Example WIC analysis using R Markdown

by Martin Brown, Martin.Brown@state.or.us

### Introduction

This R Markdown document provides an example application of the Waste Impact Calculator (WIC) framework. Using an actual real-world waste management question from an Oregon city (here renamed Anytown), it demonstrates the basic steps involved in nearly any WIC analysis.

You, the reader, play the part of a waste manager or sustainability analyst from Anytown. You will:

* convert a waste management question into waste streams linked to scenario labels in a ***massProfiles*** table;
* combine your ***massProfiles*** with the ***impactFactors*** table provided by WIC, and filter and summarize the results for each of your scenarios, in multiple impact categories (GHGs, water use, etc.);
* delve into the reasons the scenarios differ – or perhaps, fail to differ much; and finally,
* reflect on what WIC has and hasn’t told you.

Along the way, you’ll find pointers on how to properly combine, filter, and summarize WIC’s data tables, so that the final results actually represent the scenarios you want to evaluate.

### Intended audience

This document is oriented towards more technical users: those familiar with statistical or database operations. This document assumes a beginning-to-intermediate familiarity with the R language, especially the packages “base,” “dplyr”, “rmarkdown”, and “ggplot2”. Though R is used here, R is not specifically required to create results using WIC. No matter what computer setup is involved, the elementary steps will be the same – and you are responsible for your own work! :)

Before using this document, or WIC in general, as the basis for new analyses, you should have a clear understanding of:

* how WIC calculates impacts for individual life cycle stages (see “Prerequisites”, below); and
* how file structures and joining commands (for example *left\_join* and *full\_join* in R’s dplyr package) can create sets of impacts that represent the phases of the materials life cycle.

### Prerequisites

* Read “Technical overview of the Waste Impact Calculator”. This describes the the purpose, limitations, and basic operation of WIC. It also documents the meaning of all the fields in each of WIC’s data tables – information that will, for the most part, not be repeated here.
* Have experience with R as described above.

### Conventions

* the words “table” and “data frame” are used interchangeably. The latter refers to a specific data structure in R, but it is equivalent to a table.
* the words “mass” and “weight” are used interchangeably. The distinction between these terms is important in other fields, but not in Earthbound waste management.
* the term “impact category” is used to describe quantities such as global warming potentials, which are technically “LCIA profiles”, as well as quantities like water use summaries, which are technically “inventory metrics.”
* weights are in short tons

### Anytown’s situation and your expectations of WIC

You work for Anytown’s sustainability department and are concerned about the environmental impacts of Anytown’s municipal food waste.

In particular you are considering implementing a food waste composting scheme, but don’t know if it is worth the effort it would require. You feel that composting food waste is likely to be better on an environmental impact basis than simply landfilling it, but it is unclear how big the improvement would be. Another option is the idea of trying to reduce the sheer generation of food waste through a public education campaign – which would have its own difficulties, but, as an “upstream” strategy, might be more powerful than the “downstream” one of composting.

You will use WIC to compare Anytown’s current treatment of food waste (the “baseline” scenario) with two scenarios for increasing composting of food waste, and two scenarios for reducing food waste generation.

You expect that WIC will estimate life cycle impacts for each scenario, in enough detail that the reasons for differences between scenarios can be understood. At the same time, you understand that WIC is about environmental impacts only – it will not specifically output other relevant information such as the cost of infrastructure or staff.

Therefore WIC is unlikely to be the only basis for Anytown’s final decision about how to proceed. But its environmental impact numbers should play an important part.

### Defining baseline and alternative scenarios

Currently, none of the food waste Anytown collects is composted. Instead, food waste is included in general trash collection and taken to a landfill 178 miles away. However, Anytown’s municipal waste service includes a separate collection of yard debris, which is taken to a site 4 miles away and composted aerobically.

It has been proposed to allow Anytowners to throw food waste in their yard debris bins, so that the materials can be composted together. However, there is a complication: the nearby composting site cannot accept food waste, because it is too close to an airport, and birds attracted to the waste could present a danger to aircraft. The yard debris/food waste mixture will need to be taken to a different site for composting, 77 miles away.

Accordingly, your analysis is really about two materials: yard debris and food waste. Food waste can’t be considered on its own because food waste treatment can affect yard debris’ impacts. Management scenarios will differ in amounts of those two materials going to two end-of-life dispositions (landfilling and composting), as well as end-of-life transport distances.

To establish the tonnages linked to the baseline situation, you must do a a bit of estimating. Anytown knows how many tons of yard debris it picks up and composts each year (9000 tons), but not how much food waste it disposes as part of regular garbage disposal. Anytown does not do its own “waste sort” of disposed materials. However, Anytown is part of a larger metropolitan region, which has information for both food waste and yard debris. Based on the metropolitan region’s studies, you estimate that for every 9000 tons of yard debris composted by Anytown’s system, there are 7669 tons of food waste generated and landfilled as part of Anytown’s garbage.

Accordingly,

* the “baseline” scenario in your ***massProfiles*** is defined by 9000 tons of yard debris going 4 miles to composting, and 7669 tons of food debris going 178 miles to landfill.

To represent the proposed addition of food waste to yard debris, you define two more scenarios:

* “compostFW585”: where 585 tons of food waste are added to yard debris and all composting happens at the new site, 77 miles away; the remainder of food waste is landfilled as usual.
* “compostFW1000”: where 1000 tons food waste are added to yard debris and all composting happens at a new site, 77 miles away; the remainder of food waste is landfilled as usual.

You do not create a scenario where *all* food waste is included with yard debris. You consider it unlikely that all homes and businesses contributing to Anytown’s garbage collection will have the interest and ability to put all their food waste in with yard debris. In fact the compostFW585 scenario is considered most realistic, as it is based on the observed mixture of food waste and yard debris in a nearby city that collects food waste and yard debris together.

As an alternative to increased composting, you also create scenarios representing successful efforts to reduce the generation of food waste. Though this “upstream” solution is potentially powerful, anecdotal experience suggests that reducing food waste generation is not easy. An optimistic example is provided by [the comprehensive WRAP progam in the UK, which reduced household food waste by 6% over 3 years](https://www.wrap.org.uk/sites/files/wrap/Courtauld_Commitment_2025_Milestone_Progress_Report.pdf). Half of that, a 3 percent reduction might be more realistic for Anytown.

The food waste reduction scenarios are:

* “reduceFW03”: no change in management sites or methods, but generation of food waste is reduced by 3 percent to 7439 tons; and
* “reduceFW06”: no change in management sites or method, but generation of food waste is reduced by 6 percent to 7209 tons.

You want to know:

* which scenario is associated with the lowest life cycle impacts, in multiple impact categories?
* in general, what are the reasons that scenarios perform the way they do?
* for example, which materials and life cycle stages represent the biggest part of this system’s associated life cycle impacts?
* and, does the necessity of adding transport distance undermine the benefits of composting?

### Outline of the analysis

Your analysis will proceed in this order:

* preparing the R workspace
* loading in the two source data frames, ***massProfiles*** and ***impactFactors***
* calculating impacts and creating the master results data table, ***impactsInDetail***
* checking for internal consistency of ***impactsInDetail***
* creating summary statistics and graphics using both weight and impact perspectives

### Preparing the R workspace

# checking working directory  
getwd()

## [1] "C:/Users/mbrown2/Documents/Local repositories/wic3/wic-base/exampleAnalysis"

# loading packages useful for the analysis  
library(tidyverse) # many useful functions for data management

## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.3 v purrr 0.3.4  
## v tibble 3.0.6 v dplyr 1.0.4  
## v tidyr 1.1.2 v stringr 1.4.0  
## v readr 1.4.0 v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(ggthemes) # some themes for plotting  
library(scales) # useful functions for labeling charts

##   
## Attaching package: 'scales'

## The following object is masked from 'package:purrr':  
##   
## discard

## The following object is masked from 'package:readr':  
##   
## col\_factor

library(knitr) # helps generate formatted output of various kinds  
library(rmarkdown) # converts RMarkdown documents to other formats  
library(viridis) # nice & accessible color schemes

## Loading required package: viridisLite

##   
## Attaching package: 'viridis'

## The following object is masked from 'package:scales':  
##   
## viridis\_pal

library(svglite) # helps write charts to SVG files

### Loading *massProfiles* and *impactFactors*

As you recall from *Technical overview of the Waste Impact Calculator*, the ***massProfiles*** table describes waste management scenarios by listing, in detail, the mass of each waste material going to specific end-of-life dispositions (e.g. landfilling, recycling), from areas of interest (“wastesheds”), as well as (optionally) setting transport distances for those end-of-life treatments. Different waste management ideas, or “scenarios”, are expressed as different numbers of tons going to different dispositions, and (optionally) different transport distances.

You have prepared a ***massProfiles*** table representing your five scenarios in a CSV file. Here you load it in to an R data frame, and print it out in a formatted table.

# loading the mass profile data into an R data frame  
massProfiles <-  
 read.csv(  
 file = "massProfiles.csv",   
 header = TRUE,   
 stringsAsFactors = FALSE   
 )  
# a formatted printout  
kable(  
 massProfiles,   
 caption="massProfiles for Anytown's analysis of food waste"  
)

massProfiles for Anytown’s analysis of food waste

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **scenario** | **wasteshed** | **material** | **disposition** | **umbDisp** | **tons** | **miles** |
| baseline | Anytown | FoodWaste | landfilling | disposal | 7669 | 178 |
| baseline | Anytown | YardDebris | composting | recovery | 9000 | 4 |
| compostFW585 | Anytown | FoodWaste | composting | recovery | 585 | 77 |
| compostFW585 | Anytown | FoodWaste | landfilling | disposal | 7084 | 77 |
| compostFW585 | Anytown | YardDebris | composting | recovery | 9000 | 77 |
| compostFW1000 | Anytown | FoodWaste | composting | recovery | 1000 | 77 |
| compostFW1000 | Anytown | FoodWaste | landfilling | disposal | 6669 | 178 |
| compostFW1000 | Anytown | YardDebris | composting | recovery | 9000 | 77 |
| reduceFW03 | Anytown | FoodWaste | landfilling | disposal | 7439 | 178 |
| reduceFW03 | Anytown | YardDebris | composting | recovery | 9000 | 4 |
| reduceFW06 | Anytown | FoodWaste | landfilling | disposal | 7209 | 178 |
| reduceFW06 | Anytown | YardDebris | composting | recovery | 9000 | 4 |

Notice how the ***massProfiles*** table, in its compact way, provides ALL the details of the management scenarios you defined earlier. The ***massProfiles*** table is the main place you provide input to the WIC system.

Here you list the fields in ***massProfiles***:

str(massProfiles)

## 'data.frame': 12 obs. of 7 variables:  
## $ scenario : chr "baseline" "baseline" "compostFW585" "compostFW585" ...  
## $ wasteshed : chr "Anytown" "Anytown" "Anytown" "Anytown" ...  
## $ material : chr "FoodWaste" "YardDebris" "FoodWaste" "FoodWaste" ...  
## $ disposition: chr "landfilling" "composting" "composting" "landfilling" ...  
## $ umbDisp : chr "disposal" "recovery" "recovery" "disposal" ...  
## $ tons : int 7669 9000 585 7084 9000 1000 6669 9000 7439 9000 ...  
## $ miles : int 178 4 77 77 77 77 178 77 178 4 ...

