16101  
replace subgroups = 2 if digcode5_ing == 16101  
*      16102  Apples, commercially sterile  
replace nova_group = 3 if digcode5_ing == 16102  
replace subgroups = 21 if digcode5_ing == 16102  
*      16103  Pears  
replace nova_group = 1 if digcode5_ing == 16103  
replace subgroups = 2 if digcode5_ing == 16103  
*      16104  Pears, commercially sterile  
replace nova_group = 3 if digcode5_ing == 16104  
replace subgroups = 21 if digcode5_ing == 16104  
      *Exception (in light syrup or in intense sweetened liquid)  
replace nova_group = 4 if (ingcode == 16104004 | ingcode == 16104005 | ingcode == 16104006 |  
ingcode == 16104010 | ingcode == 16104011 | ingcode == 16104012)  
replace subgroups = 45 if (ingcode == 16104004 | ingcode == 16104005 | ingcode == 16104006 |  
ingcode == 16104010 | ingcode == 16104011 | ingcode == 16104012)  
*      16105  Other pome fruit  
replace nova_group = 1 if ingcode == 16105001  
replace subgroups = 2 if ingcode == 16105001  
replace nova_group = 3 if ingcode == 16105002  
replace subgroups = 21 if ingcode == 16105002  
  
*      162    Berry fruit  
*      16201  Berry fruit  
replace nova_group = 1 if digcode5_ing == 16201  
replace subgroups = 2 if digcode5_ing == 16201  
*      16202  Berry fruit, commercially sterile  
replace nova_group = 3 if digcode5_ing == 16202  
replace subgroups = 21 if digcode5_ing == 16202  
  
*      163    Citrus fruit  
*      16301  Oranges  
replace nova_group = 1 if digcode5_ing == 16301  
replace subgroups = 2 if digcode5_ing == 16301  
*      16302  Lemons and limes  
replace nova_group = 1 if digcode5_ing == 16302  
replace subgroups = 2 if digcode5_ing == 16302  
*      16303  Other citrus fruit  
replace nova_group = 1 if digcode5_ing == 16303  
replace subgroups = 2 if digcode5_ing == 16303  
*      16304  Citrus fruit, commercially sterile  
replace nova_group = 3 if digcode5_ing == 16304  
replace subgroups = 21 if digcode5_ing == 16304  
  
*      164    Stone fruit
```

```
*      16401 Peaches and nectarines
replace nova_group = 1 if digcode5_ing == 16401
replace subgroups = 2 if digcode5_ing == 16401
*      16402 Peaches and nectarines, commercially sterile
replace nova_group = 3 if digcode5_ing == 16402
replace subgroups = 21 if digcode5_ing == 16402
      *Exception (in light syrup or in intense sweetened liquid)
replace nova_group = 4 if (ingcode == 16402004 | ingcode == 16402005 | ingcode == 16402006 |
ingcode == 16402010 | ingcode == 16402011 | ingcode == 16402012)
replace subgroups = 45 if (ingcode == 16402004 | ingcode == 16402005 | ingcode == 16402006 |
ingcode == 16402010 | ingcode == 16402011 | ingcode == 16402012)
*      16403 Other stone fruit
replace nova_group = 1 if digcode5_ing == 16403
replace subgroups = 2 if digcode5_ing == 16403
*      16404 Other stone fruit, commercially sterile
replace nova_group = 3 if digcode5_ing == 16404
replace subgroups = 21 if digcode5_ing == 16404
      *Exception (in light syrup or in intense sweetened liquid)
replace nova_group = 4 if (ingcode == 16404004 | ingcode == 16404005 | ingcode == 16404006 |
ingcode == 16404010 | ingcode == 16404011 | ingcode == 16404012)
replace subgroups = 45 if (ingcode == 16404004 | ingcode == 16404005 | ingcode == 16404006 |
ingcode == 16404010 | ingcode == 16404011 | ingcode == 16404012)

*      165 Tropical and subtropical fruit
*      16501 Bananas
replace nova_group = 1 if digcode5_ing == 16501
replace subgroups = 2 if digcode5_ing == 16501
*      16502 Pineapples
replace nova_group = 1 if digcode5_ing == 16502
replace subgroups = 2 if digcode5_ing == 16502
*      16503 Other tropical and subtropical fruit, edible peel
replace nova_group = 1 if digcode5_ing == 16503
replace subgroups = 2 if digcode5_ing == 16503
*      16504 Other tropical and subtropical fruit, inedible peel
replace nova_group = 1 if digcode5_ing == 16504
replace subgroups = 2 if digcode5_ing == 16504
*      16505 Tropical and subtropical fruit, commercially sterile
replace nova_group = 3 if digcode5_ing == 16505
replace subgroups = 21 if digcode5_ing == 16505
replace nova_group = 4 if ingcode == 16505004
replace subgroups = 45 if ingcode == 16505004

*      166 Other fruit
replace nova_group = 1 if digcode3_ing == 166
replace subgroups = 2 if digcode3_ing == 166
*      16601 Other fruit

*      167 Mixtures of two or more groups of fruit
*      16701 Mixtures of two or more groups of fruit
* Disaggregated
*      16702 Mixtures of two or more groups of fruit, commercially sterile
replace nova_group = 3 if digcode5_ing == 16702
replace subgroups = 21 if digcode5_ing == 16702
```

```

*      168    Dried fruit, preserved fruit
*      16801  Dried vine fruit
replace nova_group = 1 if digcode5_ing == 16801
replace subgroups = 2 if digcode5_ing == 16801
*      16802  Other dried fruit including mixed dried fruit
replace nova_group = 1 if digcode5_ing == 16802
replace subgroups = 2 if digcode5_ing == 16802
      *Exception (banana chip)
replace nova_group = 3 if ingcode == 16802003
replace subgroups = 21 if ingcode == 16802003
      *Except (cramberry dried and sweetened)
replace nova_group = 3 if ingcode == 16802005
replace subgroups = 21 if ingcode == 16802005
*      16803  Dried fruit and nut mixes
*Disaggregated
*      16804  Preserved fruit
replace nova_group = 3 if digcode5_ing == 16804
replace subgroups = 21 if digcode5_ing == 16804

*      169    Mixed dishes where fruit is the major component
*      16901  Mixed dishes where fruit is the major component

*****
*      17     Egg products and dishes
*****

*      171    Eggs
replace nova_group = 1 if digcode3_ing == 171
replace subgroups = 9 if digcode3_ing == 171
*      17101  Eggs, chicken
*      17102  Eggs, chicken, modified (e.g. Omega-3, folate)
*      17103  Eggs, other

*      172    Dishes where egg is the major ingredient
*Disaggregated, except (egg dish prepared from dry mix):
replace nova_group = 4 if ingcode == 17201057
replace subgroups = 26 if ingcode == 17201057
*      17201  Egg dishes, savoury
*      17202  Egg dishes, sweet

*      174    Wild harvested eggs (NATSINPAS)
*      17401  Wild harvested eggs
replace nova_group = 1 if digcode3_ing == 174
replace subgroups = 9 if digcode3_ing == 174
* No obs.

