check_effect_transformation

Sayani Gupta

01/09/2021

```r
library(gravitas)
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.2
```

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

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.0     v dplyr   1.0.5
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## Warning: package 'ggplot2' was built under R version 4.0.2
```

```
## Warning: package 'tibble' was built under R version 4.0.2
```

```
## Warning: package 'tidyr' was built under R version 4.0.2
```

```
## Warning: package 'readr' was built under R version 4.0.2
```

```
## Warning: package 'dplyr' was built under R version 4.0.2
```

```
## Warning: package 'forcats' was built under R version 4.0.2
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(gracsr)
library(tsibble)
```

```
## Warning: package 'tsibble' was built under R version 4.0.2
```

```
##
## Attaching package: 'tsibble'
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, union
```

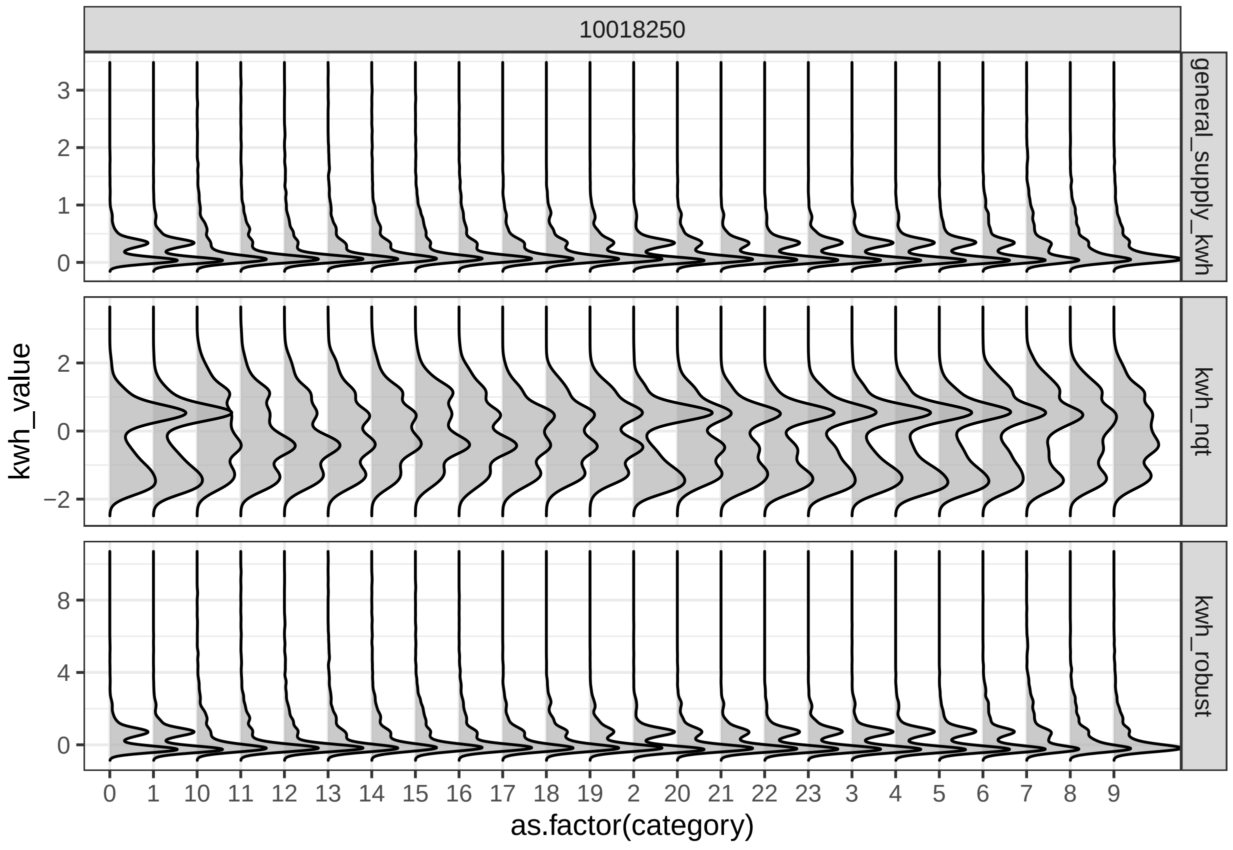
```r
# Feed in data and other inputs
sm <- smart_meter10 %>%
filter(customer_id %in% c("10006704", "10017936","10006414", "10018250"))
gran1 = "hour_day"
gran2 = NULL
response = "general_supply_kwh"