As you recall, *tons* is the critical variable. This is a mass of some waste material, in short tons. All the other variables serve to identify or qualify where the *tons* came from, which disposition is being used, etc.

Note that the technical disposition of the material (landfilling or composting) is recorded independently of its legal classification (recovery or disposal) in the field umbDisp. The umbDisp field is provided as a convenience, so that you can calculate weight-based statistics such as diversion rates. However, impacts are always calculated based on the disposition name. The legal classification should have no effect on impact results.

Here are a few weight-based waste statistics for each scenario from ***massProfiles***: the tons of waste generated, the tons of waste recovered, and the weight-based recovery rate, as a fraction.

massProfiles %>%  
 group\_by(scenario) %>%  
 summarize(  
 tonsGenerated=sum(tons),  
 recoveredTons=sum(ifelse(umbDisp=="recovery",tons,0))  
 ) %>%  
 mutate(  
 weightBasedRecoveryRate=round(recoveredTons/tonsGenerated,2)  
 ) %>%  
 kable()

|  |  |  |  |
| --- | --- | --- | --- |
| **scenario** | **tonsGenerated** | **recoveredTons** | **weightBasedRecoveryRate** |
| baseline | 16669 | 9000 | 0.54 |
| compostFW1000 | 16669 | 10000 | 0.60 |
| compostFW585 | 16669 | 9585 | 0.58 |
| reduceFW03 | 16439 | 9000 | 0.55 |
| reduceFW06 | 16209 | 9000 | 0.56 |

It appears that the scenario compostFW1000 has the highest weight-based recovery rate. You are interested in finding out – does that mean compostFW1000 will also represent the smallest life cycle impact? Stay tuned!

WIC’s other source data table is ***impactFactors***. It contains environmental impact magnitudes for standard weights of solid waste materials, classified by disposition and life cycle stage.

Here you load the complete “impactFactors.csv” file, provided with WIC, into an R data frame, which you name ***impactFactorsAll***. Since it is thousands of records long, you print out only a small sample of it to see what it looks like…

impactFactorsAll <-  
 read.csv(  
 file = "../impactFactors/distributable/impactFactors.csv",   
 header = TRUE,   
 stringsAsFactors = FALSE #  
 )  
kable(impactFactorsAll %>% head(20))

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **material** | **LCstage** | **disposition** | **corporateSource** | **impactCategory** | **impactUnits** | **impliedMiles** | **impactCategoryLong** | **impactFactor** | **gabiExportDate** | **wicImportDate** |
| AcceptedOtherSteel | endOfLife | incinerationNoER | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 85.14568 | 2020-11-06 | 2021-02-11 |
| AcceptedOtherSteel | endOfLife | landfilling | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 39.36742 | 2020-11-06 | 2021-02-11 |
| AcceptedOtherSteel | endOfLife | recyclingGeneric | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | -1455.27759 | 2020-11-06 | 2021-02-11 |
| AcceptedOtherSteel | endOfLifeTransport | incinerationNoER | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 41.01883 | 2020-11-06 | 2021-02-11 |
| AcceptedOtherSteel | endOfLifeTransport | landfilling | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 41.01883 | 2020-11-06 | 2021-02-11 |
| AcceptedOtherSteel | endOfLifeTransport | recyclingGeneric | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 41.01883 | 2020-11-06 | 2021-02-11 |
| AcceptedOtherSteel | production | production | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 2213.56286 | 2020-11-06 | 2021-02-11 |
| Aluminum | endOfLife | incinerationNoER | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 142.37133 | 2020-11-06 | 2021-02-11 |
| Aluminum | endOfLife | landfilling | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 39.36742 | 2020-11-06 | 2021-02-11 |
| Aluminum | endOfLife | recyclingGeneric | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | -2267.13376 | 2020-11-06 | 2021-02-11 |
| Aluminum | endOfLifeTransport | incinerationNoER | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 41.01883 | 2020-11-06 | 2021-02-11 |
| Aluminum | endOfLifeTransport | landfilling | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 41.01883 | 2020-11-06 | 2021-02-11 |
| Aluminum | endOfLifeTransport | recyclingGeneric | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 41.01883 | 2020-11-06 | 2021-02-11 |
| Aluminum | production | production | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 4724.99869 | 2020-11-06 | 2021-02-11 |
| AsepticContainers | endOfLife | incinerationER | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 210.11534 | 2020-11-06 | 2021-02-11 |
| AsepticContainers | endOfLife | landfilling | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 388.83016 | 2020-11-06 | 2021-02-11 |
| AsepticContainers | endOfLife | recyclingGeneric | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | -578.46351 | 2020-11-06 | 2021-02-11 |
| AsepticContainers | endOfLifeTransport | incinerationER | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 41.01006 | 2020-11-06 | 2021-02-11 |
| AsepticContainers | endOfLifeTransport | landfilling | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 41.01006 | 2020-11-06 | 2021-02-11 |
| AsepticContainers | endOfLifeTransport | recyclingGeneric | IPCC AR5 | GWP 100 | kg CO2 eq. | 180 | GWP100, excl biogenic carbon | 41.01006 | 2020-11-06 | 2021-02-11 |

Many of these records, and some of the diagnostic fields such as *corporateSource* and *gabiExportDate*, are unnecessary for the current analysis. You cut ***impactFactorsAll*** down to a manageable size – using only the materials and dispositions and impact categories that you need – and save the result as ***impactFactors***.

# learning the materials in the massProfiles table and   
# saving them as a vector of string values  
materialNamesToUse <-  
 massProfiles %>%  
 select(material) %>%  
 distinct() %>%  
 pull(material)  
  
# doing the same thing for disposition names  
dispositionNamesToUse <-  
 massProfiles %>%  
 select(disposition) %>%  
 distinct() %>%  
 pull(disposition)  
# need to keep production impacts too  
dispositionNamesToUse <-  
 c("production",dispositionNamesToUse)   
  
# creating a list of impactCategories to use  
allImpactCategories <-  
 impactFactorsAll %>% select(impactCategory) %>% distinct %>% pull()  
impactCategoriesToUse <-  
 setdiff(  
 allImpactCategories,   
 # what follow are the categories to NOT use  
 c(  
# "GWP 100",  
# "GWP 100 (EpaFcs)",  
 "GWP 100 (Slash)",  
 "GWP 20 (Slash)",  
 "GWP 20 (EpaFcs)",  
 "GWP 20",  
# "Human health particulate air",  
 "Human toxicity, non-cancer",  
 "Human toxicity, cancer"  
 )  
 )  
  
# for the sake of brevity in printouts for this   
# example analysis,  
# limiting the impactFactors to the materials and   
# dispositions in massProfiles, and ten impact   
# categories. In regular usage there is no  
# need to do such filtering -- the impactFactors   
# data frame may be left complete.  
impactFactors <-  
 impactFactorsAll %>%  
 filter(  
 material %in% materialNamesToUse &  
 disposition %in% dispositionNamesToUse &  
 impactCategory %in% impactCategoriesToUse  
 ) %>%  
 # there are also several diagnostic columns that may be   
 # removed for the sake of this example analysis.. see   
 # Technical Overview of the Waste Impact Calculator  
 # for their meaning.  
 select(  
 -corporateSource, -impactCategoryLong,   
 -gabiExportDate, -wicImportDate  
 ) %>%  
 # sorting it for easier reading  
 arrange(impactCategory, material, LCstage, disposition)  
# a formatted printout of 20 random lines  
kable(  
 impactFactors %>% sample\_n(20),  
 caption="20 random lines of Anytown's impactFactors table"  
)

20 random lines of Anytown’s impactFactors table

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **material** | **LCstage** | **disposition** | **impactCategory** | **impactUnits** | **impliedMiles** | **impactFactor** |
| YardDebris | endOfLifeTransport | composting | Energy demand | MJ | 180 | 603.4704492 |
| FoodWaste | endOfLife | landfilling | Water consumption | kg | 180 | -250.7538380 |
| FoodWaste | endOfLifeTransport | landfilling | GWP 100 | kg CO2 eq. | 180 | 41.0188344 |
| YardDebris | production | production | Eutrophication | kg N eq. | 180 | 0.0327298 |
| FoodWaste | endOfLife | landfilling | Eutrophication | kg N eq. | 180 | 1.5840044 |
| YardDebris | endOfLifeTransport | composting | GWP 100 (EpaFcs) | kg CO2 eq. | 180 | 40.8536951 |
| FoodWaste | endOfLife | landfilling | Ecotoxicity | CTUe | 180 | 3.7735100 |
| YardDebris | endOfLifeTransport | composting | Ozone depletion | kg CFC 11 eq. | 180 | 0.0000000 |
| YardDebris | endOfLife | landfilling | GWP 100 (EpaFcs) | kg CO2 eq. | 180 | 819.0801850 |
| FoodWaste | endOfLife | landfilling | GWP 100 | kg CO2 eq. | 180 | 343.6771280 |
| YardDebris | production | production | Energy demand | MJ | 180 | 258.3278333 |
| FoodWaste | endOfLifeTransport | composting | Energy demand | MJ | 180 | 603.4704492 |
| FoodWaste | endOfLifeTransport | landfilling | Human health particulate air | kg PM2.5 eq. | 180 | 0.0048426 |
| FoodWaste | endOfLife | composting | Smog air | kg O3 eq. | 180 | 1.9381548 |
| YardDebris | endOfLifeTransport | composting | GWP 100 | kg CO2 eq. | 180 | 41.0188344 |
| FoodWaste | endOfLifeTransport | landfilling | Energy demand | MJ | 180 | 603.4704492 |
| YardDebris | endOfLifeTransport | landfilling | GWP 100 (EpaFcs) | kg CO2 eq. | 180 | 40.8449565 |
| FoodWaste | endOfLifeTransport | landfilling | Eutrophication | kg N eq. | 180 | 0.0120146 |
| YardDebris | endOfLife | landfilling | Water consumption | kg | 180 | -507.7033219 |
| YardDebris | endOfLife | composting | Acidification | kg SO2 eq. | 180 | 0.9002805 |

As you recall from *Technical overview of the Waste Impact Calculator*, the critical field in this table is *impactFactor*. This number expresses an environmental impact for a particular mass of a particular material in a particular life cycle stage. All the other variables in each record identify or qualify the impact factor somehow – e.g. name the material, label its units, etc.

The ***impactFactors*** data frame should have EXACTLY one record for each combination of material, life cycle stage, disposition, and impactCategory of interest. Though ***impactFactors*** tables provided by Oregon DEQ should have this characteristic, you can check it if you desire, for example like this:

# checking for rows of impactFactors that might be duplicates  
# and printing a summary sentence  
print(  
 paste(  
 "There are ",  
 impactFactors %>%  
 group\_by(material, LCstage, disposition, impactCategory) %>%  
 summarise(myCount=n()) %>% # number of rows in each group  
 filter(myCount != 1) %>% # keep only rows where count <> 1  
 nrow(),  
 " rows in impactFactors that need to be inspected for duplicates.",  
 sep=""  
 )  
)

## `summarise()` has grouped output by 'material', 'LCstage', 'disposition'. You can override using the `.groups` argument.