*****
*      18     Meat, poultry and game products and dishes
*****

*      180    Wild harvested meat, and meat dishes (NATSINPAS)
replace nova_group = 1 if digcode3_ing == 180
replace subgroups = 4 if digcode3_ing == 180

```

```
*      18011  Wild harvested mammalian meat
*      18012  Wild harvested meat, mixed dishes

*      181    Beef, sheep and pork, unprocessed
replace nova_group = 1 if digcode3_ing == 181
replace subgroups = 4 if digcode3_ing == 181
      *Exception (silverside beef)
replace nova_group = 3 if ingcode == 18101165
replace subgroups = 23 if ingcode == 18101165
*      18101  Beef
*      18102  Lamb and mutton
*      18103  Pork
*      18104  Veal

*      182    Mammalian game meats
replace nova_group = 1 if digcode3_ing == 182
replace subgroups = 4 if digcode3_ing == 182
*      18201  Kangaroo
*      18202  Other mammalian game

*      183    Poultry and feathered game
replace nova_group = 1 if digcode3_ing == 183
replace subgroups = 5 if digcode3_ing == 183
*      18301  Chicken
*      18302  Other poultry
*      18303  Feathered game

*      184    Organ meats and offal, products and dishes
*      18401  Liver
replace nova_group = 1 if digcode5_ing == 18401
replace subgroups = 4 if digcode5_ing == 18401
*      18402  Kidney
replace nova_group = 1 if digcode5_ing == 18402
replace subgroups = 4 if digcode5_ing == 18402
*      18403  Other organ meats and offal
replace nova_group = 1 if digcode5_ing == 18403
replace subgroups = 4 if digcode5_ing == 18403
*      18404  Liver paste, pate and dishes
replace nova_group = 3 if digcode5_ing == 18404
replace subgroups = 23 if digcode5_ing == 18404
*      18405  Other organ meat, blood and offal dishes
replace nova_group = 3 if digcode5_ing == 18405
replace subgroups = 23 if digcode5_ing == 18405

*      185    Sausages, frankfurts and saveloys
replace nova_group = 4 if digcode3_ing == 185
replace subgroups = 28 if digcode3_ing == 185
      *Exception - kangaroo
replace nova_group = 3 if ingcode == 18503009
replace subgroups = 23 if ingcode == 18503009
*      18501  Sausage, saturated fat content >5 g/100g
*      18502  Frankfurts and saveloys, saturated fat content >5 g/100g
*      18503  Sausages, frankfurts and saveloys, saturated fat content ≤5 g/100g
```



```
*      186      Processed meat
replace nova_group = 4 if digcode3_ing == 186
replace subgroups = 28 if digcode3_ing == 186
      *Exception (Bacon)
*      18601    Bacon
replace nova_group = 3 if digcode5_ing == 18601
replace subgroups = 23 if digcode5_ing == 18601
*      18602    Ham
*      18603    Fermented, comminuted meats (e.g. Salami)
*      18604    Processed delicatessen meat, mammalian
*      18605    Processed delicatessen meat, poultry
*      18606    Processed meat, commercially sterile (includes canned meats)
*      18607    Dried meats

*      187      Mixed dishes where beef, sheep, pork or mammalian game is the major component
*      18701    Beef dishes with gravy, sauce or vegetables
*Disaggregated, except (frozen and bolognese sauce):
replace nova_group = 4 if (ingcode == 18701014 | ingcode == 18701021 | ingcode == 18701022 |
ingcode == 18701023)
replace subgroups = 26 if (ingcode == 18701021 | ingcode == 18701022)
replace subgroups = 36 if (ingcode == 18701014 | ingcode == 18701023)
*      18702    Beef dishes, added pasta, noodles or rice
*Disaggregated, except (frozen meals):
replace nova_group = 4 if (ingcode == 18702012 | ingcode == 18702013)
replace subgroups = 26 if (ingcode == 18702012 | ingcode == 18702013)
*      18703    Beef, crumbed, battered, meatloaf or patty type with cereal and/or vegetables
*Disaggregated, except (frozen meals and fast food):
replace nova_group = 4 if (ingcode == 18703009 | ingcode == 18703010 | ingcode == 18703022)
replace subgroups = 26 if ingcode == 18703009
replace subgroups = 46 if (ingcode == 18703010 | ingcode == 18703022)
*      18704    Veal, crumbed, battered, meatloaf or patty type with cereal and/or vegetables
*Disaggregated
*      18705    Lamb or mutton dishes with gravy, sauce or vegetables
*Disaggregated, except (frozen meals):
replace nova_group = 4 if ingcode == 18705014
replace subgroups = 26 if ingcode == 18705014
*      18706    Lamb or mutton dishes, added pasta, noodles or rice
*Disaggregated
*      18707    Lamb or mutton, crumbed, battered, meatloaf or patty type with cereal and/or
vegetables
*Disaggregated
*      18708    Pork dishes with gravy, sauce or vegetables
*Disaggregated
*      18709    Pork dishes, added pasta, noodles or rice
*Disaggregated, except (frozen meals):
replace nova_group = 4 if ingcode == 18709001
replace subgroups = 26 if ingcode == 18709001
*      18710    Pork, crumbed, battered, meatloaf or patty type with cereal and/or vegetables
*Disaggregated
*      18711    Mammalian game meat, mixed dishes
*Disaggregated
*      18712    Unspecified meat, mixed dishes
*Disaggregated, except (commercial meatloaf):
replace nova_group = 4 if ingcode == 18712001
```

replace subgroups = 26 if ingcode == 18712001

* 188 Mixed dishes where sausage, bacon, ham or other processed meat is the major component

* 18801 Sausage dishes with gravy, sauce or vegetables

*Disaggregated, except (frozen meals and saveloy):

replace nova_group = 4 if (ingcode == 18801003 | ingcode == 18801004)

replace subgroups = 26 if ingcode == 18801003

replace subgroups = 28 if ingcode == 18801004

* 18802 Sausage dishes with gravy, sauce or vegetables, added pasta, noodles or rice

*Disaggregated

* 18803 Other processed meat, mixed dish

*Disaggregated

* 189 Mixed dishes where poultry or feathered game is the major component

* 18901 Poultry dishes, with gravy, sauce or vegetables

*Disaggregated, except (frozen meals):

replace nova_group = 4 if (ingcode == 18901018 | ingcode == 18901019)

replace subgroups = 26 if (ingcode == 18901018 | ingcode == 18901019)

* 18902 Poultry dishes, with gravy, sauce or vegetables, added pasta, noodles or rice

*Disaggregated, except (frozen meals):

replace nova_group = 4 if (ingcode == 18902015 | ingcode == 18902016 | ingcode == 18902017 | ingcode == 18902018 | ingcode == 18902019)

replace subgroups = 26 if (ingcode == 18902015 | ingcode == 18902016 | ingcode == 18902017 | ingcode == 18902018 | ingcode == 18902019)

* 18903 Poultry crumbed, battered, meatloaf or patty type with cereal and/or vegetables

*Disaggregated, except (frozen meals and fast food):

replace nova_group = 4 if (ingcode == 18903003 | ingcode == 18903004 | ingcode == 18903030 | ingcode == 18903031 | ingcode == 18903034 | ingcode == 18903047)

replace subgroups = 26 if (ingcode == 18903003 | ingcode == 18903004 | ingcode == 18903030 | ingcode == 18903031 | ingcode == 18903034 | ingcode == 18903047)

replace nova_group = 4 if (ingcode == 18903026 | ingcode == 18903027 | ingcode == 18903028 | ingcode == 18903029 | ingcode == 18903035 | ingcode == 18903036 | ingcode == 18903037 | ingcode == 18903038 | ingcode == 18903045 | ingcode == 18903046)

replace subgroups = 46 if (ingcode == 18903026 | ingcode == 18903027 | ingcode == 18903028 | ingcode == 18903029 | ingcode == 18903035 | ingcode == 18903036 | ingcode == 18903037 | ingcode == 18903038 | ingcode == 18903045 | ingcode == 18903046)

* 19 Milk products and dishes

* 191 Dairy milk (cow, sheep and goat)

* 19101 Milk, cow, fluid, regular whole, full fat

replace nova_group = 1 if digcode5_ing == 19101

replace subgroups = 1 if digcode5_ing == 19101

* 19102 Milk, cow, fluid, regular whole, full fat, fortified

replace nova_group = 1 if digcode5_ing == 19102

replace subgroups = 1 if digcode5_ing == 19102

* 19103 Milk, cow, fluid, reduced fat, <2 g/100g

replace nova_group = 1 if digcode5_ing == 19103

replace subgroups = 1 if digcode5_ing == 19103

* 19104 Milk, cow, fluid, reduced fat, <2 g/100g, fortified

