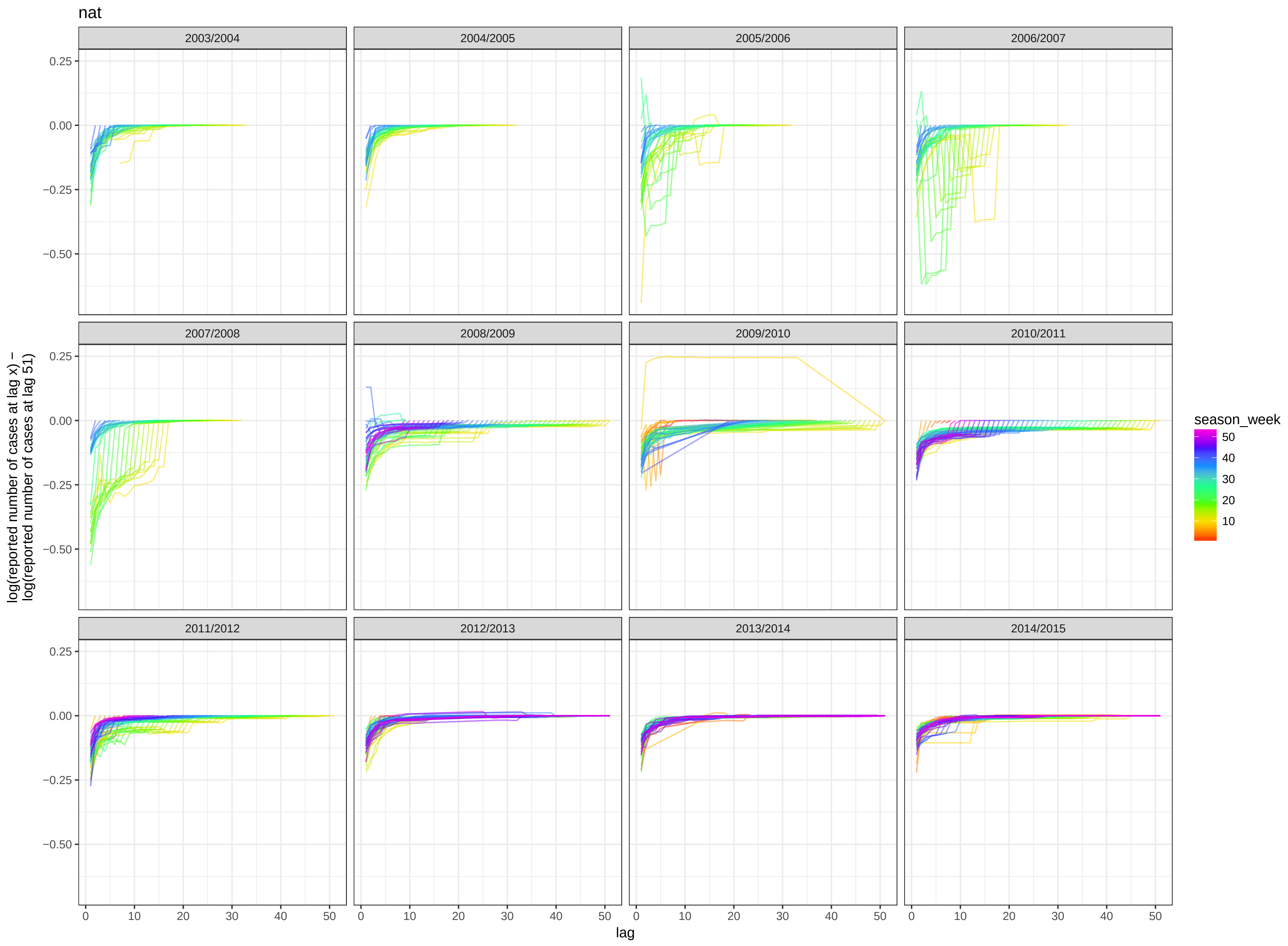
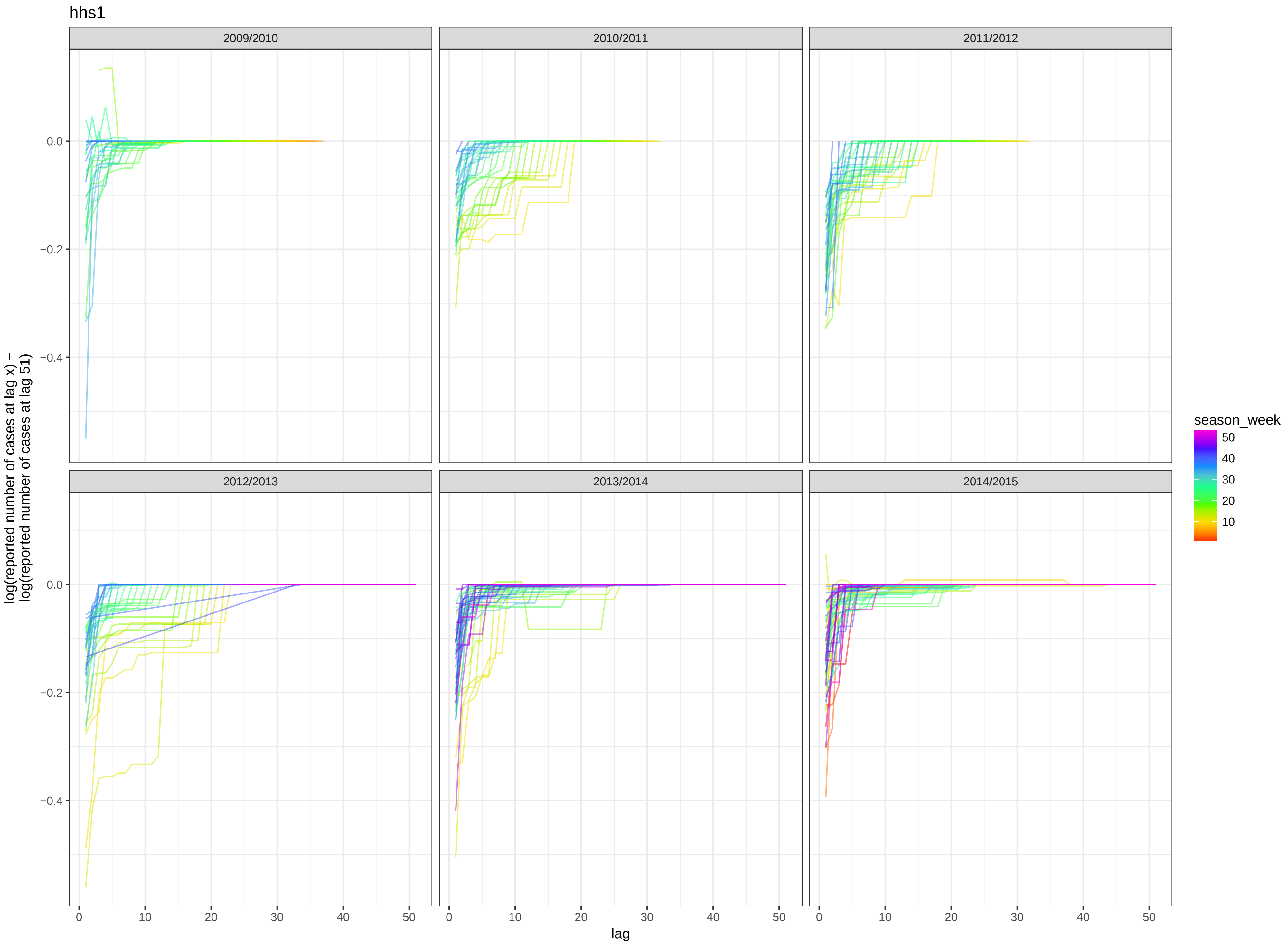
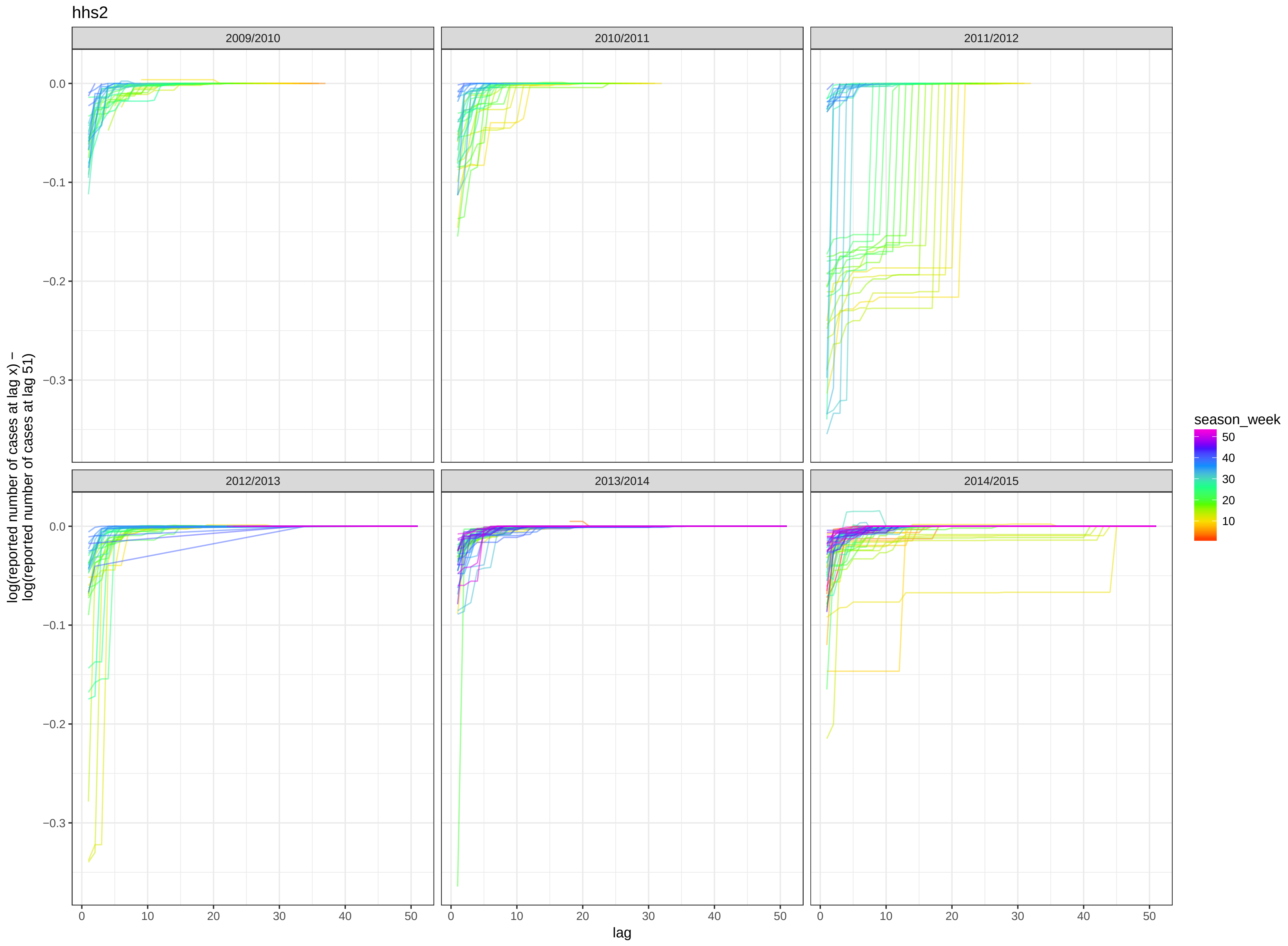
```r
library(plyr)
library(dplyr)

##
## Attaching package:  'dplyr'
## The following objects are masked from 'package:plyr':
##
##    arrange, count, desc, failwith, id, mutate, rename, summarise,
##    summarize
## The following objects are masked from 'package:stats':
##
##    filter, lag
## The following objects are masked from 'package:base':
##
##    intersect, setdiff, setequal, union

library(tidyr)
library(lubridate)

##
## Attaching package:  'lubridate'
## The following object is masked from 'package:plyr':
##
##    here
## The following object is masked from 'package:base':
##
##    date

library(ggplot2)
library(grid)

library(MMWRweek)
library(cdcfluview)

get_legend_grob <- function(x) {
  data <- ggplot2:::ggplot_build(x)

  plot <- data$plot
  panel <- data$panel
  data <- data$data
  theme <- ggplot2:::plot_theme(plot)
  position <- theme$legend.position
  if (length(position) == 2) {
    position <- "manual"
  }

  legend_box <- if (position != "none") {
    ggplot2:::build_guides(plot$scales, plot$layers, plot$mapping,
      position, theme, plot$guides, plot$labels)
  } else {
    ggplot2:::zeroGrob()
  }
  if (ggplot2:::is.zero(legend_box)) {
    position <- "none"
  }
  else {
    legend_width <- gtable:::gtable_width(legend_box) + theme$legend.margin
    legend_height <- gtable:::gtable_height(legend_box) + theme$legend.margin
    just <- valid.just(theme$legend.justification)
    xjust <- just[1]
    yjust <- just[2]
    if (position == "manual") {
      xpos <- theme$legend.position[1]
      ypos <- theme$legend.position[2]
      legend_box <- editGrob(legend_box, vp = viewport(x = xpos,
        y = ypos, just = c(xjust, yjust), height = legend_height,
        width = legend_width))
    }
    else {
      legend_box <- editGrob(legend_box, vp = viewport(x = xjust,
        y = yjust, just = c(xjust, yjust)))
    }
  }
  return(legend_box)
}
```