## [1] "There are 0 rows in impactFactors that need to be inspected for duplicates."

### Merging the two tables to produce *impactsInDetail*.

You must merge ***massProfiles*** and ***impactFactors*** to calculate impacts, but before you do that you must address a limitation of ***massProfiles***. So far ***massProfiles*** only includes tons of materials handled at the end-of-life phase of the life cycle. You must also account for the tons of those materials that are handled at two other life cycle stages: end-of-life transport and production.

You will add tonnages representing production here using a simple copy- and append operation. In the following code, all the cases from ***massProfiles*** are copied, labeled with a *disposition* (and *umbDisp*) of “production,” and then added back to ***massProfiles***, creating a new data frame, ***massProfilesPlus***.

# copy end-of-life tons and label them as production tons  
tempProductionMasses <-  
 massProfiles %>%  
 mutate(  
 disposition="production",  
 umbDisp="production",  
 miles=NA  
 )  
# add the production tons to the end-of-life tons  
massProfilesPlus <-  
 bind\_rows(  
 massProfiles,  
 tempProductionMasses  
 ) %>%  
 # sort the new, larger table  
 arrange(  
 scenario, wasteshed, material, disposition  
 )  
rm(tempProductionMasses) # remove temporary table

The resulting table, ***massProfilesPlus***, should have exactly twice the total tonnage of ***massProfiles***. Moreover, within each *scenario*, production tons should have the same sum as end-of-life tons. This too you can check…

print(  
 paste(  
 "Total tonnage in massProfiles is ",  
 sum(massProfiles$tons),  
 ".",  
 sep=""  
 )  
)

## [1] "Total tonnage in massProfiles is 82655."

print(  
 paste(  
 "Total tonnage in massProfilesPlus is ",  
 sum(massProfilesPlus$tons),  
 ".",  
 sep=""  
 )  
)

## [1] "Total tonnage in massProfilesPlus is 165310."

massProfilesPlus %>%   
 group\_by(scenario) %>%  
 summarise(  
 prodTons=sum(ifelse(umbDisp=="production",tons,0)),  
 eolTons=sum(ifelse(umbDisp!="production",tons,0))  
 ) %>%  
 print()

## # A tibble: 5 x 3  
## scenario prodTons eolTons  
## \* <chr> <dbl> <dbl>  
## 1 baseline 16669 16669  
## 2 compostFW1000 16669 16669  
## 3 compostFW585 16669 16669  
## 4 reduceFW03 16439 16439  
## 5 reduceFW06 16209 16209

The tonnages associated with end-of-life transport are still missing, but you will generate them during the following merge of ***massProfiles*** and ***impactFactors***.

The merge is made on unique combinations of *material* and *disposition* name. However, since ***impactFactors*** has *two* life cycle stages (endOfLifeTransport and endOfLife) in the field *LCstage* for each *disposition* name, records will effectively be added to represent endOfLifeTransport tons.

The merged file has both tons (from the ***massProfiles*** table) and *impactFactor* scaled to tons (from the ***impactFactors*** table), which can then be multiplied to get an impact in units of *impactUnits*.

Like so:

impactsInDetail <-  
 # joining all impact factors relevant to massProfiles  
 left\_join( # important: use left\_join not full\_join  
 massProfilesPlus,  
 impactFactors,  
 by = c("material", "disposition")  
 ) %>%  
 # calculating impacts with special considerations   
 # for end-of-life transport impacts  
 mutate(  
 # if miles is missing replace it with default value  
 miles = ifelse(is.na(miles), impliedMiles, miles),  
 # calculate impact   
 impact =  
 case\_when(  
 LCstage != "endOfLifeTransport" ~ tons\*impactFactor,  
 LCstage == "endOfLifeTransport" ~  
 tons\*(miles/impliedMiles)\*impactFactor  
 )  
 ) %>%  
 arrange(impactCategory, scenario, material, LCstage, disposition)

This creates a data frame, ***impactsInDetail***, with records for each combination of *scenario, wasteshed, material, LCstage, disposition,* and *impactCategory*. A printout of this table is very lengthy, so as a visual check, you print out only the records associated with a single *impactCategory*:

kable(  
 impactsInDetail %>%   
 filter(impactCategory==sample(impactCategoriesToUse, 1)),  
 caption="impactsInDetail for a single impactCategory"  
)

impactsInDetail for a single impactCategory

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **scenario** | **wasteshed** | **material** | **disposition** | **umbDisp** | **tons** | **miles** | **LCstage** | **impactCategory** | **impactUnits** | **impliedMiles** | **impactFactor** | **impact** |
| baseline | Anytown | FoodWaste | landfilling | disposal | 7669 | 178 | endOfLife | Acidification | kg SO2 eq. | 180 | 3.0183015 | 23147.35415 |
| baseline | Anytown | FoodWaste | landfilling | disposal | 7669 | 178 | endOfLifeTransport | Acidification | kg SO2 eq. | 180 | 0.0881314 | 668.36960 |
| baseline | Anytown | FoodWaste | production | production | 7669 | 180 | production | Acidification | kg SO2 eq. | 180 | 15.8027864 | 121191.56916 |
| baseline | Anytown | YardDebris | composting | recovery | 9000 | 4 | endOfLife | Acidification | kg SO2 eq. | 180 | 0.9002805 | 8102.52432 |
| baseline | Anytown | YardDebris | composting | recovery | 9000 | 4 | endOfLifeTransport | Acidification | kg SO2 eq. | 180 | 0.0881314 | 17.62627 |
| baseline | Anytown | YardDebris | production | production | 9000 | 180 | production | Acidification | kg SO2 eq. | 180 | 0.0629918 | 566.92620 |
| compostFW1000 | Anytown | FoodWaste | composting | recovery | 1000 | 77 | endOfLife | Acidification | kg SO2 eq. | 180 | 0.9002805 | 900.28048 |
| compostFW1000 | Anytown | FoodWaste | landfilling | disposal | 6669 | 178 | endOfLife | Acidification | kg SO2 eq. | 180 | 3.0183015 | 20129.05266 |
| compostFW1000 | Anytown | FoodWaste | composting | recovery | 1000 | 77 | endOfLifeTransport | Acidification | kg SO2 eq. | 180 | 0.0881314 | 37.70064 |
| compostFW1000 | Anytown | FoodWaste | landfilling | disposal | 6669 | 178 | endOfLifeTransport | Acidification | kg SO2 eq. | 180 | 0.0881314 | 581.21748 |
| compostFW1000 | Anytown | FoodWaste | production | production | 1000 | 180 | production | Acidification | kg SO2 eq. | 180 | 15.8027864 | 15802.78643 |
| compostFW1000 | Anytown | FoodWaste | production | production | 6669 | 180 | production | Acidification | kg SO2 eq. | 180 | 15.8027864 | 105388.78272 |
| compostFW1000 | Anytown | YardDebris | composting | recovery | 9000 | 77 | endOfLife | Acidification | kg SO2 eq. | 180 | 0.9002805 | 8102.52432 |
| compostFW1000 | Anytown | YardDebris | composting | recovery | 9000 | 77 | endOfLifeTransport | Acidification | kg SO2 eq. | 180 | 0.0881314 | 339.30572 |
| compostFW1000 | Anytown | YardDebris | production | production | 9000 | 180 | production | Acidification | kg SO2 eq. | 180 | 0.0629918 | 566.92620 |
| compostFW585 | Anytown | FoodWaste | composting | recovery | 585 | 77 | endOfLife | Acidification | kg SO2 eq. | 180 | 0.9002805 | 526.66408 |
| compostFW585 | Anytown | FoodWaste | landfilling | disposal | 7084 | 77 | endOfLife | Acidification | kg SO2 eq. | 180 | 3.0183015 | 21381.64778 |
| compostFW585 | Anytown | FoodWaste | composting | recovery | 585 | 77 | endOfLifeTransport | Acidification | kg SO2 eq. | 180 | 0.0881314 | 22.05487 |
| compostFW585 | Anytown | FoodWaste | landfilling | disposal | 7084 | 77 | endOfLifeTransport | Acidification | kg SO2 eq. | 180 | 0.0881314 | 267.07130 |
| compostFW585 | Anytown | FoodWaste | production | production | 585 | 180 | production | Acidification | kg SO2 eq. | 180 | 15.8027864 | 9244.63006 |
| compostFW585 | Anytown | FoodWaste | production | production | 7084 | 180 | production | Acidification | kg SO2 eq. | 180 | 15.8027864 | 111946.93909 |
| compostFW585 | Anytown | YardDebris | composting | recovery | 9000 | 77 | endOfLife | Acidification | kg SO2 eq. | 180 | 0.9002805 | 8102.52432 |
| compostFW585 | Anytown | YardDebris | composting | recovery | 9000 | 77 | endOfLifeTransport | Acidification | kg SO2 eq. | 180 | 0.0881314 | 339.30572 |
| compostFW585 | Anytown | YardDebris | production | production | 9000 | 180 | production | Acidification | kg SO2 eq. | 180 | 0.0629918 | 566.92620 |
| reduceFW03 | Anytown | FoodWaste | landfilling | disposal | 7439 | 178 | endOfLife | Acidification | kg SO2 eq. | 180 | 3.0183015 | 22453.14481 |
| reduceFW03 | Anytown | FoodWaste | landfilling | disposal | 7439 | 178 | endOfLifeTransport | Acidification | kg SO2 eq. | 180 | 0.0881314 | 648.32461 |
| reduceFW03 | Anytown | FoodWaste | production | production | 7439 | 180 | production | Acidification | kg SO2 eq. | 180 | 15.8027864 | 117556.92828 |
| reduceFW03 | Anytown | YardDebris | composting | recovery | 9000 | 4 | endOfLife | Acidification | kg SO2 eq. | 180 | 0.9002805 | 8102.52432 |
| reduceFW03 | Anytown | YardDebris | composting | recovery | 9000 | 4 | endOfLifeTransport | Acidification | kg SO2 eq. | 180 | 0.0881314 | 17.62627 |
| reduceFW03 | Anytown | YardDebris | production | production | 9000 | 180 | production | Acidification | kg SO2 eq. | 180 | 0.0629918 | 566.92620 |
| reduceFW06 | Anytown | FoodWaste | landfilling | disposal | 7209 | 178 | endOfLife | Acidification | kg SO2 eq. | 180 | 3.0183015 | 21758.93547 |
| reduceFW06 | Anytown | FoodWaste | landfilling | disposal | 7209 | 178 | endOfLifeTransport | Acidification | kg SO2 eq. | 180 | 0.0881314 | 628.27962 |
| reduceFW06 | Anytown | FoodWaste | production | production | 7209 | 180 | production | Acidification | kg SO2 eq. | 180 | 15.8027864 | 113922.28740 |
| reduceFW06 | Anytown | YardDebris | composting | recovery | 9000 | 4 | endOfLife | Acidification | kg SO2 eq. | 180 | 0.9002805 | 8102.52432 |
| reduceFW06 | Anytown | YardDebris | composting | recovery | 9000 | 4 | endOfLifeTransport | Acidification | kg SO2 eq. | 180 | 0.0881314 | 17.62627 |
| reduceFW06 | Anytown | YardDebris | production | production | 9000 | 180 | production | Acidification | kg SO2 eq. | 180 | 0.0629918 | 566.92620 |

Note that each line is labeled with the *umbDisp* from ***massProfiles***, so distinctions can be made between recovery and disposal impacts or tonnages if desired.