```
replace nova_group = 1 if digcode5_ing == 19104
replace subgroups = 1 if digcode5_ing == 19104
* 19105 Milk, cow, fluid, skim, non-fat
replace nova_group = 1 if digcode5_ing == 19105
replace subgroups = 1 if digcode5_ing == 19105
* 19106 Milk, evaporated or condensed, undiluted
replace nova_group = 1 if digcode5_ing == 19106
replace subgroups = 1 if digcode5_ing == 19106
*Exception (sweetened condensed milk)
replace nova_group = 3 if ingcode == 19106004 | ingcode == 19106005
replace subgroups = 24 if ingcode == 19106004 | ingcode == 19106005
* 19107 Milk, powder, cow, dry
replace nova_group = 1 if digcode5_ing == 19107
replace subgroups = 1 if digcode5_ing == 19107
* 19108 Milk, non-bovine species
replace nova_group = 1 if digcode5_ing == 19108
replace subgroups = 1 if digcode5_ing == 19108
* 19109 Milk, fluid, unspecified
replace nova_group = 1 if digcode5_ing == 19109
replace subgroups = 1 if digcode5_ing == 19109

* 192 Yoghurt
* 19201 Yoghurt, natural, regular fat and high fat (>4 g/100g fat)
replace nova_group = 1 if digcode5_ing == 19201
replace subgroups = 1 if digcode5_ing == 19201
* 19202 Yoghurt, natural, reduced fat
replace nova_group = 1 if digcode5_ing == 19202
replace subgroups = 1 if digcode5_ing == 19202
* 19203 Yoghurt, natural, skim and non-fat
replace nova_group = 1 if digcode5_ing == 19203
replace subgroups = 1 if digcode5_ing == 19203
* 19204 Yoghurt, flavoured or added fruit and/or cereal, high fat (>4 g/100g fat)
replace nova_group = 4 if digcode5_ing == 19204
replace subgroups = 35 if digcode5_ing == 19204
* 19205 Yoghurt, flavoured or added fruit, full fat
replace nova_group = 4 if digcode5_ing == 19205
replace subgroups = 35 if digcode5_ing == 19205
* 19206 Yoghurt, flavoured or added fruit with added cereal, full fat
replace nova_group = 4 if digcode5_ing == 19206
replace subgroups = 35 if digcode5_ing == 19206
* 19207 Yoghurt, flavoured or added fruit, reduced fat
replace nova_group = 4 if digcode5_ing == 19207
replace subgroups = 35 if digcode5_ing == 19207
* 19208 Yoghurt, flavoured or added fruit, low fat or skim, sugar sweetened
replace nova_group = 4 if digcode5_ing == 19208
replace subgroups = 35 if digcode5_ing == 19208
* 19209 Yoghurt, flavoured or added fruit, low fat or skim, intense sweetened
replace nova_group = 4 if digcode5_ing == 19209
replace subgroups = 35 if digcode5_ing == 19209
* 19210 Yoghurt, drinks, buttermilk
replace nova_group = 4 if digcode5_ing == 19210
replace subgroups = 35 if digcode5_ing == 19210
* 19211 Yoghurt, added nutrients or other substances
replace nova_group = 4 if digcode5_ing == 19211
replace subgroups = 35 if digcode5_ing == 19211
```

```
*      19212  Yoghurt, unspecified fat
replace nova_group = 4 if digcode5_ing == 19212
replace subgroups = 35 if digcode5_ing == 19212
      *Except (Yoghurt, natural, not further defined)
replace nova_group = 1 if ingcode == 19212001
replace subgroups = 1 if ingcode == 19212001

*      193    Cream
replace nova_group = 2 if digcode3_ing == 193
replace subgroups = 16 if digcode3_ing == 193
      *Exception (19305    Cream substitute, artificial cream )
replace nova_group = 4 if digcode5_ing == 19305
replace subgroups = 45 if digcode5_ing == 19305
*      19301  Cream, regular and increased fat
*      19302  Cream, reduced fat
*      19303  Cream, sour
*      19304  Cream, sour, reduced fat
*      19305  Cream substitute, artificial cream
*      19306  Cream, unspecified type or fat level

*      194    Cheese
*      19401  Cheese, hard cheese ripened styles
replace nova_group = 3 if digcode5_ing == 19401
replace subgroups = 20 if digcode5_ing == 19401
      *Exception (flavoured cheese)
replace nova_group = 4 if (ingcode == 19401003)
replace subgroups = 44 if (ingcode == 19401003)
*      19402  Cheese, hard cheese ripened styles, reduced fat
replace nova_group = 3 if digcode5_ing == 19402
replace subgroups = 20 if digcode5_ing == 19402
*      19403  Cheese, unripened styles, including cream and cottage cheese, regular fat
replace nova_group = 3 if digcode5_ing == 19403
replace subgroups = 20 if digcode5_ing == 19403
      *Exception (flavoured cheese)
replace nova_group = 4 if (ingcode == 19403006 | ingcode == 19403007)
replace subgroups = 44 if (ingcode == 19403006 | ingcode == 19403007)
*      19404  Cheese, unripened styles, including cream and cottage cheese, reduced fat
replace nova_group = 3 if digcode5_ing == 19404
replace subgroups = 20 if digcode5_ing == 19404
*      19405  Cheese, camembert, brie and other surface ripened cheeses
replace nova_group = 3 if digcode5_ing == 19405
replace subgroups = 20 if digcode5_ing == 19405
*      19406  Cheese, processed
replace nova_group = 4 if digcode5_ing == 19406
replace subgroups = 44 if digcode5_ing == 19406
*      19407  Cheese, processed, reduced fat
replace nova_group = 4 if digcode5_ing == 19407
replace subgroups = 44 if digcode5_ing == 19407
*      19408  Cheese, not further defined
replace nova_group = 3 if digcode5_ing == 19408
replace subgroups = 20 if digcode5_ing == 19408