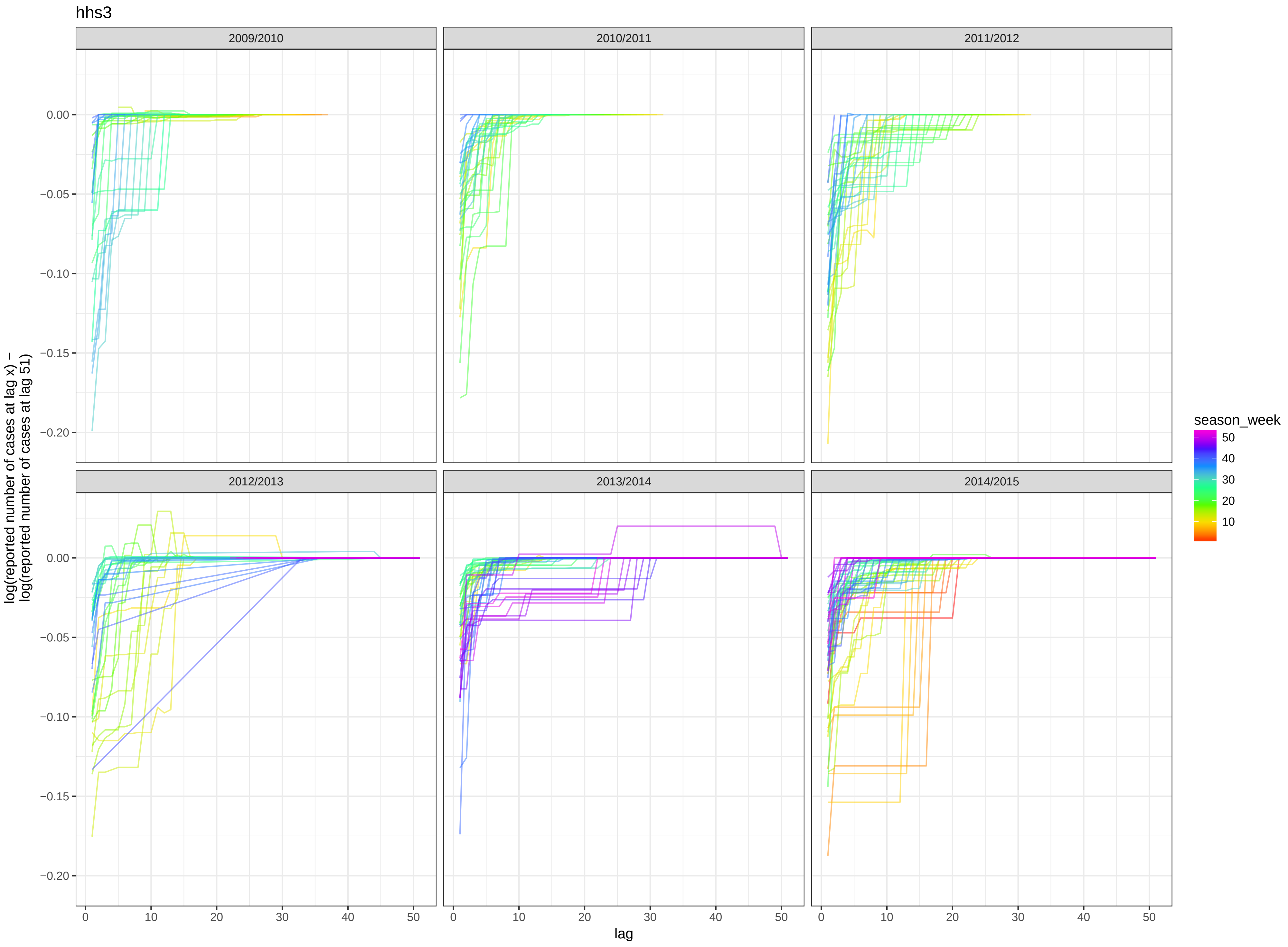
```r
flu <- readRDS("data/flu_data_with_backfill.rds")
flu <- flu %>%
  mutate(
    year = substr(epiweek, 1, 4) %>% as.numeric(),
    week = substr(epiweek, 5, 6) %>% as.numeric(),
    report_year = substr(issue, 1, 4) %>% as.numeric(),
    report_week = substr(issue, 5, 6) %>% as.numeric(),
    epi_week_date = MMWRweek::MMWRweek2Date(
      MMWRyear = year,
      MMWRweek = week
    ),
    report_week_date = MMWRweek::MMWRweek2Date(
      MMWRyear = report_year,
      MMWRweek = report_week
    )
  )

flu$season <- ifelse(
  flu$week <= 30,
  paste0(flu$year - 1, "/", flu$year),
  paste0(flu$year, "/", flu$year + 1)
)

## Season week column: week number within season
## weeks after week 30 get season_week = week - 30
## weeks before week 30 get season_week = week + (number of weeks in previous year) - 30
## This computation relies on the start_date function in package MMWRweek,
## which is not exported from that package's namespace!!!
flu$season_week <- ifelse(
  flu$week <= 30,
  flu$week + MMWRweek(MMWRweek:::start_date(flu$year) - 1)$MMWRweek - 30,
  flu$week - 30
)
```