### Checking the internal consistency of *impactsInDetail*

Before using the ***impactsInDetail*** table to calculate results, some basic quality checks should be performed.

For example, tonnages associated with all life cycles should have the same value within each scenario. That is, within each scenario, tons for “production” should be the same as tons for “endOfLifeTransport” and “endOfLife”. You use code like the following to confirm that:

impactsInDetail %>%  
 group\_by(LCstage, scenario) %>%  
 summarise(tons=sum(tons)) %>%  
 arrange(scenario, LCstage) %>%  
 kable()

## `summarise()` has grouped output by 'LCstage'. You can override using the `.groups` argument.

|  |  |  |
| --- | --- | --- |
| **LCstage** | **scenario** | **tons** |
| endOfLife | baseline | 166690 |
| endOfLifeTransport | baseline | 166690 |
| production | baseline | 166690 |
| endOfLife | compostFW1000 | 166690 |
| endOfLifeTransport | compostFW1000 | 166690 |
| production | compostFW1000 | 166690 |
| endOfLife | compostFW585 | 166690 |
| endOfLifeTransport | compostFW585 | 166690 |
| production | compostFW585 | 166690 |
| endOfLife | reduceFW03 | 164390 |
| endOfLifeTransport | reduceFW03 | 164390 |
| production | reduceFW03 | 164390 |
| endOfLife | reduceFW06 | 162090 |
| endOfLifeTransport | reduceFW06 | 162090 |
| production | reduceFW06 | 162090 |

You note that tonnages in the table above are not identical to tonnages summarized earlier for the the ***massProfiles*** table. Besides the recent addition of production-related tons, and end-of-life transport tons, ***impactsInDetail*** has a complete set of tons for every *impactCategory* in use.

You also check that every record has a value in the *impactFactor* field. No impact factors should be missing, and any impact factors that are exactly zero should be viewed with suspicion (because impact factors of exactly zero are unlikely, and may represent a computation error or lazy assumption). In addition, *impact* and *tons* may be zero but should not be missing. These things can be checked with code like this:

impactsInDetail %>%   
 filter(is.na(impactFactor) | impactFactor==0) %>%  
 nrow()

## [1] 0

impactsInDetail %>%  
 filter(is.na(impact)) %>%  
 nrow()

## [1] 0

impactsInDetail %>%  
 filter(is.na(tons)) %>%  
 nrow()

## [1] 0

In each of these cases, your nrow() call has output 0. This means that your ***impactsInDetail*** table has passed these particular quality checks. If nrow() output >1, then it would be necessary to backtrack and correct something.

When ***impactsInDetail*** fails such simple internal-consistency checks, it is likely to be the result of mismatches between the ***massProfiles*** and ***impactFactors*** tables. Spellings of *material* and *disposition* names must match exactly, and every field in every table (with the exception of the *miles* field) must be filled in with a reasonable value.

### Creating tabular and graphical output

##### Guidelines

The ***impactsInDetail*** data frame is the source of all future output from your analysis. Most results of interest – for example, the total waste tonnages and total impacts linked to each scenario – are the result of simple filter, group, and summation operations on tons or impacts in ***impactsInDetail***.

When creating results from ***impactsInDetail***, you recall that:

* there is much redundancy in this data table now: records representing every combination of scenario, wasteshed, material, LCstage, disposition, and impactCategory. So data must be filtered down to the desired specific content to avoid miscalculation.
* when tons are summed, they should be restricted to tons marked with the “endOfLife” LCstage. The tons that appear in other LCstages are redundant and only serve for the calculation of the impacts of those stages.
* furthermore, when tons are summed, they should be restricted to a single impact category (it should not matter which) – as the complete set of tonnages has been repeated for every impact category.
* impacts should be summed only within a single impactCategory – unless you are willing to create, program, and defend a method for normalizing and/or summarizing across multiple impact categories.

##### Some utility objects

For the purpose of creating charts and tables, a few miscellaneous objects could be useful:

* a plaintext list of material names, sorted in descending order of abundance. (While the current example analysis has only 2 materials, many WIC analyses will be considerably more involved.)
* a table of likely impact category labels. (Impact categories like “Energy demand” do not currently include physical units, such as “MJ” for megajoules. An impact category label would merge those for use on chart axes.)
* a graphical theme for charts
* an ordered list of scenario names

Creating those things…

# most abundant materials in the wastestream, in order  
materialSortOrder <-  
 massProfiles %>%  
 group\_by(material) %>%  
 summarise(tons=sum(tons)) %>%  
 arrange(desc(tons)) %>%  
 pull(material)  
# a table of impact categories combined with units  
# (for use in chart labels)  
impactLabels <-  
 impactFactors %>%  
 select(impactCategory, impactUnits) %>%  
 distinct() %>%  
 mutate(  
 impactLabel =   
 paste(  
 impactCategory,  
 " (",  
 impactUnits,  
 ")",  
 sep=""  
 )  
 )  
# a custom graphic theme for charts, inspired by   
# the fivethirtyeight theme  
theme\_539 <- function() {  
 theme\_fivethirtyeight() +  
 theme(  
 rect=element\_rect(fill="transparent"),  
 panel.grid = element\_blank(),  
 axis.ticks = element\_line()  
 )  
}  
  
# making an ordered list of scenarios, where "baseline" is first  
scenarioOrder <-   
 c(  
 "baseline",  
 setdiff(  
 massProfiles %>%   
 select(scenario) %>%   
 distinct() %>%   
 pull(scenario),  
 "baseline")  
 )

##### Weights of waste in each of the scenarios

Your analysis starts with a review of the weights handled in each scenario. It’s a good way to assure yourself your ***massProfiles*** have been entered accurately.

You recall that when weights are summed, only the “endOfLife” *LCstage* is used, and only a single *impactCategory* is used.

# summing weights by disposition for each scenario  
tempWeightData1 <-   
 impactsInDetail %>%  
 filter(  
 LCstage == "endOfLife" &   
 impactCategory==sample(impactCategoriesToUse,1)  
 ) %>%   
 group\_by(scenario, disposition) %>%   
 summarise(tons=sum(tons)) %>%  
 ungroup() %>%  
 filter(tons != 0) %>%  
 mutate(scenario= factor(scenario, levels=rev(scenarioOrder)))

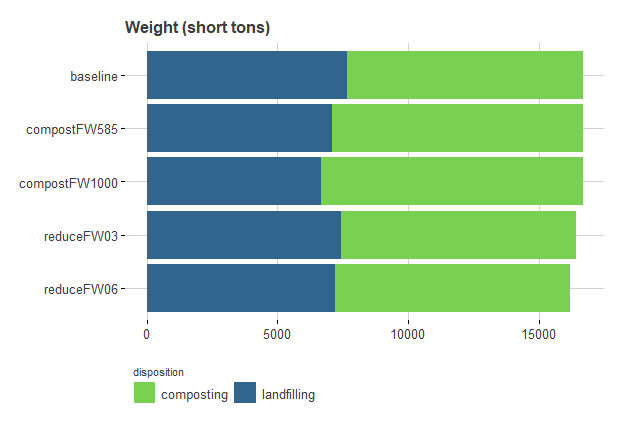
## `summarise()` has grouped output by 'scenario'. You can override using the `.groups` argument.

kable(tempWeightData1)

|  |  |  |
| --- | --- | --- |
| scenario | disposition | tons |
| baseline | composting | 9000 |
| baseline | landfilling | 7669 |
| compostFW1000 | composting | 10000 |
| compostFW1000 | landfilling | 6669 |
| compostFW585 | composting | 9585 |
| compostFW585 | landfilling | 7084 |
| reduceFW03 | composting | 9000 |
| reduceFW03 | landfilling | 7439 |
| reduceFW06 | composting | 9000 |
| reduceFW06 | landfilling | 7209 |

You make that weight data into a chart…

tempWeightChart1 <-  
 ggplot()+  
 ggtitle("Weight (short tons)")+  
 theme\_539()+  
 geom\_bar(  
 data = tempWeightData1,  
 aes(x = scenario, y= tons, fill= disposition),  
 color=NA,  
 stat="identity"  
 )+  
 scale\_fill\_viridis(begin=0.32, end=0.8, discrete = TRUE,   
 direction = -1)+  
 coord\_flip()+  
 guides(fill=guide\_legend(ncol=2, title.position = "top"))+  
 theme(  
 rect=element\_rect(fill="transparent"),  
 plot.title = element\_text(size=12),  
 legend.position="bottom",  
 legend.title = element\_text(size=8),  
 legend.justification="left"  
 )  
# printing the chart to the current device  
tempWeightChart1



# saving the chart as external files  
ggsave("chart\_output/weights.png")

## Saving 6.5 x 4.5 in image

ggsave("chart\_output/weights.svg")

## Saving 6.5 x 4.5 in image

That chart shows you that most of the scenarios are similar in terms of total weight, and management changes between scenarios are modest.

With a bit more coding, you can produce the same chart with individual materials separated.

# summing weights by disposition for each scenario and material  
tempWeightData2 <-   
 impactsInDetail %>%  
 filter(  
 LCstage == "endOfLife" &   
 impactCategory==sample(impactCategoriesToUse, 1)  
 ) %>% # correct set for weight calculations  
 group\_by(scenario, material, disposition) %>%   
 summarise(tons=sum(tons)) %>%  
 ungroup() %>%  
 filter(tons != 0) %>%  
 mutate(scenario= factor(scenario, levels=scenarioOrder))

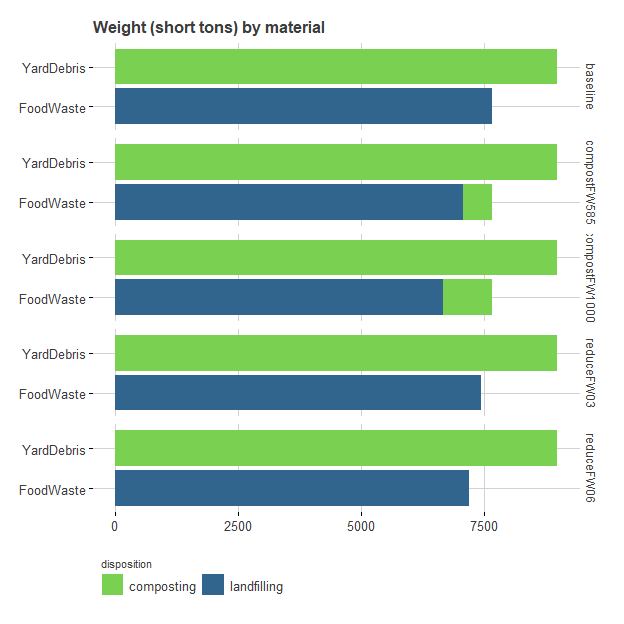
## `summarise()` has grouped output by 'scenario', 'material'. You can override using the `.groups` argument.