# Scale the data
```

```r
v2 <- suppressWarnings(robust_scale_data(sm, "hour_day")) %>%
    dplyr::rename( "kwh_robust" = "scaled_response") %>%
    dplyr::mutate(kwh_nqt = stats::qqnorm(general_supply_kwh, plot.it=FALSE)$x)


C
```

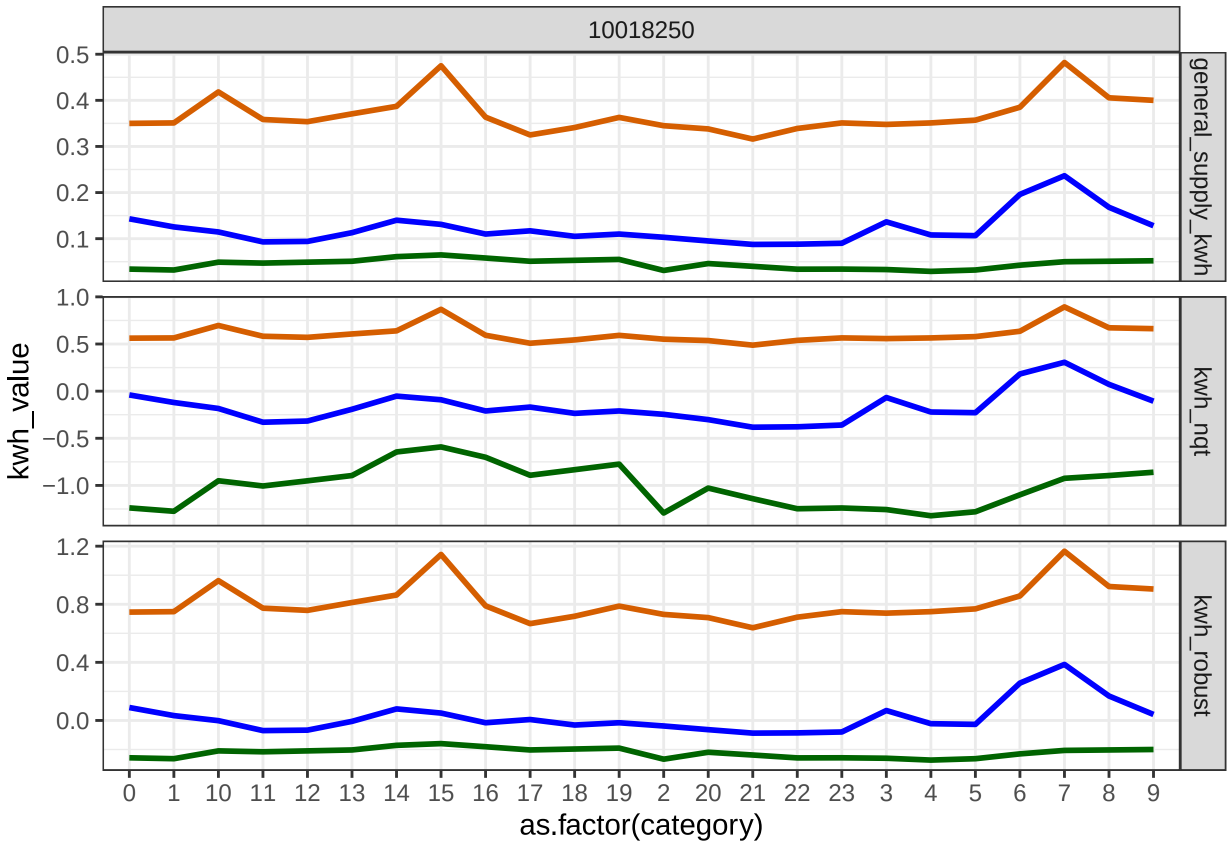
```
## function (object, contr, how.many, ...)
## {
##     if (isFALSE(as.logical(Sys.getenv("_R_OPTIONS_STRINGS_AS_FACTORS_"))))
##         object <- as.factor(object)
##     if (!nlevels(object))
##         stop("object not interpretable as a factor")
##     if (!missing(contr) && is.name(Xcontr <- substitute(contr)))
##         contr <- switch(as.character(Xcontr), poly = "contr.poly",
##             helmert = "contr.helmert", sum = "contr.sum", treatment = "contr.treatment",
##             SAS = "contr.SAS", contr)
##     if (missing(contr)) {
##         oc <- getOption("contrasts")
##         contr <- if (length(oc) < 2L)
##             if (is.ordered(object))
##                 contr.poly
##             else contr.treatment
##         else oc[1 + is.ordered(object)]
##     }
##     if (missing(how.many) && missing(...))
##         contrasts(object) <- contr
##     else {
##         if (is.character(contr))
##             contr <- get(contr, mode = "function")
##         if (is.function(contr))
##             contr <- contr(nlevels(object), ...)
##         contrasts(object, how.many) <- contr
##     }
##     object
## }
## <bytecode: 0x7fd9a8da3c70>
## <environment: namespace:stats>
```