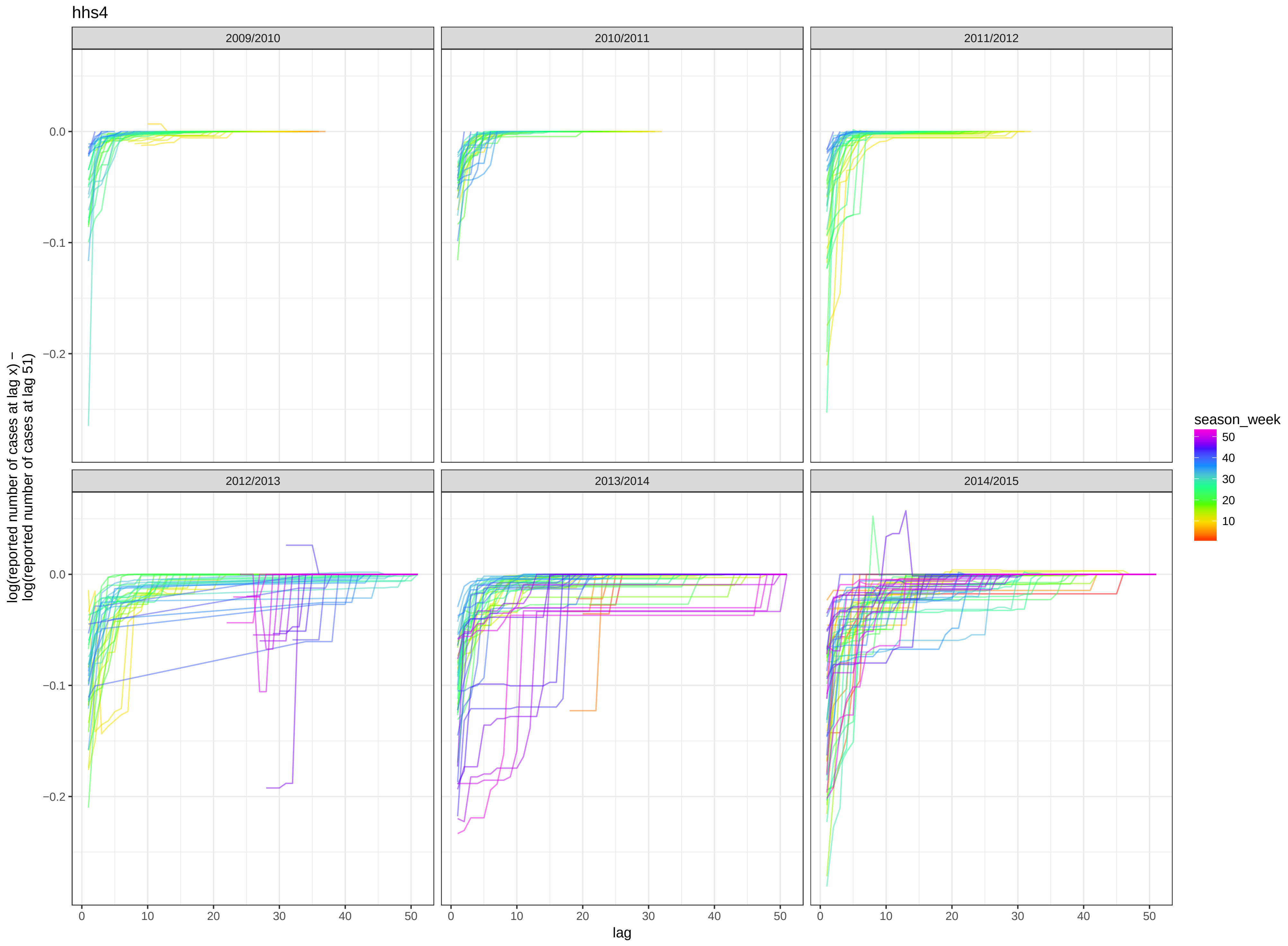
```r
flu <- flu %>%
  group_by_("region", "epi_week_date") %>%
  mutate(
    final_num_ili = num_ili[lag == max(lag)],
    diff_log_curr_final_num_ili = log(num_ili) - log(final_num_ili)
  ) %>%
  ungroup()

seasons_to_plot <- paste0(2003:2014, "/", 2004:2015)

for(region_val in unique(flu$region)) {
  p <- ggplot(data = flu %>% filter(region == region_val & season %in% seasons_to_plot)) +
    geom_line(
      aes(x = lag,
        y = diff_log_curr_final_num_ili,
        group = epi_week_date,
        color = season_week),
      alpha = 0.5) +
    scale_color_gradientn(colors = rainbow(7)) +
    facet_wrap(~ season) +
    ylab("log(reported number of cases at lag x) -\nlog(reported number of cases at lag 51)") +
    ggtitle(region_val) +
    theme_bw()
  print(p)
}
```