kable(tempWeightData2)

|  |  |  |  |
| --- | --- | --- | --- |
| **scenario** | **material** | **disposition** | **tons** |
| baseline | FoodWaste | landfilling | 7669 |
| baseline | YardDebris | composting | 9000 |
| compostFW1000 | FoodWaste | composting | 1000 |
| compostFW1000 | FoodWaste | landfilling | 6669 |
| compostFW1000 | YardDebris | composting | 9000 |
| compostFW585 | FoodWaste | composting | 585 |
| compostFW585 | FoodWaste | landfilling | 7084 |
| compostFW585 | YardDebris | composting | 9000 |
| reduceFW03 | FoodWaste | landfilling | 7439 |
| reduceFW03 | YardDebris | composting | 9000 |
| reduceFW06 | FoodWaste | landfilling | 7209 |
| reduceFW06 | YardDebris | composting | 9000 |

Making a weight chart with materials separated:

tempWeightChart2 <-  
 ggplot()+  
 ggtitle("Weight (short tons) by material")+  
 theme\_539()+  
 geom\_bar(  
 data = tempWeightData2,  
 aes(x = material, y= tons, fill= disposition),  
 color=NA,  
 stat="identity"  
 )+  
 scale\_fill\_viridis(begin=0.32, end=0.8, discrete = TRUE,   
 direction = -1)+  
 coord\_flip()+  
 facet\_grid(scenario~.)+  
 guides(fill=guide\_legend(ncol=2, title.position = "top"))+  
 theme(  
 rect=element\_rect(fill="transparent"),  
 plot.title = element\_text(size=12),  
 legend.position="bottom",  
 legend.title = element\_text(size=8),  
 legend.justification="left",  
 strip.text=element\_text(size=9)  
 )  
# printing the chart to the current device  
tempWeightChart2



# saving the chart as external files  
ggsave("chart\_output/weightsInd.png")

## Saving 6.5 x 6.5 in image

ggsave("chart\_output/weightsInd.svg")

## Saving 6.5 x 6.5 in image

This chart shows you your ***massProfiles*** have described your scenarios fairly. Yard debris does not change its weight throughout all scenarios, while the treatment of food waste varies somewhat.

##### Life cycle impacts for waste in each scenario

Now for comparison, you look at the impacts associated with those scenarios. But here, output will be voluminous, since you have a large number of impact categories to consider. All your choices are in this list: GWP 100, GWP 100 (EpaFcs), Acidification, Ecotoxicity, Eutrophication, Human health particulate air, Ozone depletion, Smog air, Energy demand, Water consumption.

You start by summing up the impacts in similar detail to the first weight chart:

tempImpactData1 <-  
 impactsInDetail %>%  
 group\_by(scenario, impactCategory, impactUnits) %>%  
 summarise(impact=sum(impact)) %>%  
 ungroup() %>%  
 mutate(  
 scenario = factor(scenario, levels = rev(scenarioOrder)),  
 impactLabel =   
 paste(  
 impactCategory,  
 " (",  
 impactUnits,  
 ")",  
 sep=""  
 )  
 )

## `summarise()` has grouped output by 'scenario', 'impactCategory'. You can override using the `.groups` argument.

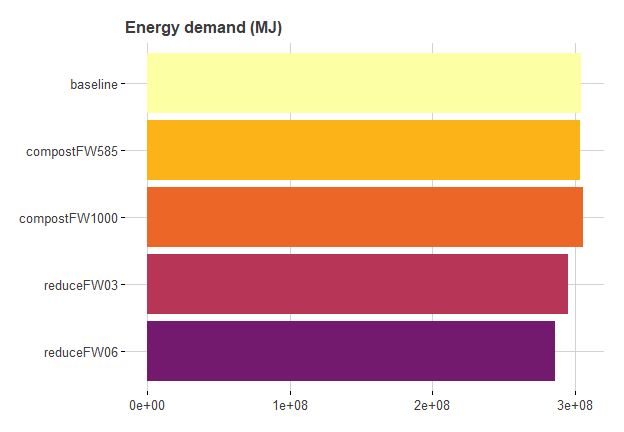
kable(  
 tempImpactData1,  
 caption="summed impacts for each scenario and impactCategory"  
)

summed impacts for each scenario and impactCategory

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **scenario** | **impactCategory** | **impactUnits** | **impact** | **impactLabel** |
| baseline | Acidification | kg SO2 eq. | 1.536944e+05 | Acidification (kg SO2 eq.) |
| baseline | Ecotoxicity | CTUe | 1.171997e+08 | Ecotoxicity (CTUe) |
| baseline | Energy demand | MJ | 3.040130e+08 | Energy demand (MJ) |
| baseline | Eutrophication | kg N eq. | 1.387923e+05 | Eutrophication (kg N eq.) |
| baseline | GWP 100 | kg CO2 eq. | 1.601632e+07 | GWP 100 (kg CO2 eq.) |
| baseline | GWP 100 (EpaFcs) | kg CO2 eq. | 1.059028e+07 | GWP 100 (EpaFcs) (kg CO2 eq.) |
| baseline | Human health particulate air | kg PM2.5 eq. | 1.584823e+04 | Human health particulate air (kg PM2.5 eq.) |
| baseline | Ozone depletion | kg CFC 11 eq. | 7.256287e-01 | Ozone depletion (kg CFC 11 eq.) |
| baseline | Smog air | kg O3 eq. | 7.646943e+05 | Smog air (kg O3 eq.) |
| baseline | Water consumption | kg | 1.004051e+09 | Water consumption (kg) |
| compostFW1000 | Acidification | kg SO2 eq. | 1.518486e+05 | Acidification (kg SO2 eq.) |
| compostFW1000 | Ecotoxicity | CTUe | 1.193495e+08 | Ecotoxicity (CTUe) |
| compostFW1000 | Energy demand | MJ | 3.050143e+08 | Energy demand (MJ) |
| compostFW1000 | Eutrophication | kg N eq. | 1.373021e+05 | Eutrophication (kg N eq.) |
| compostFW1000 | GWP 100 | kg CO2 eq. | 1.572895e+07 | GWP 100 (kg CO2 eq.) |
| compostFW1000 | GWP 100 (EpaFcs) | kg CO2 eq. | 1.063685e+07 | GWP 100 (EpaFcs) (kg CO2 eq.) |
| compostFW1000 | Human health particulate air | kg PM2.5 eq. | 1.585763e+04 | Human health particulate air (kg PM2.5 eq.) |
| compostFW1000 | Ozone depletion | kg CFC 11 eq. | 7.256287e-01 | Ozone depletion (kg CFC 11 eq.) |
| compostFW1000 | Smog air | kg O3 eq. | 7.664987e+05 | Smog air (kg O3 eq.) |
| compostFW1000 | Water consumption | kg | 1.004794e+09 | Water consumption (kg) |
| compostFW585 | Acidification | kg SO2 eq. | 1.523978e+05 | Acidification (kg SO2 eq.) |
| compostFW585 | Ecotoxicity | CTUe | 1.184438e+08 | Ecotoxicity (CTUe) |
| compostFW585 | Energy demand | MJ | 3.031141e+08 | Energy demand (MJ) |
| compostFW585 | Eutrophication | kg N eq. | 1.378910e+05 | Eutrophication (kg N eq.) |
| compostFW585 | GWP 100 | kg CO2 eq. | 1.574730e+07 | GWP 100 (kg CO2 eq.) |
| compostFW585 | GWP 100 (EpaFcs) | kg CO2 eq. | 1.051702e+07 | GWP 100 (EpaFcs) (kg CO2 eq.) |
| compostFW585 | Human health particulate air | kg PM2.5 eq. | 1.584181e+04 | Human health particulate air (kg PM2.5 eq.) |
| compostFW585 | Ozone depletion | kg CFC 11 eq. | 7.256287e-01 | Ozone depletion (kg CFC 11 eq.) |
| compostFW585 | Smog air | kg O3 eq. | 7.609366e+05 | Smog air (kg O3 eq.) |
| compostFW585 | Water consumption | kg | 1.004219e+09 | Water consumption (kg) |
| reduceFW03 | Acidification | kg SO2 eq. | 1.493455e+05 | Acidification (kg SO2 eq.) |
| reduceFW03 | Ecotoxicity | CTUe | 1.142750e+08 | Ecotoxicity (CTUe) |
| reduceFW03 | Energy demand | MJ | 2.948222e+08 | Energy demand (MJ) |
| reduceFW03 | Eutrophication | kg N eq. | 1.346540e+05 | Eutrophication (kg N eq.) |
| reduceFW03 | GWP 100 | kg CO2 eq. | 1.552105e+07 | GWP 100 (kg CO2 eq.) |
| reduceFW03 | GWP 100 (EpaFcs) | kg CO2 eq. | 1.038519e+07 | GWP 100 (EpaFcs) (kg CO2 eq.) |
| reduceFW03 | Human health particulate air | kg PM2.5 eq. | 1.538281e+04 | Human health particulate air (kg PM2.5 eq.) |
| reduceFW03 | Ozone depletion | kg CFC 11 eq. | 7.051661e-01 | Ozone depletion (kg CFC 11 eq.) |
| reduceFW03 | Smog air | kg O3 eq. | 7.425039e+05 | Smog air (kg O3 eq.) |
| reduceFW03 | Water consumption | kg | 9.736875e+08 | Water consumption (kg) |
| reduceFW06 | Acidification | kg SO2 eq. | 1.449966e+05 | Acidification (kg SO2 eq.) |
| reduceFW06 | Ecotoxicity | CTUe | 1.113503e+08 | Ecotoxicity (CTUe) |
| reduceFW06 | Energy demand | MJ | 2.856313e+08 | Energy demand (MJ) |
| reduceFW06 | Eutrophication | kg N eq. | 1.305157e+05 | Eutrophication (kg N eq.) |
| reduceFW06 | GWP 100 | kg CO2 eq. | 1.502578e+07 | GWP 100 (kg CO2 eq.) |
| reduceFW06 | GWP 100 (EpaFcs) | kg CO2 eq. | 1.018009e+07 | GWP 100 (EpaFcs) (kg CO2 eq.) |
| reduceFW06 | Human health particulate air | kg PM2.5 eq. | 1.491739e+04 | Human health particulate air (kg PM2.5 eq.) |
| reduceFW06 | Ozone depletion | kg CFC 11 eq. | 6.847034e-01 | Ozone depletion (kg CFC 11 eq.) |
| reduceFW06 | Smog air | kg O3 eq. | 7.203135e+05 | Smog air (kg O3 eq.) |
| reduceFW06 | Water consumption | kg | 9.433244e+08 | Water consumption (kg) |

Then you examine total impact results for a single impact category chosen at random – by making an impact chart similar in form to the to the previous weight charts:

# chose a single impactCategory at random  
tempImpactCat0 <-  
 impactLabels %>% sample\_n(1)  
tempImpactCat <-  
 tempImpactCat0 %>% pull(impactCategory)  
tempImpactLabel <-  
 tempImpactCat0 %>% pull(impactLabel)  
  
