misc_equations

Elizabeth Zhang

2024-04-22

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become
```

```r
library(kableExtra)
```

```
##
## Attaching package: 'kableExtra'
##
## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```r
library(knitr)
```

kmr

Kernel machine regression defines the regression relationship using a flexible function $h : \mathbb{R}^M \to \mathbb{R}$, where

$$Y_i = h(\mathbf{x}_i) + \mathbf{z}_i^\top \boldsymbol{\beta}_{\mathbf{z}} + \varepsilon_i,$$

Here, Y_i is the outcome at a given point, $\mathbf{x}_i = [x_1, \ldots, x_M]^\top$ is a vector of M chemicals, $\mathbf{z}_i$ and $\boldsymbol{\beta}_{\mathbf{z}}$ are vectors of covariates and their weights, respectively, and $\varepsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$.

$h(\cdot)$ is obtained using the Gaussian kernel $k : \mathbb{R}^M \times \mathbb{R}^M \to \mathbb{R}$, defined as

$$k(\mathbf{x}, \mathbf{x}') = \exp\bigg\{ - \frac{\sum_{m=1}^{M} (x_m - x'_m)^2}{\rho} \bigg\},$$

where x are the predictor values of a second subject, and ρ is a tuning parameter.

bkmr priors

We define a weight, r_m, on each exposure by augmenting the kernel function as

$$k(\mathbf{x}, \mathbf{x}'|\mathbf{r}) = \exp\bigg\{ -\sum_{m=1}^{M} r_m (x_m - x'_m)^2 \bigg\},$$

where $r_m = 1/\rho_m$ is the inverse of the tuning parameter ρ_m for each $\mathbf{x}_m$.

To allow r_m to equal 0 with non-zero probability, we first define an indicator variable determining whether or not a predictor is included in the model, which is denoted and distributed as

$$\delta_m \sim \text{Bernoulli}(\pi),$$

where π is the prior probability of inclusion. Now, we can assume a "slab-and-spike" prior on r_m, distributed as

$$r_m|\delta_m \sim \delta_m f(r_m) + (1 - \delta_m)P_0,$$

where $f(\cdot)$ is some pdf with support $\mathbb{R}^+$, and P_0 denotes the density with point mass at 0.

We define the following prior on r_m:

$$\begin{gathered} \delta_m \sim \text{Bernoulli}(\pi), \text{ and} \\ r_m|\delta_m \sim \delta_m f(r_m) + (1 - \delta_m)P_0, \end{gathered}$$

where π is the prior probability of inclusion, $f(\cdot)$ is some pdf with support $\mathbb{R}^+$, and P_0 denotes the density with point mass at 0.

The posterior means of δ_m represent posterior inclusion probabilities (PIPs) of $\mathbf{x}_m$, which can be used as measures of the relative importance of each exposure.

Posterior means of $\delta_m \Rightarrow$ posterior inclusion probabilities (PIPs) of $\mathbf{x}_m$.

spline

BSR uses spline regression to define the regression relationship as

$$Y_i = f(\mathbf{x}_i) + \mathbf{z}_i^\top \boldsymbol{\beta}_{\mathbf{z}} + \varepsilon_i,$$

where f is defined by a set of basis functions on the exposures, $\mathbf{x}_i$, $\mathbf{z}_i$ and $\boldsymbol{\beta}_{\mathbf{z}}$ are the covariates and their associated weights, and ε_i is a random variable where $\boldsymbol{\varepsilon} \overset{\text{iid}}{\sim} N(0, \sigma^2)$.

BSR uses a natural spline regression. A general definition of the K basis functions for a natural spline with interior knots ξ_j, $j = 1, \ldots, K$ over $K+1$ disjoint intervals is given by:

$$b_1(X) = 1, \qquad b_2(X) = X, \qquad b_{k+2}(X) = d_k(X) - d_{K-1}(X),$$
$$d_k(X) = \frac{(X - \xi_k)_+^3 - (X - \xi_K)_+^3}{\xi_K - \xi_k}.$$

Here, the regression model is defined as $f(X) = \sum_{j=1}^{K} \beta_j b_j(X)$.

bsr priors

BSR assumes the following general model formulation:

$$f(\mathbf{x}_i) = \sum_{h=1}^{H} f^{(h)}(\mathbf{x}_i),$$
$$f^{(h)}(\mathbf{x}_i) = \sum_{m_1=1}^{M} \widetilde{x}_{im_1}\boldsymbol{\beta}_{m_1}^{(h)} + \sum_{m_1=2}^{M}\sum_{m_2<m_1} \widetilde{x}_{im_1m_2}\boldsymbol{\beta}_{m_1m_2}^{(h)} + \dots,$$

where $\widetilde{X}_m = [b_{m1}(X_m), \dots, b_{md}(X_m)]$ represents a d-dimensional basis function expansion for the mth term, and $f^{(h)}(\mathbf{x}_i)$ includes a summation of all M-way interactions.

We define the following prior on $\boldsymbol{\beta}_S^{(h)}$:

$$P(\zeta_{mh} = 1) = \tau_h^{\zeta_{mh}}(1-\tau_h)^{1-\zeta_{mh}} I(A_h \not\subset A_{h'} \forall h' \neq h \text{ or } A_h = \{\}),$$
$$\text{where } A_h = \{m : \zeta_h = 1\}, \text{ and}$$

$$P(\boldsymbol{\beta}_S^{(h)}|\boldsymbol{\zeta}) = \left(1 - \prod_{m\in S}\zeta_{mh}\right)P_{\mathbf{0}} + \left(\prod_{m\in S}\zeta_{mh}\right)\psi_1(\boldsymbol{\beta}_S^{(h)}),$$
$$\text{where } S \text{ is some subset of } 1, 2, \dots, m,$$

where ζ_{mh} have prior probability of inclusion τ_h, $I()$ is an indicator to help with identifiability issues, $P_{\mathbf{0}}$ denotes the density with point mass at $\mathbf{0}$, and $\psi_1()$ is a multivariate normal distribution with mean $\mathbf{0}$ and covariance $\boldsymbol{\Sigma}_{\boldsymbol{\beta}}$, a diagonal matrix with $\sigma^2\sigma^2_{\boldsymbol{\beta}}$ on the diagonals.

equations for interactions

$$Y = x_1 + x_2$$

$$Y = x_1 + x_2 + 0.5(x_1 * x_2)$$

$$Y = x_1 + x_2 + 0.2(x_1 * (x_2 - 1)^2)$$

$$\widehat{Y} = \widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \widehat{\beta}_2 x_2 + \widehat{\beta}_{12} x_1 x_2$$

Table 1: Specification of interaction terms in simulations.

	Effect size	
	Lower	Higher
Chemical by chemical		
Multiplicative	$0.35\text{Hg}*\text{Ni}$	$0.7\text{Hg}*\text{Ni}$
Polynomial	$0.13\text{Hg}*(\text{Ni}-1)^2$	$0.26\text{Hg}*(\text{Ni}-1)^2$
Chemical by race		
Smaller (n=27/252)	$0.5\text{Hg}*\text{race}_{\text{black}}$	$\text{Hg}*\text{race}_{\text{black}}$
Larger (n=109/252)	$0.5\text{Hg}*\text{race}_{\text{hisp.non}}$	$\text{Hg}*\text{race}_{\text{hisp.non}}$

specification of interaction terms