nat
2003/2004
2004/2005
2005/2006
2006/2007
2007/2008
2008/2009
2009/2010
2010/2011
2011/2012
2012/2013
2013/2014
2014/2015
0.25
0.00
−0.25
−0.50
0
10
20
30
40
50
lag
log(reported number of cases at lag x) −
log(reported number of cases at lag 51)
season_week
50
40
30
20
10

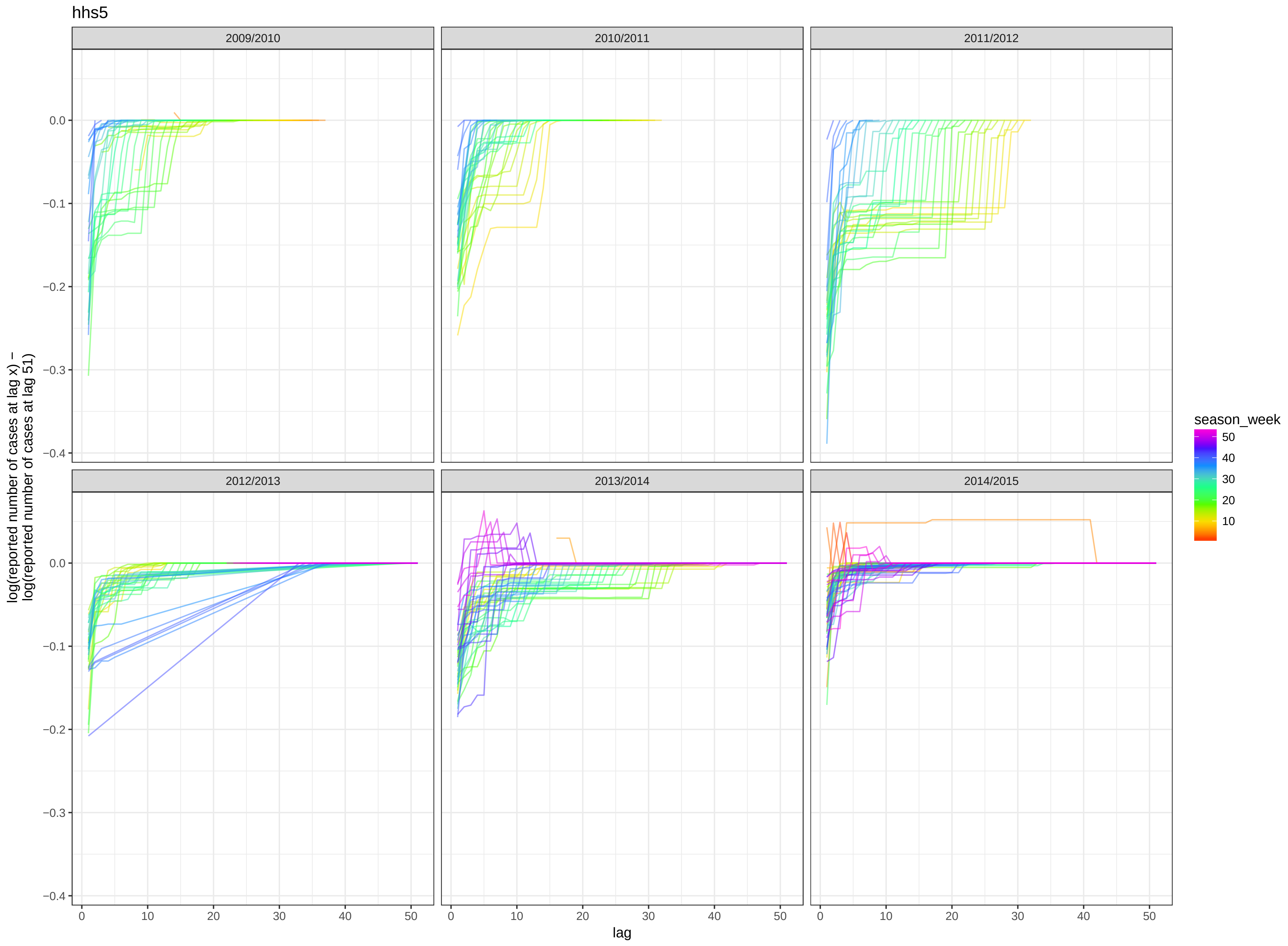

hhs1
2009/2010
2010/2011
2011/2012
2012/2013
2013/2014
2014/2015
0.0
−0.2
−0.4
log(reported number of cases at lag x) −
log(reported number of cases at lag 51)
0
10
20
30
40
50
lag
season_week
50
40
30
20
10