*      195    Frozen milk products
replace nova_group = 4 if digcode3_ing == 195
replace subgroups = 30 if digcode3_ing == 195
```

```

*      19501 Ice cream, tub varieties, fat content >10 g/100 g
*      19502 Ice cream, tub varieties, fat content 4 - 10 g/100 g
*      19503 Ice cream, tub varieties, fat content <4 g/100 g
*      19504 Ice cream, individual bar, stick and cone varieties, fat content >10 g/100 g
*      19505 Ice cream, individual bar, stick and cone varieties, fat content 4 - 10 g/100 g
*      19506 Ice cream, individual bar, stick and cone varieties, fat content <4 g/100 g
*      19507 Frozen yoghurts, all types
*      19508 Frozen dairy desserts, other

*      196      Custards
replace nova_group = 4 if digcode3_ing == 196
replace subgroups = 39 if digcode3_ing == 196
*      19601 Custard, fat content ≥ 4 g/100 g
*      19602 Custard, fat content <4 g/100 g

*      197      Other dishes where milk or a milk product is the major component
replace nova_group = 4 if digcode3_ing == 197
replace subgroups = 39 if digcode3_ing == 197
*      19701 Dairy desserts, smooth or gelatin-based dairy desserts
*      19702 Other milk, cheese or cream-based desserts

*      198      Flavoured milks and milkshakes
replace nova_group = 4 if digcode3_ing == 198
replace subgroups = 35 if digcode3_ing == 198
*      19801 Milk, coffee/chocolate flavoured and milk-based drinks, full fat
*      19802 Milk, other flavoured and milk-based drinks, full fat
*      19803 Milk, coffee/chocolate flavoured and milk-based drinks, reduced fat
*      19804 Milk, other flavoured and milk-based drinks, reduced fat
*      19805 Milk, other flavoured and milk-based drinks, not stated as to fat
*      19806 Milk-based fruit drinks

*****
*      20      Dairy & meat substitutes
*****

*      201      Dairy milk substitutes, unflavoured
replace nova_group = 4 if digcode3_ing == 201
replace subgroups = 35 if digcode3_ing == 201
*      20101 Soy-based beverage, plain
*      20102 Soy-based beverage, plain, fortified
*      20103 Soy-based beverage, plain, reduced fat
*      20104 Soy-based beverage, plain, reduced fat, fortified
*      20105 Soy-based beverage, plain, skim, fortified
*      20106 Cereal- or nut-based milk substitute
*      20107 Dairy milk substitute, unspecified

*      202      Dairy milk substitutes, flavoured
replace nova_group = 4 if digcode3_ing == 202
replace subgroups = 35 if digcode3_ing == 202
*      20201 Soy-based beverage, regular fat, flavoured
*      20202 Soy-based beverage, reduced fat, flavoured

*      203      Cheese substitute
replace nova_group = 4 if digcode3_ing == 203
replace subgroups = 45 if digcode3_ing == 203

```

```

*      20301 Cheese substitute

*      204 Soy-based ice confection
replace nova_group = 4 if digcode3_ing == 204
replace subgroups = 29 if digcode3_ing == 204
*      20401 Soy-based ice confection

*      205 Soy-based yoghurts
replace nova_group = 4 if digcode3_ing == 205
replace subgroups = 35 if digcode3_ing == 205
*      20501 Soy-based yoghurts, regular fat
*      20502 Soy-based yoghurts, reduced fat

*      206 Meat substitutes
*      20601 Meat substitutes
replace nova_group = 4 if digcode3_ing == 206
replace subgroups = 28 if digcode3_ing == 206

*      207 Dishes where meat substitutes are the major component
*      20701 Dishes where meat substitutes are the major component
* Desaggregated

*****
*      21 Soup
*****

*      211 Soup, homemade from basic ingredients
* Desaggregated
*      21101 Soup containing meat, poultry or seafood
*      21102 Soup, vegetable only

*      212 Dry soup mix
replace nova_group = 4 if digcode3_ing == 212
replace subgroups = 40 if digcode3_ing == 212
*      21201 Dry soup mix containing meat, poultry or seafood
*      21202 Dry soup mix, vegetable only

*      213 Soup, prepared from dry soup mix
replace nova_group = 4 if digcode3_ing == 213
replace subgroups = 40 if digcode3_ing == 213
*      21301 Soup containing meat, poultry or seafood
*      21302 Soup, vegetable only

*      214 Canned condensed soup (unprepared)
replace nova_group = 4 if digcode3_ing == 214
replace subgroups = 40 if digcode3_ing == 214
*      21401 Soup containing meat, poultry or seafood
*      21402 Soup, vegetable only

*      215 Soup, commercially sterile, prepared from condensed or sold ready to heat
replace nova_group = 4 if digcode3_ing == 215
replace subgroups = 40 if digcode3_ing == 215
*      21501 Soup containing meat, poultry or seafood
*      21502 Soup, vegetable only

```

```

*      216    Soup, not commercially sterile, purchased ready to eat
* Desaggregated
*      21601  Soup containing meat, poultry or seafood
*      21602  Soup, vegetable only

*****
*      22     Seed and nut products and dishes
*****

*      221    Seeds and seed products
replace nova_group = 1 if digcode3_ing == 221
replace subgroups = 10 if digcode3_ing == 221
*      22101  Seeds
*      22102  Seed products

*      222    Nuts and nut products
*      22201  Peanuts
replace nova_group = 1 if digcode5_ing == 22201
replace subgroups = 10 if digcode5_ing == 22201
*Exception (salted)
replace nova_group = 3 if ingcode == 22201002 | ingcode == 22201003
replace subgroups = 24 if ingcode == 22201002 | ingcode == 22201003
*      22202  Peanut products
replace nova_group = 3 if digcode5_ing == 22202
replace subgroups = 24 if digcode5_ing == 22202
*Exception (Satay sauce)
replace nova_group = 4 if ingcode == 22202006
replace subgroups = 36 if ingcode == 22202006
*      22203  Coconut and coconut products
*Coconut fresh, juice
replace nova_group = 1 if digcode5_ing == 22203
replace subgroups = 2 if digcode5_ing == 22203
*Coconut grated, canned
replace nova_group = 3 if ( ingcode == 22203005 | ingcode == 22203006 | ingcode == 22203007 |
ingcode == 22203008 | ingcode == 22203009 | ingcode == 22203010)
replace subgroups = 24 if ( ingcode == 22203005 | ingcode == 22203006 | ingcode == 22203007 |
ingcode == 22203008 | ingcode == 22203009 | ingcode == 22203010)
*      22204  Other nuts and nut products and dishes
replace nova_group = 1 if digcode5_ing == 22204
replace subgroups = 10 if digcode5_ing == 22204
*Exception (salted, w/honey)
replace nova_group = 3 if (ingcode == 222040003 | ingcode == 22204009 | ingcode == 22204011 |
ingcode == 22204016 | ingcode == 22204017 | ingcode == 22204020)
replace subgroups = 24 if (ingcode == 222040003 | ingcode == 22204009 | ingcode == 22204011 |
ingcode == 22204016 | ingcode == 22204017 | ingcode == 22204020)