```r
equations <- data.frame(
  type = c("Multiplicative", "Polynomial",
           "Smaller (n=27/252)", "Larger (n=109/252)"),
  small = c("0.35Hg$*$Ni", "0.13Hg$*($Ni$-1)^2$",
            "0.5Hg$*\\text{race}_{\\text{black}}$",
            "0.5Hg$*\\text{race}_{\\text{hisp.non}}$"),
  large = c("0.7Hg$*$Ni", "0.26Hg$*($Ni$-1)^2$",
            "Hg$*\\text{race}_{\\text{black}}$",
            "Hg$*\\text{race}_{\\text{hisp.non}}$")
)
labels <- c(
  "Chemical by chemical" = 2,
  "Chemical by race" = 2
)

equations |>
  kbl(booktabs = TRUE, escape = FALSE,
      col.names = c("", "Lower", "Higher"),
      align = "lcc",
      caption = "Specification of interaction terms in simulations.") |>
  column_spec(1, width = "10em") |>
  add_header_above(header = c(" " = 1, "Effect size" = 2)) |>
  pack_rows(index = labels)
```

race by ethnicity sensitivity

```r
# table
bkmr_re_sens <- read_csv("/Users/elizabethzhang/thesis/thesis/index/data/bkmr_re_sens.csv") |>
```

```
  rename(sensitivity = sens)
```

```
## Rows: 12 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): size
## dbl (2): case, sens
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
oracle_re_int <- read_csv("/Users/elizabethzhang/thesis/thesis/index/data/oracle_re_sens.csv") |>
  filter(variable == "Int") |>
  group_by(size, case) |>
  summarize(sensitivity = sum(p<0.05)/n()) |>
  mutate(size = ifelse(size == "Small", "Small uncollapsed", size))
```

```
## Rows: 4000 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): size, variable
## dbl (3): case, trial, p
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
## ‘summarise()‘ has grouped output by ’size’. You can override using the ‘.groups‘ argument.
```