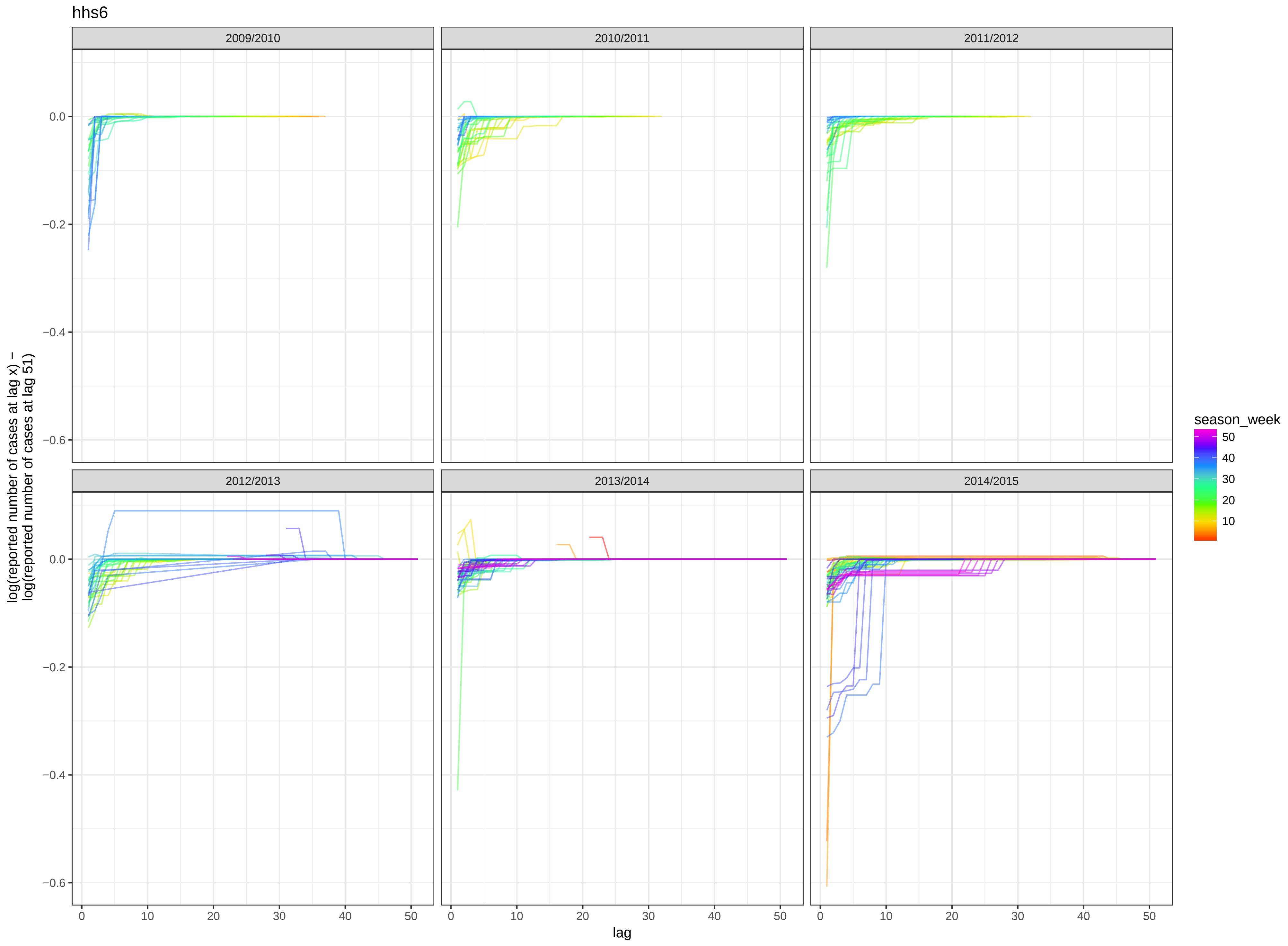

hhs2
2009/2010
2010/2011
2011/2012
2012/2013
2013/2014
2014/2015
log(reported number of cases at lag x) –
log(reported number of cases at lag 51)
0.0
−0.1
−0.2
−0.3
lag
0
10
20
30
40
50
season_week
50
40
30
20
10

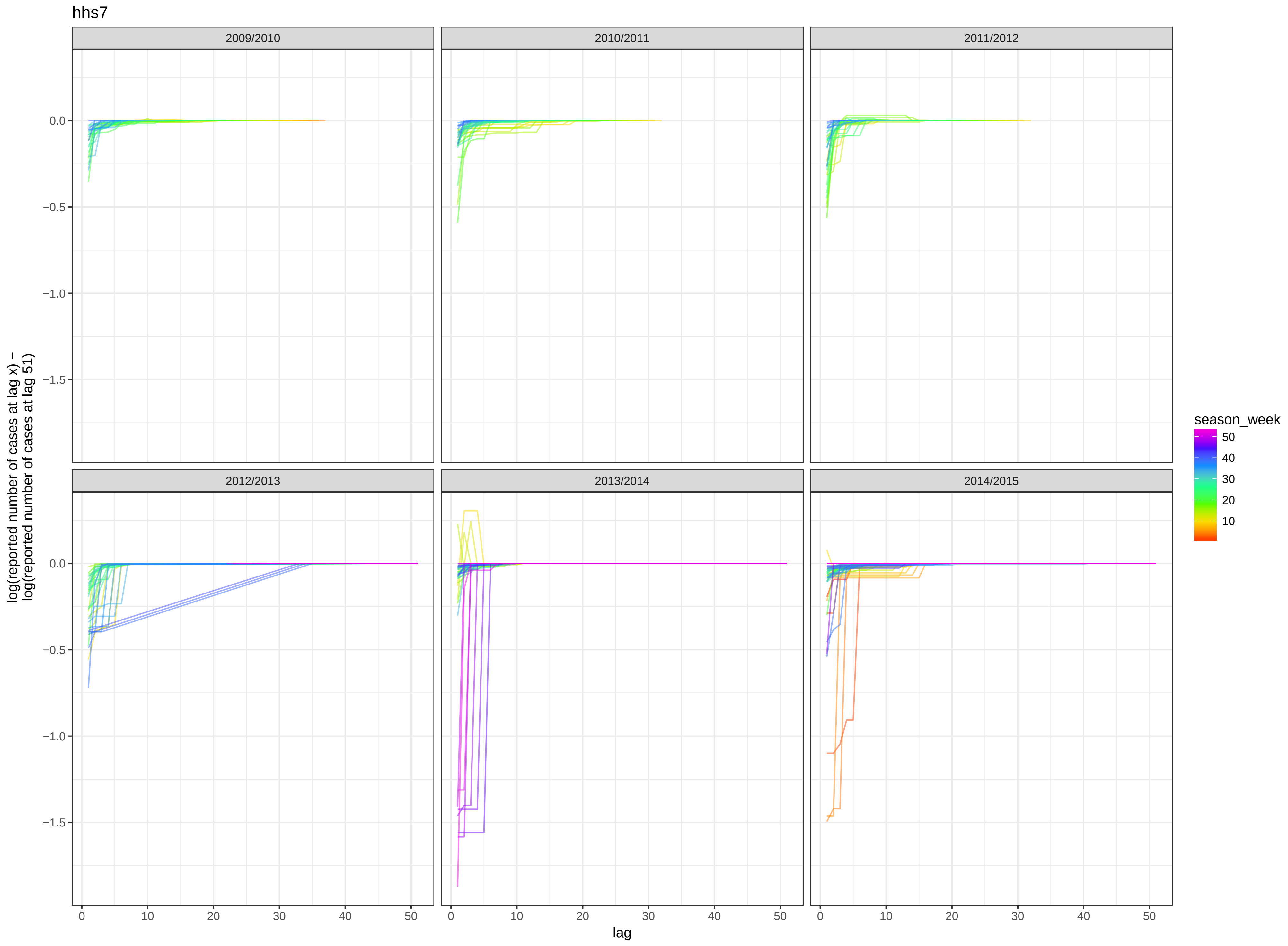

hhs3
2009/2010
2010/2011
2011/2012
2012/2013
2013/2014
2014/2015
log(reported number of cases at lag x) –
log(reported number of cases at lag 51)
0.00
−0.05
−0.10
−0.15
−0.20
lag
0
10
20
30
40
50
season_week
50
40
30
20
10