# get the impacts for that category  
tempImpactChart1 <-  
 ggplot()+  
 ggtitle(tempImpactLabel)+  
 theme\_539()+  
 geom\_bar(  
 data =   
 tempImpactData1 %>%   
 filter(impactCategory==tempImpactCat),  
 aes(x = scenario, y= impact, fill=scenario),  
 color=NA,  
 stat="identity"  
 )+  
 scale\_fill\_viridis(begin=0.32, end=1, discrete = TRUE, option="B")+  
 coord\_flip()+  
 guides(fill=guide\_legend(ncol=2, title.position = "top"))+  
 theme(  
 rect=element\_rect(fill="transparent"),  
 plot.title = element\_text(size=12),  
 legend.position="none",  
 legend.title = element\_text(size=8),  
 legend.justification="left"  
 )  
tempImpactChart1



ggsave("chart\_output/impacts1.png")

## Saving 6.5 x 4.5 in image

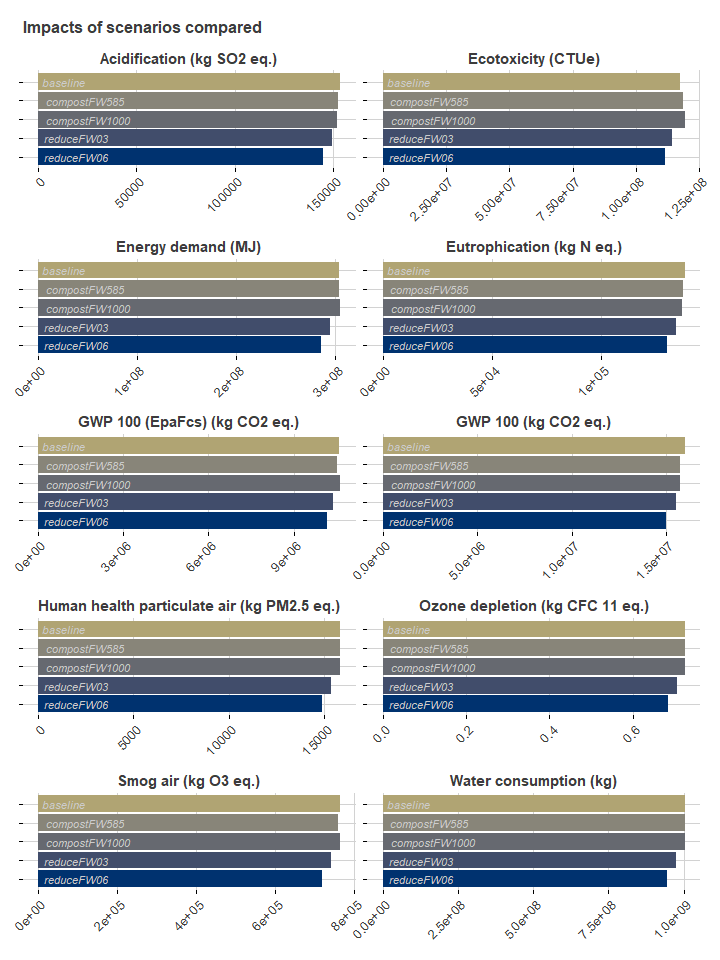
ggsave("chart\_output/impacts1.svg")

## Saving 6.5 x 4.5 in image

These are the net life cycle impacts for Anytown’s combined food waste and yard debris in the impact category Energy demand. All scenarios have similar results, but reducing food waste represents a bigger reduction in impact, compared to baseline, than composting.

But this is only for a single impact category. You make a bigger display to calculate the results for all impact categories on a single 8.5x11 page (with 1 inch margins).

tempImpactChart1a <-  
 ggplot()+  
 ggtitle("Impacts of scenarios compared")+  
 theme\_539()+  
 geom\_bar(  
 data = tempImpactData1,  
 aes(x = scenario, y= impact, fill=scenario),  
 color=NA,  
 # size=2,  
 stat="identity"  
 )+  
 geom\_text(  
 data=tempImpactData1,  
 aes(x=scenario, y=0, label=scenario),  
 color="gray80",  
 size=3,  
 fontface="italic",  
 hjust=-0.1  
 )+  
 facet\_wrap(~impactLabel, ncol=2, scales="free")+  
 scale\_fill\_viridis(  
 begin=0.1, end=0.7, discrete = TRUE, option="cividis"  
 )+  
 coord\_flip()+  
# guides(fill=guide\_legend(ncol=2, title.position = "top"))+  
 theme(  
 rect=element\_rect(fill="transparent"),  
 plot.title = element\_text(size=12),  
 legend.position="none",  
 axis.text.x=element\_text(angle=45, hjust=1),  
 axis.text.y=element\_blank(),  
 strip.text = element\_text(size=11, face="bold")  
 )  
tempImpactChart1a



ggsave("chart\_output/impacts2.png")

## Saving 7.5 x 10 in image

ggsave("chart\_output/impacts2.svg")

## Saving 7.5 x 10 in image

These results are somewhat repetitive. In most or all impact categories, total impacts do not differ much between scenarios. Composting food waste sometimes adds a bit to, and sometimes subtracts a bit from, the baseline impacts. Reducing food waste tends to reduce impacts in all categories, by a somewhat larger but not breathtaking quantity.

You go for a more space-efficient expression of the same set of results, using a “heatmap” where all impacts are expressed in percents, where the baseline scenario is defined as 100%.

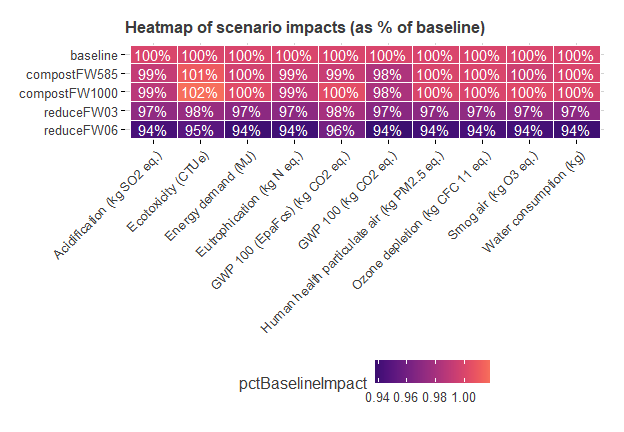
tempImpactData3 <-  
 tempImpactData1 %>%   
 filter(scenario=="baseline") %>%  
 select(impactLabel, impact) %>%  
 rename(baselineImpact=impact)  
tempImpactData3a <-  
 left\_join(  
 tempImpactData1,  
 tempImpactData3,  
 by= c("impactLabel")  
 ) %>%  
 mutate(  
 pctBaselineImpact=impact/baselineImpact  
 )  
kable(tempImpactData3a, caption="heatmap data set")

heatmap data set

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **scenario** | **impactCategory** | **impactUnits** | **impact** | **impactLabel** | **baselineImpact** | **pctBaselineImpact** |
| baseline | Acidification | kg SO2 eq. | 1.536944e+05 | Acidification (kg SO2 eq.) | 1.536944e+05 | 1.0000000 |
| baseline | Ecotoxicity | CTUe | 1.171997e+08 | Ecotoxicity (CTUe) | 1.171997e+08 | 1.0000000 |
| baseline | Energy demand | MJ | 3.040130e+08 | Energy demand (MJ) | 3.040130e+08 | 1.0000000 |
| baseline | Eutrophication | kg N eq. | 1.387923e+05 | Eutrophication (kg N eq.) | 1.387923e+05 | 1.0000000 |
| baseline | GWP 100 | kg CO2 eq. | 1.601632e+07 | GWP 100 (kg CO2 eq.) | 1.601632e+07 | 1.0000000 |
| baseline | GWP 100 (EpaFcs) | kg CO2 eq. | 1.059028e+07 | GWP 100 (EpaFcs) (kg CO2 eq.) | 1.059028e+07 | 1.0000000 |
| baseline | Human health particulate air | kg PM2.5 eq. | 1.584823e+04 | Human health particulate air (kg PM2.5 eq.) | 1.584823e+04 | 1.0000000 |
| baseline | Ozone depletion | kg CFC 11 eq. | 7.256287e-01 | Ozone depletion (kg CFC 11 eq.) | 7.256287e-01 | 1.0000000 |
| baseline | Smog air | kg O3 eq. | 7.646943e+05 | Smog air (kg O3 eq.) | 7.646943e+05 | 1.0000000 |
| baseline | Water consumption | kg | 1.004051e+09 | Water consumption (kg) | 1.004051e+09 | 1.0000000 |
| compostFW1000 | Acidification | kg SO2 eq. | 1.518486e+05 | Acidification (kg SO2 eq.) | 1.536944e+05 | 0.9879905 |
| compostFW1000 | Ecotoxicity | CTUe | 1.193495e+08 | Ecotoxicity (CTUe) | 1.171997e+08 | 1.0183437 |
| compostFW1000 | Energy demand | MJ | 3.050143e+08 | Energy demand (MJ) | 3.040130e+08 | 1.0032936 |
| compostFW1000 | Eutrophication | kg N eq. | 1.373021e+05 | Eutrophication (kg N eq.) | 1.387923e+05 | 0.9892625 |
| compostFW1000 | GWP 100 | kg CO2 eq. | 1.572895e+07 | GWP 100 (kg CO2 eq.) | 1.601632e+07 | 0.9820582 |
| compostFW1000 | GWP 100 (EpaFcs) | kg CO2 eq. | 1.063685e+07 | GWP 100 (EpaFcs) (kg CO2 eq.) | 1.059028e+07 | 1.0043972 |
| compostFW1000 | Human health particulate air | kg PM2.5 eq. | 1.585763e+04 | Human health particulate air (kg PM2.5 eq.) | 1.584823e+04 | 1.0005930 |
| compostFW1000 | Ozone depletion | kg CFC 11 eq. | 7.256287e-01 | Ozone depletion (kg CFC 11 eq.) | 7.256287e-01 | 1.0000000 |
| compostFW1000 | Smog air | kg O3 eq. | 7.664987e+05 | Smog air (kg O3 eq.) | 7.646943e+05 | 1.0023596 |
| compostFW1000 | Water consumption | kg | 1.004794e+09 | Water consumption (kg) | 1.004051e+09 | 1.0007408 |
| compostFW585 | Acidification | kg SO2 eq. | 1.523978e+05 | Acidification (kg SO2 eq.) | 1.536944e+05 | 0.9915637 |
| compostFW585 | Ecotoxicity | CTUe | 1.184438e+08 | Ecotoxicity (CTUe) | 1.171997e+08 | 1.0106151 |
| compostFW585 | Energy demand | MJ | 3.031141e+08 | Energy demand (MJ) | 3.040130e+08 | 0.9970433 |
| compostFW585 | Eutrophication | kg N eq. | 1.378910e+05 | Eutrophication (kg N eq.) | 1.387923e+05 | 0.9935056 |
| compostFW585 | GWP 100 | kg CO2 eq. | 1.574730e+07 | GWP 100 (kg CO2 eq.) | 1.601632e+07 | 0.9832034 |
| compostFW585 | GWP 100 (EpaFcs) | kg CO2 eq. | 1.051702e+07 | GWP 100 (EpaFcs) (kg CO2 eq.) | 1.059028e+07 | 0.9930819 |
| compostFW585 | Human health particulate air | kg PM2.5 eq. | 1.584181e+04 | Human health particulate air (kg PM2.5 eq.) | 1.584823e+04 | 0.9995951 |
| compostFW585 | Ozone depletion | kg CFC 11 eq. | 7.256287e-01 | Ozone depletion (kg CFC 11 eq.) | 7.256287e-01 | 1.0000000 |
| compostFW585 | Smog air | kg O3 eq. | 7.609366e+05 | Smog air (kg O3 eq.) | 7.646943e+05 | 0.9950860 |
| compostFW585 | Water consumption | kg | 1.004219e+09 | Water consumption (kg) | 1.004051e+09 | 1.0001680 |
| reduceFW03 | Acidification | kg SO2 eq. | 1.493455e+05 | Acidification (kg SO2 eq.) | 1.536944e+05 | 0.9717043 |
| reduceFW03 | Ecotoxicity | CTUe | 1.142750e+08 | Ecotoxicity (CTUe) | 1.171997e+08 | 0.9750455 |
| reduceFW03 | Energy demand | MJ | 2.948222e+08 | Energy demand (MJ) | 3.040130e+08 | 0.9697682 |
| reduceFW03 | Eutrophication | kg N eq. | 1.346540e+05 | Eutrophication (kg N eq.) | 1.387923e+05 | 0.9701834 |
| reduceFW03 | GWP 100 | kg CO2 eq. | 1.552105e+07 | GWP 100 (kg CO2 eq.) | 1.601632e+07 | 0.9690773 |
| reduceFW03 | GWP 100 (EpaFcs) | kg CO2 eq. | 1.038519e+07 | GWP 100 (EpaFcs) (kg CO2 eq.) | 1.059028e+07 | 0.9806335 |
| reduceFW03 | Human health particulate air | kg PM2.5 eq. | 1.538281e+04 | Human health particulate air (kg PM2.5 eq.) | 1.584823e+04 | 0.9706328 |
| reduceFW03 | Ozone depletion | kg CFC 11 eq. | 7.051661e-01 | Ozone depletion (kg CFC 11 eq.) | 7.256287e-01 | 0.9718001 |
| reduceFW03 | Smog air | kg O3 eq. | 7.425039e+05 | Smog air (kg O3 eq.) | 7.646943e+05 | 0.9709813 |
| reduceFW03 | Water consumption | kg | 9.736875e+08 | Water consumption (kg) | 1.004051e+09 | 0.9697594 |
| reduceFW06 | Acidification | kg SO2 eq. | 1.449966e+05 | Acidification (kg SO2 eq.) | 1.536944e+05 | 0.9434085 |
| reduceFW06 | Ecotoxicity | CTUe | 1.113503e+08 | Ecotoxicity (CTUe) | 1.171997e+08 | 0.9500909 |
| reduceFW06 | Energy demand | MJ | 2.856313e+08 | Energy demand (MJ) | 3.040130e+08 | 0.9395364 |
| reduceFW06 | Eutrophication | kg N eq. | 1.305157e+05 | Eutrophication (kg N eq.) | 1.387923e+05 | 0.9403668 |
| reduceFW06 | GWP 100 | kg CO2 eq. | 1.502578e+07 | GWP 100 (kg CO2 eq.) | 1.601632e+07 | 0.9381547 |
| reduceFW06 | GWP 100 (EpaFcs) | kg CO2 eq. | 1.018009e+07 | GWP 100 (EpaFcs) (kg CO2 eq.) | 1.059028e+07 | 0.9612670 |
| reduceFW06 | Human health particulate air | kg PM2.5 eq. | 1.491739e+04 | Human health particulate air (kg PM2.5 eq.) | 1.584823e+04 | 0.9412655 |
| reduceFW06 | Ozone depletion | kg CFC 11 eq. | 6.847034e-01 | Ozone depletion (kg CFC 11 eq.) | 7.256287e-01 | 0.9436003 |
| reduceFW06 | Smog air | kg O3 eq. | 7.203135e+05 | Smog air (kg O3 eq.) | 7.646943e+05 | 0.9419627 |
| reduceFW06 | Water consumption | kg | 9.433244e+08 | Water consumption (kg) | 1.004051e+09 | 0.9395188 |