```
re_ints <- bind_rows(
  mutate(bkmr_re_sens, mod = "BKMR"),
  mutate(oracle_re_int, mod = "Oracle MLR")
) |>
  arrange(desc(size), desc(mod)) |>
  pivot_wider(names_from = c(mod, size), values_from = sensitivity) |>
  mutate(effect_size = ifelse(case %in% c(1, 3), "Lower", "Higher"),
         case = ifelse(case %in% c(1, 2),
                       paste0("Original n=27", footnote_marker_symbol(1)),
                       paste0("Original n=109", footnote_marker_symbol(2)))
         ) |>
  relocate(case, effect_size) |>
  select(-5)

re_ints |>
  kbl(booktabs = TRUE, escape = FALSE,
      align = "llcccc",
      caption = "Sensitivity to interactions between the categorical race variable and Hg.",
      col.names = c("Interaction in", "Effect size",
```

Table 2: Sensitivity to interactions between the categorical race variable and Hg.

		Small (n=252)		**Large (n=1000)**	
Interaction in	Effect size	Oracle	BKMR	Oracle	BKMR
Original n=27[*]	Lower	0.07	0.00	0.21	0.01
	Higher	0.19	0.00	0.51	0.03
Original n=109[†]	Lower	0.12	0.00	0.39	0.03
	Higher	0.24	0.02	0.83	0.21

[*] Non-Hispanic black
[†] Hispanic born outside US

```
                    "Oracle", "BKMR",
                    "Oracle", "BKMR")
      ) |>
  add_header_above(header = c(" " = 2, "Small (n=252)" = 2, "Large (n=1000)" = 2),
                   bold = TRUE) |>
  collapse_rows(columns = 1, valign = "middle", latex_hline = "linespace") |>
  # column_spec(7, width = "6em") |>
  add_footnote(c("Non-Hispanic black", "Hispanic born outside US"), notation = "symbol", threepar
```

emissions factors for ng by cogen in 2023

```
# read in data
df <- read_csv("cleaned_02-24.csv")
```

```
## Rows: 108 Columns: 18
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (18): month, year, NG Small (therms), NG Cogen (therms), NG Boilers (the...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# get natural gas monthly amounts
ng <- df |>
  select(month, year, cogen = 4, boilers = 5) |>
  filter(year != 2024) |>
  rowwise() |>
  mutate(total_cf = (cogen + boilers)*100,
         date = as.Date(paste(year, str_pad(month, 2, pad = "0"), "01", sep = "-")))
```

```
comps <- readxl::read_xlsx("naturalgas_emissions.xlsx", sheet = 1)
metals <- readxl::read_xlsx("naturalgas_emissions.xlsx", sheet = 2)

all_poll <- bind_rows(
  mutate(comps, type = "Compounds"),
  mutate(metals[,2:4], type = "Metals")
)

ng2023 <- ng |>
  group_by(year) |>
  summarize(total_cf = sum(total_cf)) |>
  filter(year == 2023) |>
  select(-year) |>
  as.numeric()

all_poll2023 <- all_poll |>
  janitor::clean_names() |>
  mutate(emission_factor_lb_106_scf =
           as.numeric(str_replace_all(emission_factor_lb_106_scf,
                                      "[,<]", "")),
         emissions_lb = emission_factor_lb_106_scf * ng2023 / 106)

metals2023 <- all_poll2023 |>
  filter(type == "Metals")

metals2023 |>
  select(1, 2, 5, 3) |>
  mutate(emissions_lb = format(round(emissions_lb, 2), big.mark = ",")) |>
  kbl(booktabs = TRUE, align = "lccc", escape = FALSE,
      col.names = c("Metal",
                    linebreak("Factor\n(lb/106scf)", align = "c"),
                    linebreak("Emissions\n(lb)", align = "c"), "Grade"))
```

Metal	Factor (lb/106scf)	Emissions (lb)	Grade
Arsenic	2.0e-04	376.02	E
Barium	4.4e-03	8,272.39	D
Beryllium	1.2e-05	22.56	E
Cadmium	1.1e-03	2,068.10	D
Chromium	1.4e-03	2,632.13	D
Cobalt	8.4e-05	157.93	D
Copper	8.5e-04	1,598.08	C
Lead	5.0e-04	940.04	D
Manganese	3.8e-04	714.43	D
Mercury	2.6e-04	488.82	D
Molybdenum	1.1e-03	2,068.10	D
Nickel	2.1e-03	3,948.19	C
Selenium	2.4e-05	45.12	E
Vanadium	2.3e-03	4,324.21	D
Zinc	2.9e-02	54,522.60	E