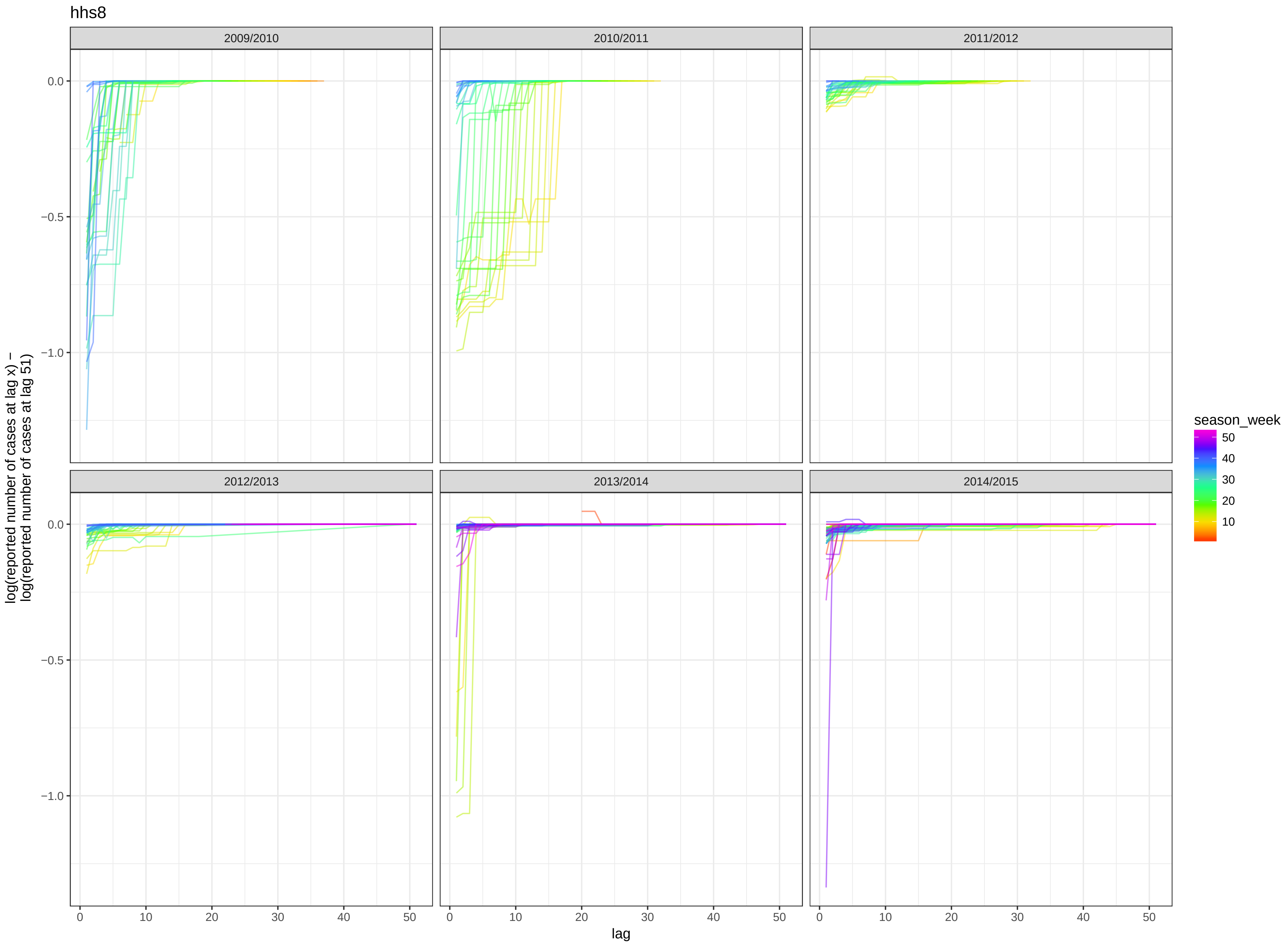


hhs4
2009/2010
2010/2011
2011/2012
2012/2013
2013/2014
2014/2015
log(reported number of cases at lag x) –
log(reported number of cases at lag 51)
0.0
−0.1
−0.2
0
10
20
30
40
50
lag
season_week
50
40
30
20
10

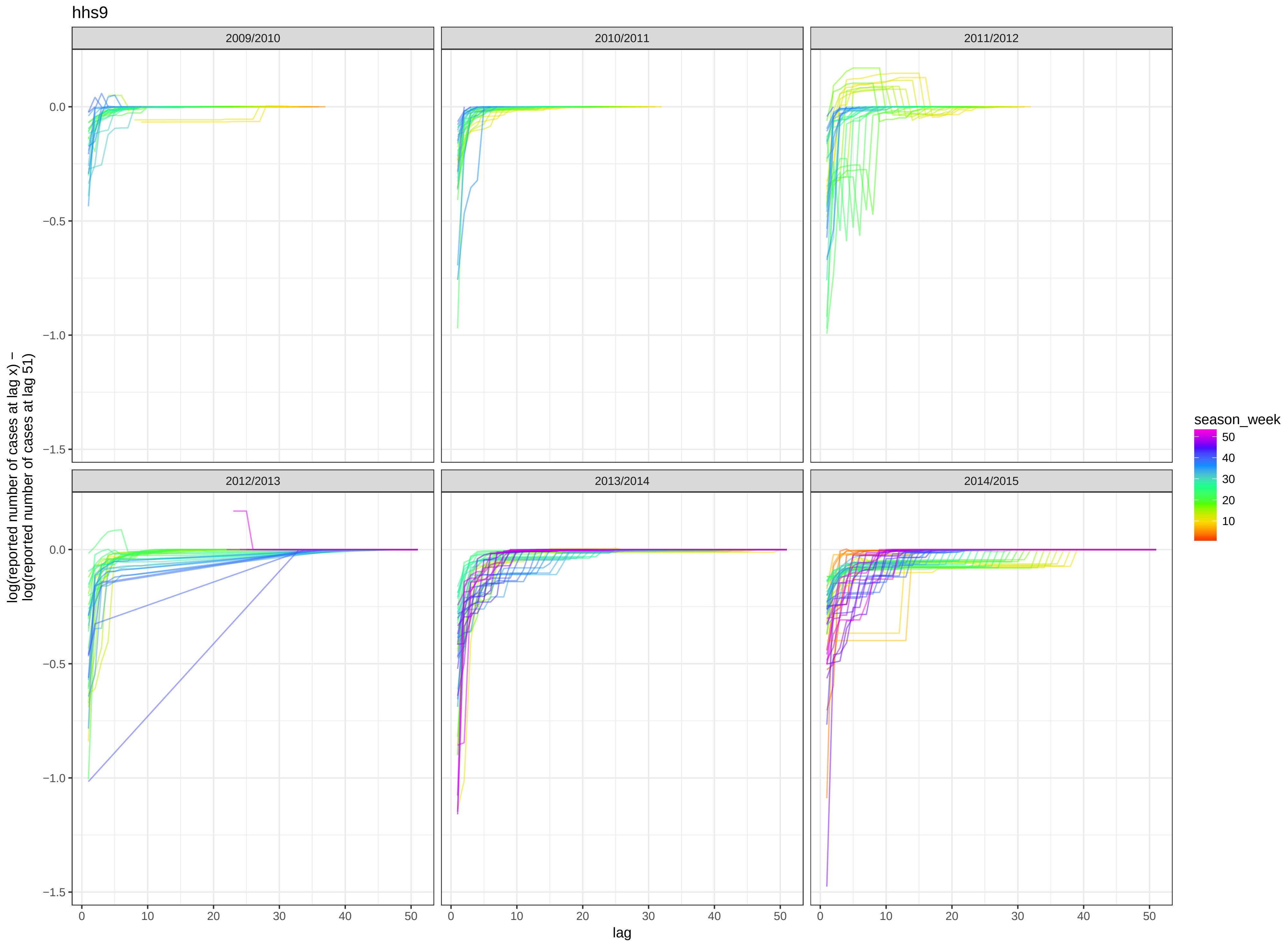

hhs5
2009/2010
2010/2011
2011/2012
2012/2013
2013/2014
2014/2015
0.0
−0.1
−0.2
−0.3
−0.4
0
10
20
30
40
50
lag
log(reported number of cases at lag x) −
log(reported number of cases at lag 51)
season_week
50
40
30
20
10