now, to make that into a chart:

tempImpactChart3 <-  
 ggplot()+  
 ggtitle("Heatmap of scenario impacts (as % of baseline)")+  
 theme\_539()+  
 geom\_tile(  
 data=tempImpactData3a,  
 aes(y=scenario, x=impactLabel, fill=pctBaselineImpact),  
 color="white"  
 )+  
 geom\_text(  
 data=tempImpactData3a,  
 aes(  
 y=scenario, x=impactLabel, label=percent(pctBaselineImpact,1)  
 ),  
 color="white"  
 )+  
 scale\_fill\_viridis(begin=0.2, end=0.7, option="A")+  
 theme(  
 plot.title = element\_text(size=12),  
 rect=element\_rect(fill="transparent"),  
 panel.grid = element\_blank(),  
 axis.ticks = element\_line(),  
 axis.text.x = element\_text(hjust=1, angle=45)  
 )  
tempImpactChart3



ggsave("chart\_output/impacts4.png")

## Saving 6.5 x 4.5 in image

ggsave("chart\_output/impacts4.svg")

## Saving 6.5 x 4.5 in image

This display shows you that for many impact categories, baseline, compostFW585 and compostFW1000 have very similar impacts. The exceptions are the two GWP categories, which represent 1-2% decreases in impact through composting, and ecotoxicity, where impacts actually go up through additional composting.

Meanwhile, the reduceFW03 and reduceFW06 scenarios show declines in impact, compared to baseline, that are very similar to the amounts of food waste reduced – 3 and 6 percent respectively.

This suggests that food waste generation is dominating the impacts of the whole system. To check this, you display the impacts of materials and life cycle stages individually.

# calculating detailed impacts by scenario, material, and LCstage  
tempImpactData5 <-  
 impactsInDetail %>%  
 group\_by(  
 scenario, material, LCstage, impactCategory, impactUnits  
 ) %>%  
 summarise(impact=sum(impact)) %>%  
 ungroup() %>%  
 mutate(  
 scenario = factor(scenario, levels = (scenarioOrder)),  
 impactLabel =   
 paste(  
 impactCategory,  
 " (",  
 impactUnits,  
 ")",  
 sep=""  
 ),  
 LCstage=  
 factor(  
 LCstage,  
 levels=rev(c("production","endOfLife","endOfLifeTransport"))  
 )  
 )

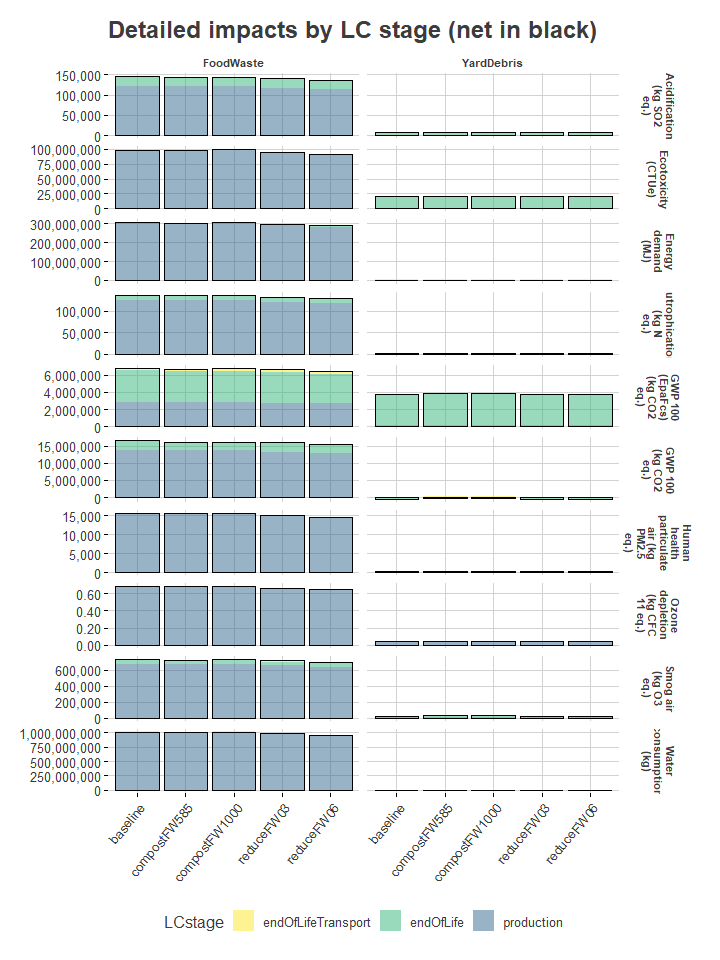
## `summarise()` has grouped output by 'scenario', 'material', 'LCstage', 'impactCategory'. You can override using the `.groups` argument.

# calculating the net totals by scenario and material  
tempImpactData5b <-  
 tempImpactData5 %>%  
 group\_by(scenario, material, impactCategory, impactUnits,  
 impactLabel) %>%  
 summarize(impact=sum(impact))

## `summarise()` has grouped output by 'scenario', 'material', 'impactCategory', 'impactUnits'. You can override using the `.groups` argument.

Now making the chart…

tempImpactChart5 <-  
 ggplot()+  
 ggtitle("Detailed impacts by LC stage (net in black)")+  
 theme\_539()+  
 geom\_bar(  
 data=tempImpactData5,  
 aes(x=scenario, y=impact, fill=LCstage),  
 color=NA,  
 stat="identity",  
 alpha=0.5  
 )+  
 geom\_bar(  
 data=tempImpactData5b,  
 aes(x=scenario, y=impact),  
 stat="identity",  
 fill=NA,  
 color="black"  
 )+  
 scale\_y\_continuous(labels=comma)+  
 #coord\_flip()+  
 facet\_grid(  
 facets = c("impactLabel","material"),   
 scales="free\_y",  
 labeller = labeller(impactLabel=label\_wrap\_gen(10))  
 )+  
 scale\_fill\_viridis(begin=0.32, end=1, discrete = TRUE,   
 direction = -1)+  
 theme(  
 axis.text.x=element\_text(angle=50, hjust=1, vjust=1),  
 panel.background=element\_blank(),  
 panel.grid = element\_blank(),  
 strip.text=element\_text(size=8, face="bold")  
 )  
tempImpactChart5



# saving the chart to a disk file  
ggsave(filename = "chart\_output/impacts5.png")

## Saving 7.5 x 10 in image

ggsave(filename = "chart\_output/impacts5.svg")

## Saving 7.5 x 10 in image

This chart shows you several important things:

* despite the fact that food waste has approximately the same total weight as yard debris (see earlier charts based only on weight), their life cycle impacts are very different.
* in most impact categories, it is the production phase of the materials life cycle for food waste that is dominating the impact total;
* transport impacts are usually too small to even register in this display.

The food waste reduction scenarios reduce impacts more than the composting scenarios because they reduce the amount of food waste WIC assumes is produced.