```r
quantile_q2 <-  function(x){
  y = quantile(x, probs = c(0.5))
  #c(y[1], y[2]) %>% as_tibble() %>% bind_cols(names(y)) %>% set_names(c("quant_value", "quantile"))
}


quantile_q1 <-  function(x){
  y = quantile(x, probs = c(0.25))
  #c(y[1], y[2]) %>% as_tibble() %>% bind_cols(names(y)) %>% set_names(c("quant_value", "quantile"))
}


quantile_q3 <-  function(x){
  y = quantile(x, probs = c(0.75))
  #c(y[1], y[2]) %>% as_tibble() %>% bind_cols(names(y)) %>% set_names(c("quant_value", "quantile"))
}


v2 %>%
dplyr::filter(customer_id %in% c("10018250")) %>%
  pivot_longer(c("general_supply_kwh", "kwh_robust", "kwh_nqt"),
               names_to = "kwh_type",
               values_to = "kwh_value") %>% ggplot(fill = "#999999") +
  ggridges::geom_density_ridges(aes(x = kwh_value, y = as.factor(category)), alpha = 0.7) +
  facet_grid(kwh_type~customer_id, scales = "free") +
  coord_flip() +
  theme(legend.position = "bottom") +
  theme_bw()
```

```
## Picking joint bandwidth of 0.0522

## Picking joint bandwidth of 0.213

## Picking joint bandwidth of 0.166
```



```
v2 %>%
dplyr::filter(customer_id %in% c("10018250")) %>%
  pivot_longer(c("general_supply_kwh", "kwh_robust", "kwh_nqt"),
               names_to = "kwh_type",
               values_to = "kwh_value") %>%
ggplot(aes(x = kwh_value, y = as.factor(category)), fill = "#999999") +
  #ggridges::geom_density_ridges(alpha = 0.7) +
  facet_grid(kwh_type~customer_id, scales = "free") +
  coord_flip() +
  stat_summary(
    fun = quantile_q2,
    geom = 'line',
    aes(group = 1), size = 1, color = "blue") +
  theme(legend.position = "bottom") +
  stat_summary(
    fun = quantile_q1,
    geom = 'line',
    aes(group = 1), size = 1, color = "darkgreen") +
  theme(legend.position = "bottom") +
  stat_summary(
    fun = quantile_q3,
    geom = 'line',
    aes(group = 1), size = 1, color = "#D55E00") +
  theme(legend.position = "bottom") +
  theme_bw()
```



```
# library(tidyverse)
#
# # for loop
#
# for (x in 1:2){
#   for(y in 1:2){
#     for(z in 1:5){
#       dist_data[x, y] = x*y + y*z
#     }
#   }
#
# }
#
#
#
# tab <- expand.grid(x = 1:2, y = 1:2, z =1:5)
# tab
#
# # Using map
#
#
#
# # using pmap
#
# dist_data <- purrr::pmap(tab,
#                                    function(x, y, z){
#                                      value3 =
#                                        x*y + y*z
# }) %>%
#   unlist () %>%
#   as_tibble() %>%
#   bind_cols(tab)
```