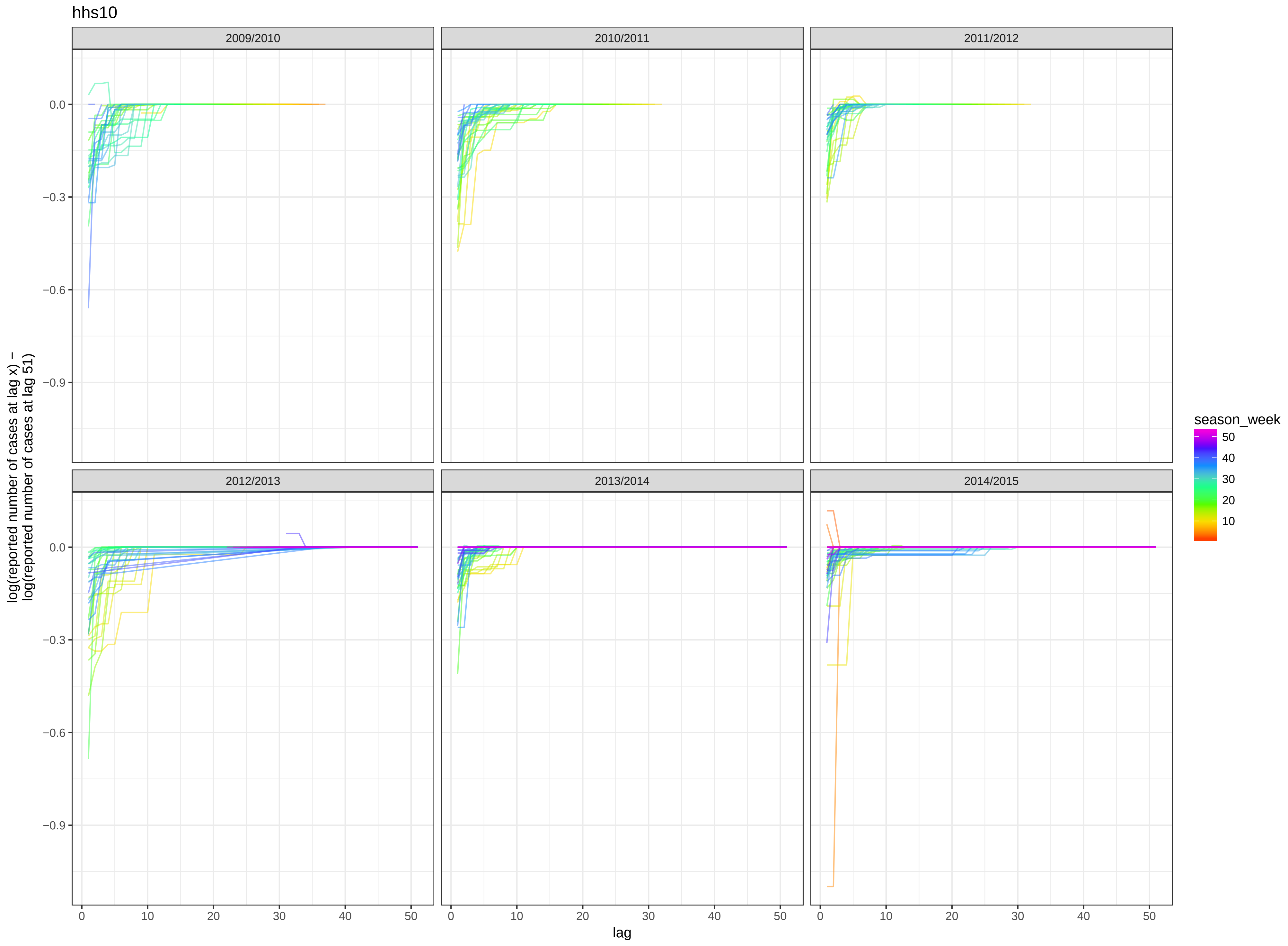

hhs6
2009/2010
2010/2011
2011/2012
2012/2013
2013/2014
2014/2015
log(reported number of cases at lag x) −
log(reported number of cases at lag 51)
0.0
−0.2
−0.4
−0.6
0
10
20
30
40
50
lag
season_week
50
40
30
20
10


hhs7
2009/2010
2010/2011
2011/2012
2012/2013
2013/2014
2014/2015
0.0
−0.5
−1.0
−1.5
log(reported number of cases at lag x) −
log(reported number of cases at lag 51)
0
10
20
30
40
50
lag
season_week
50
40
30
20
10



hhs8
2009/2010
2010/2011
2011/2012
2012/2013
2013/2014
2014/2015
0.0
−0.5
−1.0
log(reported number of cases at lag x) −
log(reported number of cases at lag 51)
0
10
20
30
40
50
lag
season_week
50
40
30
20
10


hhs9
2009/2010
2010/2011
2011/2012
2012/2013
2013/2014
2014/2015
log(reported number of cases at lag x) –
log(reported number of cases at lag 51)
lag
season_week
50
40
30
20
10
0.0
−0.5
−1.0
−1.5
0
10
20
30
40
50


hhs10
2009/2010
2010/2011
2011/2012
2012/2013
2013/2014
2014/2015
log(reported number of cases at lag x) –
log(reported number of cases at lag 51)
0.0
−0.3
−0.6
−0.9
0
10
20
30
40
50
lag
season_week
50
40
30
20
10