*      22205  Mixed nuts or nuts and seeds
replace nova_group = 1 if digcode5_ing == 22205
replace subgroups = 10 if digcode5_ing == 22205
*Exception (salted, w/honey)
replace nova_group = 3 if (ingcode == 22205001 | ingcode == 22205004 | ingcode == 22205005 |
ingcode == 22205007 | ingcode == 22205009)
replace subgroups = 24 if (ingcode == 22205001 | ingcode == 22205004 | ingcode == 22205005 |
ingcode == 22205007 | ingcode == 22205009)

```

```
*      223      Wild harvested seeds and nuts (NATSINPAS)
replace nova_group = 1 if digcode3_ing == 223
replace subgroups = 10 if digcode3_ing == 223
*      22301    Wild harvested seeds
*      22302    Wild harvested nuts

*****
*      23       Savoury sauces and condiments
*****

*      231     Gravies and savoury sauces
*      23101   Gravies (prepared)
replace nova_group = 4 if digcode5_ing == 23101
replace subgroups = 36 if digcode5_ing == 23101
*      23102   Dry gravy mixes
replace nova_group = 4 if digcode5_ing == 23102
replace subgroups = 36 if digcode5_ing == 23102
*      23103   Savoury sauces, not tomato based, commercial
replace nova_group = 4 if digcode5_ing == 23103
replace subgroups = 36 if digcode5_ing == 23103
*      23104   Savoury sauces, tomato based, commercial
replace nova_group = 4 if digcode5_ing == 23104
replace subgroups = 36 if digcode5_ing == 23104
*      23105   Savoury sauces, not tomato based, homemade
*Disaggregated
*      23106   Savoury sauces, tomato based, homemade
*Disaggregated
*      23107   Savoury sauces, commercial, simmer style
replace nova_group = 4 if digcode5_ing == 23107
replace subgroups = 36 if digcode5_ing == 23107
*      23108   Savoury sauces, dairy based, homemade
*Disaggregated
*      23109   Savoury pastes
replace nova_group = 4 if digcode5_ing == 23109
replace subgroups = 36 if digcode5_ing == 23109
*      23110   Dry savoury sauces and casserole bases and dry mixes
replace nova_group = 4 if digcode5_ing == 23110
replace subgroups = 36 if digcode5_ing == 23110

*      232     Pickles, chutneys and relishes
replace nova_group = 3 if digcode3_ing == 232
replace subgroups = 21 if digcode3_ing == 232
*      23201   Fruit-based pickles, chutneys and relishes
*      23202   Vegetable-based pickles, chutneys and relishes

*      233     Salad dressings
replace nova_group = 4 if digcode3_ing == 233
replace subgroups = 36 if digcode3_ing == 233
*      23301   Mayonnaise and cream-style dressings, full fat
*      23302   Mayonnaise and cream-style dressings, reduced or non-fat
*      23303   Italian and French-style dressings, full fat
*      23304   Italian and French-style dressings, reduced or non-fat
*      23305   Vinegar
replace nova_group = 2 if digcode5_ing == 23305
```



```

replace subgroups = 18 if digcode5_ing == 23305

*      234      Stuffings
replace nova_group = 4 if digcode3_ing == 234
replace subgroups = 45 if digcode3_ing == 234
*      23401    Bread-based stuffings

*      235      Dips
replace nova_group = 4 if digcode3_ing == 235
replace subgroups = 36 if digcode3_ing == 235
*      23501    Dairy based dips
*      23502    Vegetable based dips
*      23503    Legume based dips
*      23504    Other dips

*****
*      24      Vegetable products and dishes
*****

*      240      Wild harvested vegetables, and vegetable dishes (NATSINPAS)
replace nova_group = 1 if digcode3_ing == 240
replace subgroups = 3 if digcode3_ing == 240
*      24001    Wild harvested vegetables
*      24002    Wild harvested vegetable dishes

*      241      Potatoes
*      24101    Potatoes
replace nova_group = 1 if digcode5_ing == 24101
replace subgroups = 3 if digcode5_ing == 24101
*      24102    Potato products
replace nova_group = 4 if digcode5_ing == 24102
replace subgroups = 33 if digcode5_ing == 24102
      *Exception (fast food)
replace nova_group = 4 if (ingcode == 24102002 | ingcode == 24102003 | ingcode == 24102004 |
ingcode == 24102005 | ingcode == 24102014 | ingcode == 24102015 | ingcode == 24102016 | ingcode
== 24102017 | ingcode == 24102018 | ingcode == 24102023 | ingcode == 24102027 | ingcode
== 24102028 | ingcode == 24102032 | ingcode == 24102033)
replace subgroups = 46 if (ingcode == 24102002 | ingcode == 24102003 | ingcode == 24102004 |
ingcode == 24102005 | ingcode == 24102014 | ingcode == 24102015 | ingcode == 24102016 | ingcode
== 24102017 | ingcode == 24102018 | ingcode == 24102023 | ingcode == 24102027 | ingcode
== 24102028 | ingcode == 24102032 | ingcode == 24102033)
*      24103    Potato mixed dishes
*Disaggregated, except (prepared from dry mix and fast food; and mashed (without add)
replace nova_group = 4 if (ingcode == 24103001 | ingcode == 24103002 | ingcode == 24103003)
replace subgroups = 33 if (ingcode == 24103001 | ingcode == 24103002 | ingcode == 24103003)
replace nova_group = 1 if (ingcode == 24103004 | ingcode == 24103017)
replace subgroups = 3 if (ingcode == 24103004 | ingcode == 24103017)
*      242      Cabbage, cauliflower and similar brassica vegetables
replace nova_group = 1 if digcode3_ing == 242
replace subgroups = 8 if digcode3_ing == 242
      *Except (canned)
replace nova_group = 3 if ingcode == 24201037
replace subgroups = 21 if ingcode == 24201037
*      24201    Cabbage and similar brassica vegetables
*      24202    Broccoli, broccolini and cauliflower

```

```
*      243      Carrot and similar root vegetables
replace nova_group = 1 if digcode3_ing == 243
replace subgroups = 8 if digcode3_ing == 243
      *Except (canned)
replace nova_group = 3 if ingcode == 24301004
replace subgroups = 21 if ingcode == 24301004
*      24301      Carrots
*      24302      Other root vegetables

*      244      Leaf and stalk vegetables
replace nova_group = 1 if digcode3_ing == 244
replace subgroups = 8 if digcode3_ing == 244
      *Except (canned)
replace nova_group = 3 if (ingcode == 24402011 | ingcode == 24401022 | ingcode == 24402004 |
ingcode == 24402010)
replace subgroups = 21 if (ingcode == 24402011 | ingcode == 24401022 | ingcode == 24402004 |
ingcode == 24402010)
*      24401      Leaf vegetables
*      24402      Stalk vegetables
*      24403      Herbs, fresh
*      24404      Seaweeds

*      245      Peas and beans
replace nova_group = 1 if digcode3_ing == 245
replace subgroups = 12 if digcode3_ing == 245
      *Except (canned)
replace nova_group = 3 if (ingcode == 24501006 | ingcode == 24502011)
replace subgroups = 21 if (ingcode == 24501006 | ingcode == 24502011)
*      24501      Peas and edible-podded peas
*      24502      Beans
*      24503      Sprouts