So, did composting have any benefit at all? You can check our ***impactsInDetail*** table to make sure that there are some negative impacts associated with composting…

# list some negative entries (impact reductions) associated  
# with composting in the current impactsInDetail table  
impactsInDetail %>%  
 filter(LCstage=="endOfLife" &   
 disposition == "composting" &  
 impact < 0  
 ) %>%  
 # removing a few fields to make printout narrower  
 select(-wasteshed, -impliedMiles, -umbDisp) %>%   
 sample\_n(10) %>%  
 kable()

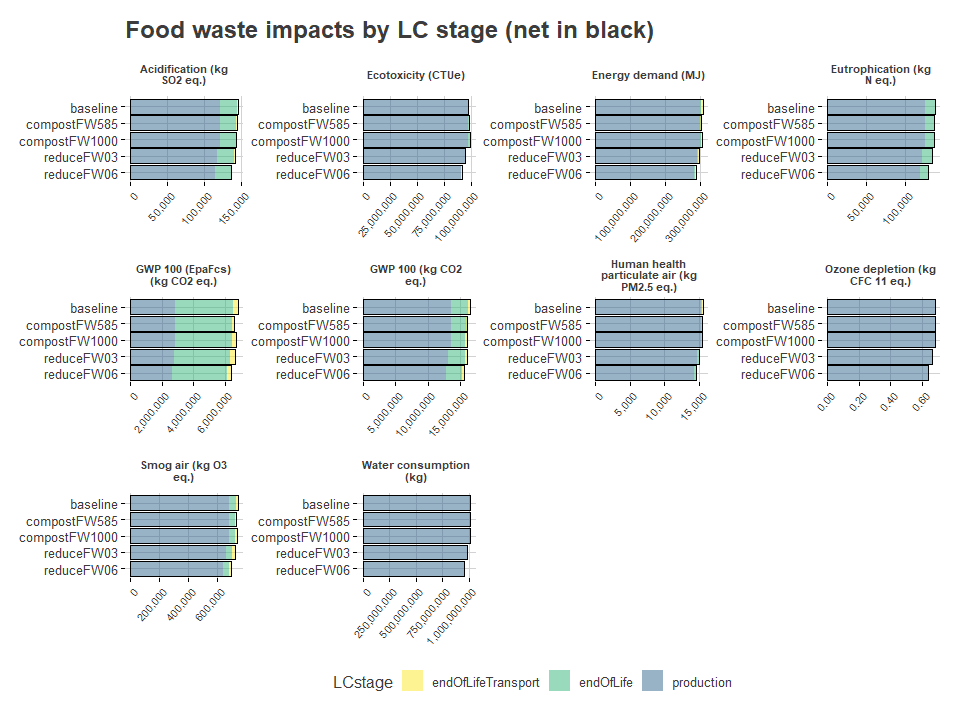
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **scenario** | **material** | **disposition** | **tons** | **miles** | **LCstage** | **impactCategory** | **impactUnits** | **impactFactor** | **impact** |
| compostFW585 | FoodWaste | composting | 585 | 77 | endOfLife | Ozone depletion | kg CFC 11 eq. | 0.00000 | 0.00 |
| compostFW1000 | FoodWaste | composting | 1000 | 77 | endOfLife | Ozone depletion | kg CFC 11 eq. | 0.00000 | 0.00 |
| reduceFW06 | YardDebris | composting | 9000 | 4 | endOfLife | GWP 100 | kg CO2 eq. | -70.38751 | -633487.57 |
| compostFW1000 | FoodWaste | composting | 1000 | 77 | endOfLife | GWP 100 | kg CO2 eq. | -70.38751 | -70387.51 |
| reduceFW03 | YardDebris | composting | 9000 | 4 | endOfLife | Ozone depletion | kg CFC 11 eq. | 0.00000 | 0.00 |
| baseline | YardDebris | composting | 9000 | 4 | endOfLife | GWP 100 | kg CO2 eq. | -70.38751 | -633487.57 |
| compostFW1000 | YardDebris | composting | 9000 | 77 | endOfLife | GWP 100 | kg CO2 eq. | -70.38751 | -633487.57 |
| compostFW585 | FoodWaste | composting | 585 | 77 | endOfLife | Energy demand | MJ | -543.11278 | -317720.98 |
| reduceFW03 | YardDebris | composting | 9000 | 4 | endOfLife | Energy demand | MJ | -543.11278 | -4888015.02 |
| compostFW1000 | FoodWaste | composting | 1000 | 77 | endOfLife | Energy demand | MJ | -543.11278 | -543112.78 |

This display shows you that composting benefits certainly are being registered by the WIC system. They aren’t showing up visibly on most of the impact charts because they are so small compared to the impacts of other processes (notably landfilling and production).

To see how composting is playing out with individual materials, we need to redraw the chart above so that the two materials have impacts displayed on different scales. We’ll do this by making two different charts, one for food waste and the other for yard debris.

Making the chart for food waste…

tempImpactChart7 <-  
 ggplot()+  
 ggtitle("Food waste impacts by LC stage (net in black)")+  
 theme\_539()+  
 geom\_bar(  
 data=tempImpactData5 %>%   
 filter(material=="FoodWaste") %>%  
 mutate(  
 scenario = factor(scenario, levels = rev(scenarioOrder)),  
 )  
 ,  
 aes(x=scenario, y=impact, fill=LCstage),  
 color=NA,  
 stat="identity",  
 alpha=0.5  
 )+  
 geom\_bar(  
 data=tempImpactData5b %>% filter(material=="FoodWaste"),  
 aes(x=scenario, y=impact),  
 stat="identity",  
 fill=NA,  
 color="black"  
 )+  
 scale\_y\_continuous(labels=comma)+  
 coord\_flip()+  
 facet\_wrap(  
 facets="impactLabel",  
 scales="free"  
 ,  
 labeller = labeller(impactLabel=label\_wrap\_gen(20))  
 )+  
 scale\_fill\_viridis(begin=0.32, end=1, discrete = TRUE,   
 direction = -1)+  
 theme(  
 axis.text.x=element\_text(size=8, angle=50, hjust=1, vjust=1),  
 panel.background=element\_blank(),  
 panel.grid = element\_blank(),  
 strip.text=element\_text(size=8, face="bold")  
 )  
tempImpactChart7



# saving the chart to a disk file  
ggsave(filename = "chart\_output/impacts7.png")

## Saving 10 x 7.5 in image

ggsave(filename = "chart\_output/impacts7.svg")

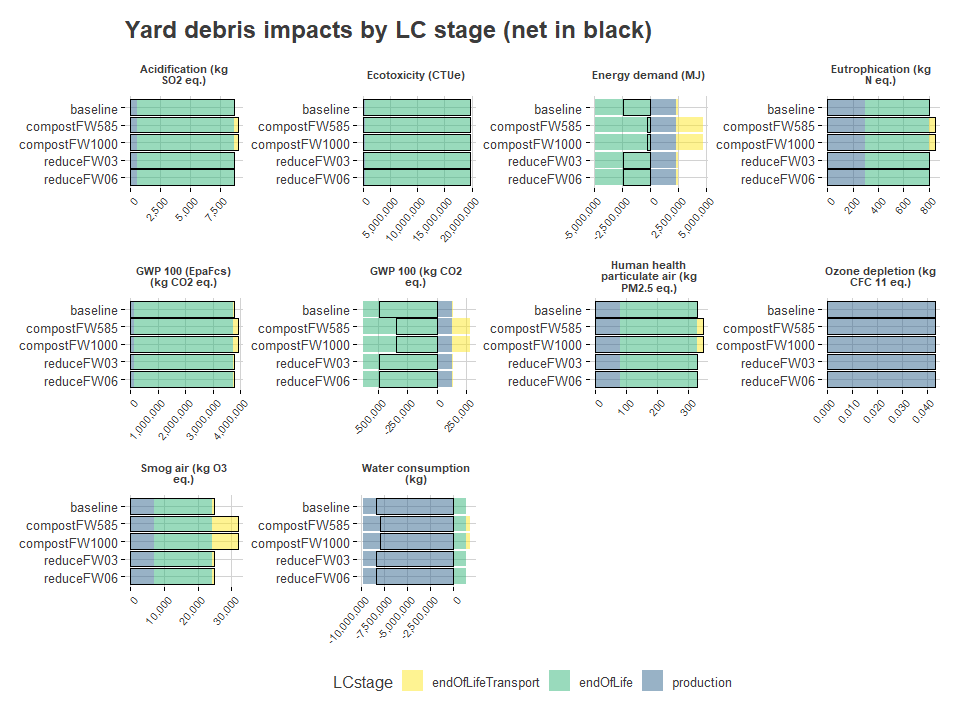
## Saving 10 x 7.5 in image

For food waste, we know that composting is contributing negative impacts in some impact categories (see earlier table), but it is not showing up in the “endOfLife” areas of the chart above because “endOfLife” also includes landfilling – and the impacts linked to landfilling are greater than the benefits linked to composting.

This chart also shows some small contributions to total impact from end-of-life-transportation.

Now, making the same chart for yard debris…

tempImpactChart6 <-  
 ggplot()+  
 ggtitle("Yard debris impacts by LC stage (net in black)")+  
 theme\_539()+  
 geom\_bar(  
 data=tempImpactData5 %>%   
 filter(material=="YardDebris") %>%  
 mutate(  
 scenario = factor(scenario, levels = rev(scenarioOrder)),  
 ),  
 aes(x=scenario, y=impact, fill=LCstage),  
 color=NA,  
 stat="identity",  
 alpha=0.5  
 )+  
 geom\_bar(  
 data=tempImpactData5b %>% filter(material=="YardDebris"),  
 aes(x=scenario, y=impact),  
 stat="identity",  
 fill=NA,  
 color="black"  
 )+  
 scale\_y\_continuous(labels=comma)+  
 coord\_flip()+  
 facet\_wrap(  
 facets="impactLabel",  
 scales="free"  
 ,  
 labeller = labeller(impactLabel=label\_wrap\_gen(20))  
 )+  
 scale\_fill\_viridis(begin=0.32, end=1, discrete = TRUE,   
 direction = -1)+  
 theme(  
 axis.text.x=element\_text(size=8, angle=50, hjust=1, vjust=1),  
 panel.background=element\_blank(),  
 panel.grid = element\_blank(),  
 strip.text=element\_text(size=8, face="bold")  
 )  
tempImpactChart6



# saving the chart to a disk file  
ggsave(filename = "chart\_output/impacts6.png")

## Saving 10 x 7.5 in image

ggsave(filename = "chart\_output/impacts6.svg")

## Saving 10 x 7.5 in image

When the analysis is limited to yard debris only, as in the chart above, some impact categories show clear benefits linked to composting – for example, see the “endOfLife” entries for energy demand and GWP100, which are negative numbers.

The role of transport impacts can also be clearly seen. Yard debris is generally a low-impact material in production and end-of-life treatment, giving end-of-life-transport impacts a proportionately large role. The extra transport associated with the compostFW585 and compostFW1000 scenarios is enough to increase the impacts of those scenarios above baseline – for example for energy demand, eutrophication and GWP 100. When yard debris is the only material in the system, transport is a significant factor.

However, yard debris is *not* the only part of the system. In Anytown’s situation, changes in the management of food waste affect management of yard debris, so they must be considered together. When they are considered together, food waste represents the vast majority of impacts.

If Anytown is interested in reducing the impacts of food waste and yard debris, there may be some promise in composting large amounts of food waste – larger than have been considered in any of the current scenarios. Of course, WIC allows you to create more scenarios and add them to the analysis, and you may follow that course if Anytown is very motivated to increase composting.

But in general, the analysis suggests to you that a reduction in food waste generation is probably the most effective strategy for reducing the impacts of this waste system.