*      246      Tomato and tomato products
*      24601      Tomato
replace nova_group = 1 if digcode5_ing == 24601
replace subgroups = 8 if digcode5_ing == 24601
*      24602      Tomato products
replace nova_group = 3 if digcode5_ing == 24602
replace subgroups = 24 if digcode5_ing == 24602
      *Except (canned in tomato juice)
replace nova_group = 1 if (ingcode == 24602006 | ingcode == 24602007 | ingcode == 24602008 |
ingcode == 24602009)
replace subgroups = 8 if (ingcode == 24602006 | ingcode == 24602007 | ingcode == 24602008 |
ingcode == 24602009)

*      247      Other fruiting vegetables
replace nova_group = 1 if digcode3_ing == 247
replace subgroups = 8 if digcode3_ing == 247
      *Except (canned)
replace nova_group = 3 if (ingcode == 24703015 | ingcode == 24703016 | ingcode == 24704001
|ingcode == 24704002 | ingcode == 24704009 | ingcode == 24704010)
replace subgroups = 21 if (ingcode == 24703015 | ingcode == 24703016 | ingcode == 24704001
|ingcode == 24704002 | ingcode == 24704009 | ingcode == 24704010)
*      24701      Pumpkin
```

```

*      24702 Squash and zucchini
*      24703 Mushrooms
*      24704 Sweetcorn
*      24705 Other fruiting vegetables

*      248    Other vegetables and vegetable combinations
replace nova_group = 1 if digcode3_ing == 248
replace subgroups = 8 if digcode3_ing == 248
    *Except (canned)
replace nova_group = 3 if ingcode == 24801003
replace subgroups = 21 if ingcode == 24801003
*      24801 Other vegetables
*      24802 Onion, leek and garlic
*      24803 Mixtures of two or more vegetables
    *Disaggregated, except:
replace nova_group = 4 if ingcode == 24803001
replace subgroups = 26 if ingcode == 24803001

*      249    Dishes where vegetable is the major component
* Disaggregated, except:
replace nova_group = 4 if ingcode == 24903005
replace subgroups = 46 if ingcode == 24903005
replace nova_group = 1 if (ingcode == 24904009 | ingcode == 24904033)
replace subgroups = 8 if (ingcode == 24904009 | ingcode == 24904033)
*      24901 Vegetables and sauce
*      24902 Stuffed vegetables and vegetable dishes
*      24903 Fried vegetable dishes
*      24904 Salads, vegetable based
*      24905 Salads, vegetable based, added meat, fish or eggs

*****
*      25    Legume and pulse products and dishes
*****

*      251    Mature legumes and pulses
*      25101 Mature legumes and pulses
replace nova_group = 1 if digcode5_ing == 25101
replace subgroups = 12 if digcode5_ing == 25101
*      25102 Mature legumes and pulses, commercially sterile
replace nova_group = 3 if digcode5_ing == 25102
replace subgroups = 21 if digcode5_ing == 25102

*      252    Mature legume and pulse products and dishes
*      25201 Legume and pulse products
replace nova_group = 1 if (ingcode == 25201001 | ingcode == 25201002)
replace subgroups = 12 if (ingcode == 25201001 | ingcode == 25201002)
replace nova_group = 4 if (ingcode == 25201004 | ingcode == 25201005 | ingcode == 25201006 |
ingcode == 25201007 | ingcode == 25201008 | ingcode == 25201009 | ingcode == 25201010)
replace subgroups = 26 if (ingcode == 25201004 | ingcode == 25201005 | ingcode == 25201006 |
ingcode == 25201007 | ingcode == 25201008 | ingcode == 25201009 | ingcode == 25201010)
*      25202 Dishes where mature legumes are the major component
*Disaggregated, except (bean paste)
replace nova_group = 3 if (ingcode == 25201003 | ingcode == 25202001)
replace subgroups = 21 if (ingcode == 25201003 | ingcode == 25202001)

```

replace nova_group = 4 if (ingcode == 25202006 | ingcode == 25202009)
 replace subgroups = 26 if (ingcode == 25202006 | ingcode == 25202009)

* 26 Snack foods

* 261 Potato snacks

replace nova_group = 4 if digcode3_ing == 261

replace subgroups = 37 if digcode3_ing == 261

* 26101 Potato crisps

* 26102 Other vegetable crisps

* 262 Corn snacks

* 26201 Corn chips

replace nova_group = 4 if digcode5_ing == 26201

replace subgroups = 37 if digcode5_ing == 26201

* 26202 Popcorn

replace nova_group = 4 if digcode5_ing == 26202

replace subgroups = 37 if digcode5_ing == 26202

*Except (natural popcorn)

replace nova_group = 1 if ingcode == 26202001

replace subgroups = 6 if ingcode == 26202001

* 263 Extruded or reformed snacks

replace nova_group = 4 if digcode3_ing == 263

replace subgroups = 37 if digcode3_ing == 263

* 26301 Extruded snacks

* 264 Other snacks

replace nova_group = 4 if digcode3_ing == 264

replace subgroups = 37 if digcode3_ing == 264

* 26401 Other snacks

* 27 Sugar products and dishes

* 271 Sugar, honey and syrups

* 27101 Sugar

replace nova_group = 2 if digcode5_ing == 27101

replace subgroups = 15 if digcode5_ing == 27101

* 27102 Honey and sugar syrups

replace nova_group = 4 if digcode5_ing == 27102

replace subgroups = 36 if digcode5_ing == 27102

*Except honey

replace nova_group = 2 if ingcode == 27102002

replace subgroups = 18 if ingcode == 27102002

* 27103 Toppings, all flavours

replace nova_group = 4 if digcode5_ing == 27103

replace subgroups = 36 if digcode5_ing == 27103

* 272 Jam and lemon spreads, chocolate spreads, sauces

* 27201 Jams and conserves, sugar sweetened

```

replace nova_group = 3 if digcode5_ing == 27201
replace subgroups = 24 if digcode5_ing == 27201
* 27202 Jams and conserves, reduced sugar
replace nova_group = 3 if digcode5_ing == 27202
replace subgroups = 24 if digcode5_ing == 27202
* 27203 Sweet spreads, fruit flavoured
* Disaggregated
* 27204 Sweet spreads or sauces, chocolate/coffee flavoured
replace nova_group = 4 if digcode5_ing == 27204
replace subgroups = 34 if digcode5_ing == 27204
* 27205 Sauces, sweet, fruit-based
replace nova_group = 3 if digcode5_ing == 27205
replace subgroups = 24 if digcode5_ing == 27205

* 273 Dishes and products other than confectionery where sugar is the major component
* 27301 Sugar-based desserts
replace nova_group = 4 if digcode5_ing == 27301
replace subgroups = 39 if digcode5_ing == 27301
* 27302 Sugar-based desserts, intense sweetened
replace nova_group = 4 if digcode5_ing == 27302
replace subgroups = 39 if digcode5_ing == 27302
* 27303 Water ice confection, gelato, sorbet
replace nova_group = 4 if digcode5_ing == 27303
replace subgroups = 30 if digcode5_ing == 27303
* 27304 Frostings and icing
replace nova_group = 4 if digcode5_ing == 27304
replace subgroups = 30 if digcode5_ing == 27304

*****
* 28 Confectionery and cereal/nut/fruit/seed bars
*****

* 281 Chocolate and chocolate-based confectionery
replace nova_group = 4 if digcode3_ing == 281
replace subgroups = 29 if digcode3_ing == 281
* 28101 Chocolate (plain, unfilled varieties)
* 28102 Chocolate-based confectionery with nut fillings or additions
* 28103 Chocolate-based confectionery with other fillings or additions
* 28104 Carob or yoghurt and carob or yoghurt-based confectionery

* 282 Fruit, nut and seed-bars
replace nova_group = 4 if digcode3_ing == 282
replace subgroups = 29 if digcode3_ing == 282
*Except (almond sugar-coated)
replace nova_group = 3 if ingcode == 28202008 | ingcode == 28202001
replace subgroups = 24 if ingcode == 28202008 | ingcode == 28202001
* 28201 Fruit bar and fruit-based confectionery
* 28202 Nut and seed based confectionery

* 283 Muesli or cereal style bars
replace nova_group = 4 if digcode3_ing == 283
replace subgroups = 29 if digcode3_ing == 283
* 28301 Muesli and cereal style bars, no fruit

```

```

*      28302 Muesli and cereal style bars, with fruit and/or nuts
*      28303 Muesli and cereal style bars, added coatings or confectionery
*      28304 Muesli bar, with fruit or fruit paste filling
*      28305 Snack bar, other

*      284    Other confectionery
replace nova_group = 4 if digcode3_ing == 284
replace subgroups = 29 if digcode3_ing == 284
    *Except (ginger crystallised)
replace nova_group = 3 if ingcode == 28405003
replace subgroups = 24 if ingcode == 28405003
*      28401 Lollies and other confectionery, sugar sweetened
*      28402 Lollies and other confectionery, intense sweetened
*      28403 Chewing gum, sugar sweetened
*      28404 Chewing gum, artificially sweetened
*      28405 Other confectionery

*****
*      29    Alcoholic beverages
*****

*      291    Beers
replace nova_group = 3 if digcode3_ing == 291
replace subgroups = 19 if digcode3_ing == 291
*      29101 Beers, > 3.5% alcohol
*      29102 Beers, 1.15- 3.5% alcohol, reduced alcohol / light

*      292    Wines
*      29201 Wines, red (including sparkling varieties and rose styles)
replace nova_group = 3 if digcode5_ing == 29201
replace subgroups = 19 if digcode5_ing == 29201
*      29202 Wines, white (including sparkling varieties)
replace nova_group = 3 if digcode5_ing == 29202
replace subgroups = 19 if digcode5_ing == 29202
*      29203 Fortified wines
replace nova_group = 3 if digcode5_ing == 29203
replace subgroups = 19 if digcode5_ing == 29203
*      29204 Reduced alcohol wines
replace nova_group = 3 if digcode5_ing == 29204
replace subgroups = 19 if digcode5_ing == 29204
    *Except (wine cooler)
replace nova_group = 4 if ingcode == 29204001
replace subgroups = 43 if ingcode == 29204001
*      29205 De-alcoholised and non-alcoholic wine (including sparkling varieties)
replace nova_group = 4 if ingcode == 29205001
replace subgroups = 43 if ingcode == 29205001
replace nova_group = 3 if ingcode == 29205002
replace subgroups = 19 if ingcode == 29205002

*      293    Spirits
replace nova_group = 4 if digcode3_ing == 293
replace subgroups = 43 if digcode3_ing == 293
*      29301 Spirits

*      294    Cider and perry

```

```

replace nova_group = 3 if digcode3_ing == 294
replace subgroups = 19 if digcode3_ing == 294
*   29401 Cider
*   29402 Perry

*   295 Other alcoholic beverages
replace nova_group = 4 if digcode3_ing == 295
replace subgroups = 43 if digcode3_ing == 295
*Except Mirin
replace nova_group = 3 if ingcode == 29505002
replace subgroups = 19 if ingcode == 29505002
*   29501 Liqueurs
*   29502 Cocktails and other mixed drinks
*   29503 Pre-mixed drinks, cola- or energy-drink based
*   29504 Pre-mixed drinks, other
*   29505 Other alcoholic beverages

*****
*   30 Special dietary foods
*****

*   301 Formula dietary foods
replace nova_group = 4 if digcode3_ing == 301
replace subgroups = 45 if digcode3_ing == 301
*   30101 Biscuit and bar meal replacement
*   30102 Meal replacement and similar prepared beverages
*   30103 Meal replacement and similar dry powders
*   30104 Sport and protein prepared beverages
*   30105 Sport and protein, dry powders
*   30106 Supplementary and medical foods prepared beverages
*   30107 Supplementary and medical foods dry powders

*****
*   31 Miscellaneous
*****

*   311 Yeast, and yeast vegetable or meat extracts
*   31101 Yeast
replace nova_group = 2 if digcode5_ing == 31101
replace subgroups = 18 if digcode5_ing == 31101
*   31102 Yeast extracts
replace nova_group = 4 if digcode5_ing == 31102
replace subgroups = 34 if digcode5_ing == 31102
*   31103 Vegetable and meat extracts
replace nova_group = 4 if digcode5_ing == 31103
replace subgroups = 45 if digcode5_ing == 31103

*   312 Intense sweetening agents
replace nova_group = 4 if digcode3_ing == 312
replace subgroups = 45 if digcode3_ing == 312
*   31201 Intense sweeteners

*   313 Herbs, spices, seasonings and stock cubes
*   31301 Salt
replace nova_group = 2 if digcode5_ing == 31301

```

```

replace subgroups = 14 if digcode5_ing == 31301
*   31302 Herbs(dried) and spices
replace nova_group = 1 if digcode5_ing == 31302
replace subgroups = 13 if digcode5_ing == 31302
*   31303 Stock cubes and seasonings
replace nova_group = 4 if digcode5_ing == 31303
replace subgroups = 36 if digcode5_ing == 31303
*   31304 Stock, prepared
*Disaggregated, except (stock commercial)
replace nova_group = 4 if ingcode == 31304003
replace subgroups = 36 if ingcode == 31304003

*   314   Essences
replace nova_group = 4 if digcode3_ing == 314
replace subgroups = 45 if digcode3_ing == 314
*   31401 Essences

*   315   Chemical raising agents and cooking ingredients
*   31501 Chemical raising agents and cooking ingredients
replace nova_group = 2 if digcode5_ing == 31501
replace subgroups = 18 if digcode5_ing == 31501
*   31502 Gelatine
replace nova_group = 4 if digcode5_ing == 31502
replace subgroups = 39 if digcode5_ing == 31502
*   31503 Other additives
replace nova_group = 5 if digcode5_ing == 31503
replace subgroups = . if digcode5_ing == 31503

*****
*   32   Infant formulae and foods
*****

*   321   Infant formulae and human breast milk
replace nova_group = 4 if digcode3_ing == 321
replace subgroups = 35 if digcode3_ing == 321
*Except (human/breast milk)
replace nova_group = 1 if ingcode == 32102001
replace subgroups = 1 if ingcode == 32102001
*   32101 Infant formula, prepared
*   32102 Human breast milk
*   32103 Toddler formula, prepared

*   322   Infant cereal products
replace nova_group = 4 if digcode3_ing == 322
replace subgroups = 27 if digcode3_ing == 322
*   32201 Infant cereals
*   32202 Infant rusks
*   32203 Infant pasta and rice dishes

*   323   Infant foods
replace nova_group = 4 if digcode3_ing == 323
replace subgroups = 26 if digcode3_ing == 323
*   32301 Infant fruit and fruit-based desserts
*   32302 Infant savoury dishes, commercial
*   32303 Infant custards or yoghurts

```



```
*      324    Infant drinks
*      32401  Infant fruit juices
replace nova_group = 4 if digcode3_ing == 324
replace subgroups = 42 if digcode3_ing == 324
```

```
*****
*      34     Reptiles, amphibia and insects
*****
```

```
*      341    Reptiles
replace nova_group = 1 if digcode3_ing == 341
replace subgroups = 13 if digcode3_ing == 341
*      34101  Reptiles
```

```
*      342    Insects
*      34201  Insects
replace nova_group = 1 if digcode3_ing == 342
replace subgroups = 13 if digcode3_ing == 342
```

```
*      343    Amphibia
*      34301  Amphibia
replace nova_group = 1 if digcode3_ing == 343
replace subgroups = 13 if digcode3_ing == 343
```

```
*
*
```

Appendix 3. Sensitivity analyses to account for exposure misclassification of breads**Table S1.** Average nutrient content of the overall diet and of two diet fractions. Australian population aged 2+ years (NNPAS 2011–2012) (n = 12,153)[‡].

Nutrients	Overall diet		Ultra-processed food diet fraction		Non-ultra-processed food diet fraction ^a	
	Mean	(SE)	Mean	(SE)	Mean	(SE)
Free sugars (% of total energy)	11.7	(0.1)	24.3	(0.2)	4.2*	(0.1)
Total fats (% of total energy)	31.1	(0.1)	35.3	(0.2)	28.7*	(0.1)
Saturated fat (% of total energy)	12.0	(0.1)	14.1	(0.1)	10.6*	(0.1)
Trans fat (% of total energy)	0.6	(0.0)	0.6	(0.0)	0.6	(0.0)
Dietary fibre (g/1,000 kcal)	11.5	(0.1)	8.9	(0.1)	13.3*	(0.1)
Sodium (mg/1,000 kcal)	1213	(6.4)	2414	(96.5)	986*	(6.7)
Potassium (mg/1,000 kcal)	1444	(6.6)	1125	(10.9)	1695*	(8.8)
Energy density (kcal/g) ^b	1.7	(0.0)	2.7	(0.0)	1.5*	(0.0)

SE: Standard Error

[‡]Analyses considering 11.6% of total energy intake from processed breads (20.6% of total energy intake from processed foods, 37.2% of total energy intake from ultra-processed foods).^aIt includes NOVA unprocessed or minimally processed foods, processed culinary ingredients and processed foods.^bBeverages excluded.

*p<0.001 for significant mean differences between the two fractions of the diet.

Table S2. Average nutrient content of the overall diet according to quintiles of the dietary share of ultra-processed foods. Australian population aged 2+ years (NNPAS 2011–2012) (n = 12,153)[‡].

Nutrients	Quintiles of the dietary contribution of ultra-processed foods (% of total dietary energy) [‡]					Standardized regression coefficients ^b	
	Q1	Q2	Q3	Q4	Q5	Crude	Adjusted [†]
Free sugars (% of total energy)	5.5	8.9	11.5	14.5	17.9	0.48*	0.46*
Total fats (% of total energy)	29.9	30.4	30.8	31.4	32.7	0.10*	0.10*
Saturated fat (% of total energy)	10.3	11.2	12.1	12.7	13.5	0.24*	0.21*
Trans fat (% of total energy)	0.52	0.56	0.59	0.63	0.65	0.12*	0.11*
Dietary fibre (g/1,000 kcal)	13.4	12.3	11.5	11.0	9.3	-0.25*	-0.22*
Sodium (mg/1,000 kcal)	1080	1160	1212	1280	1338	0.17*	0.17*
Potassium (mg/1,000 kcal)	1654	1534	1463	1388	1181	-0.28*	-0.24*
Energy density (kcal/g) ^c	1.5	1.6	1.7	1.8	2.1	0.40*	0.38*

[‡]Analyses considering 11.6% of total energy intake from processed breads (20.6% of total energy intake from processed foods, 37.2% of total energy intake from ultra-processed foods).

^aPercentage of total energy intake from ultra-processed foods. Mean (range): Q1= 9.6 (0 to 17.6); Q2= 23.6 (17.6 to 29.3); Q3= 35.2 (29.3 to 41.4); Q4= 48.2 (41.4 to 56.3); Q5= 69.5 (56.3 to 100).

^bObtained with linear regression of the dietary nutrient content on the quintiles of the dietary share of ultra-processed foods, and expressed in standard deviation units.

^cBeverages excluded.

[†]Adjusted for age, sex, educational attainment, socio-economic status and geographical location.

*Statistically significant p<0.001

Table S3. Prevalence (%) of non-recommended nutrient intake levels according to quintiles of the dietary share of ultra-processed foods Australian population aged 2+ years (NNPAS 2011–2012) (n = 12,153)[¥].

Quintiles of the dietary contribution of ultra-processed foods (% of total dietary energy) [¥]	Non-recommended nutrient intake levels of:															
	Free sugars		Total fat		Saturated fat		Trans fat		Dietary fibre		Sodium		Potassium		Energy density	
	≥10% of energy		≥30% of energy		≥10% of energy		≥1% of energy		≤12.5g/ 1,000 kcal		≥1,000 mg/ 1,000 kcal		≤1,755mg/ 1,000 kcal		≥2.25kcal/g	
	%	PR	%	PR	%	PR	%	PR	%	PR	%	PR	%	PR	%	PR
Q1 (lowest)	21.1	1.0	51.8	1.0	56.2	1.0	5.8	1.0	55.6	1.0	68.9	1.0	75.4	1.0	2.2	1.0
Q2	45.7	2.1	53.0	1.0	68.7	1.2	4.3	0.8	64.2	1.1	75.7	1.1	83.3	1.1	2.8	1.2
Q3	58.1	2.7	60.7	1.2	76.5	1.3	7.4	1.3	70.5	1.2	80.7	1.2	87.9	1.1	4.6	2.0
Q4	71.6	3.3	65.4	1.2	82.4	1.4	7.5	1.3	75.7	1.3	79.8	1.2	89.9	1.2	8.9	3.4
Q5 (highest)	83.0	3.8‡†	69.1	1.3‡†	85.1	1.5‡†	10.7	1.9‡†	84.9	1.5‡†	84.2	1.2‡†	94.1	1.2‡†	22.7	9.2‡†
Total	55.9	-	60.0	-	73.8	-	7.2	-	70.2	-	77.9	-	86.1	-	8.3	-

[¥]Analyses considering 11.6% of total energy intake from processed breads (20.6% of total energy intake from processed foods, 37.2% of total energy intake from ultra-processed foods).

PR: Prevalence ratios adjusted for age, sex, educational attainment, socio-economic status and geographical location.

‡ Significant linear trend across quintiles (p<0.001) in both unadjusted (coefficients not shown) and adjusted models.

† Wald test at 5% significance level.