Diet

2025-11-05

Table of contents

Overview

This report presents a sensitivity analysis examining the association between dietary factors and disease flare risk in inflammatory bowel disease (IBD) patients. Unlike the primary analysis which categorizes fecal calprotectin (FC) into quantiles, this analysis treats FC as a continuous variable in Cox proportional hazards models.

The key methodological differences from the primary analysis are:

1. **Continuous FC**: FC is included as a continuous predictor rather than categorical
2. **Spline modeling**: Natural splines are tested to assess non-linear relationships
3. **Hazard ratio curves**: Continuous hazard ratios are plotted across the range of each dietary variable
4. **Risk difference estimation**: Population-average risk differences are calculated using bootstrapping

Dietary variables examined include total meat protein, dietary fibre, polyunsaturated fatty acids (PUFA), and ultra-processed food (UPF) intake. Analyses are stratified by disease type (Crohn's disease vs. ulcerative colitis) and flare definition (patient-reported vs. objective).

Setting up in R

Load the data

```r
library(splines)
library(patchwork)

source("Survival/utils.R")

# Setup analysis environment
analysis_setup <- setup_analysis()
paths <- analysis_setup$paths
demo <- analysis_setup$demo

flare.df <- readRDS(paste0(paths$outdir, "flares-biochem.RDS"))
flare.cd.df <- readRDS(paste0(paths$outdir, "flares-biochem-cd.RDS"))
flare.uc.df <- readRDS(paste0(paths$outdir, "flares-biochem-uc.RDS"))
```

Data cleaning

```r
# FC has been logged twice - reverse
flare.df %<>%
  dplyr::mutate(FC = exp(FC))

flare.cd.df %<>%
  dplyr::mutate(FC = exp(FC))

flare.uc.df %<>%
  dplyr::mutate(FC = exp(FC))
# So now FC is on the original measurement scale (log transformation reversed)

# Transform age variable to decades
flare.df %<>%
  dplyr::mutate(
```

```
    age_decade = Age / 10
  )

flare.cd.df %<>%
  dplyr::mutate(
    age_decade = Age / 10
  )

flare.uc.df %<>%
  dplyr::mutate(
    age_decade = Age / 10
  )
```

Helper functions

```
plot_continuous_hr <- function(data, model, variable, splineterm = NULL){

  # Plotting continuous Hazard Ratio curves
  # Data is the same data used to fit the Cox model
  # model is a Cox model
  # variable is the variable we are plotting
  # splineterm is the spline used in the model, if there is one

  # Range of the variable to plot
  variable_range <- data %>%
    dplyr::pull(variable) %>%
    quantile(., probs = seq(0, 0.9, 0.01), na.rm = TRUE) %>%
    unname() %>%
    unique()

  # If there is a spline
  if (!is.null(splineterm)){

    # Recreate the spline
    spline <- with(data, eval(parse(text = splineterm)))

    # Get the spline terms for the variable values we are plotting
    spline_values <- predict(spline, newx = variable_range)

    # Get the reference values of the variables that were used to fit the Cox model
```

```r
    X_ref <- model$means

    # New values to calculate the HR at
    # Take the ref values and vary our variable of interest

    # Number of values
    n_values <- variable_range %>% length()

    X_ref <- matrix(rep(X_ref, n_values), nrow = n_values, byrow = TRUE)
    # Fix the column names
    colnames(X_ref) <- names(model$means)

    # Now add our new variable values
    X_new <- X_ref
    # Identify the correct columns
    column_pos <- stringr::str_detect(colnames(X_new), variable)
    # Add the values - ordering of columns and model term names should be preserved
    X_new[,column_pos] <- spline_values

    # Calculate the linear predictor and se
    # Method directly from predict.coxph from the survival package
    lp <- (X_new - X_ref)%*%coef(model)

    lp_se <- rowSums((X_new - X_ref)%*%vcov(model)*(X_new - X_ref)) %>%
      as.numeric() %>%
      sqrt()

} else {
  # If there is no spline then it is easier

  # Get the reference values of the variables that were used to fit the Cox model
  X_ref <- model$means

  # New values to calculate the HR at
  # Take the ref values and vary our variable of interest

  # Number of values
  n_values <- variable_range %>% length()

  X_ref <- matrix(rep(X_ref, n_values), nrow = n_values, byrow = TRUE)
  # Fix the column names
  colnames(X_ref) <- names(model$means)
```

```r
    # Now add our new variable values
    X_new <- X_ref

    # Add the values
    X_new[,variable] <- variable_range

    # Calculate the linear predictor and se
    # Method directly from predict.coxph from the survival package
    lp <- (X_new - X_ref)%*%coef(model) %>% as.numeric()

    lp_se <- rowSums((X_new - X_ref)%*%vcov(model)*(X_new - X_ref)) %>%
      as.numeric() %>%
      sqrt()

  }

  # Store results in a tibble
  data_pred <- tibble(
    !!sym(variable) := variable_range,
    p = lp,
    se = lp_se
  )

  # Plot
  data_pred %>%
    ggplot(aes(
      x = !!rlang::sym(variable),
      y = exp(p),
      ymin = exp(p - 1.96 * se),
      ymax = exp(p + 1.96 * se)
    )) +
    geom_point() +
    geom_line() +
    geom_ribbon(alpha = 0.2) +
    ylab("HR") +
    theme_minimal()
}

# Function to perform an LRT
summon_lrt <- function(model, remove = NULL, add = NULL) {
  # Function that removes a term and performs a
  # Likelihood ratio test to determine its significance
```

```r
  new_formula <- "~ ."

  if (!is.null(remove)) {
    new_formula <- paste0(new_formula, " - ", remove)
  } else {
    remove <- NA
  }

  if (!is.null(add)) {
    new_formula <- paste0(new_formula, " + ", add)
  } else {
    add <- NA
  }

  update(model, new_formula) %>%
    anova(model, ., test = "LRT") %>%
    broom::tidy() %>%
    dplyr::filter(!is.na(p.value)) %>%
    dplyr::select(-term) %>%
    dplyr::mutate(
      removed = remove,
      added = add
    ) %>%
    dplyr::mutate(test = "LRT") %>%
    dplyr::select(test, removed, added, tidyselect::everything()) %>%
    knitr::kable(col.names = c("Test",
                               "Remolved",
                               "Added",
                               "Log-likelihood",
                               "Statistic",
                               "DF",
                               "p-vaue"),
                 digits = 3)
}


summon_population_risk_difference <- function(data,
                                              model,
                                              times,
                                              variable,
                                              values,
                                              ref_value = NULL) {
```

```r
# Function that calculates (population average) risk difference of a variable
# at given time points with respect to specified value of the variable from a
# Cox model

# If no reference value then use one in the middle
if (is.null(ref_value)) {
  pos <- (length(values) / 2) %>% ceiling()
  ref_value <- values %>% purrr::pluck(pos)
}

# Identify the time variable - to differentiate between soft and hard flare models
time_variable <- all.vars(terms(model))[1]

df <- tidyr::expand_grid(times, values)

purrr::map2(
  .x = df$times,
  .y = df$values,
  .f = function(x, y) {
    data %>%
      # Set values for time and the variable
      dplyr::mutate(!!sym(time_variable) := x, !!sym(variable) := y) %>%
      # Estimate expected for entire population
      predict(model, newdata = ., type = "expected") %>%
      tibble(expected = .) %>%
      # Remove NA
      dplyr::filter(!is.na(expected)) %>%
      # Calculate survival and cumulative incidence (1 - survival)
      dplyr::mutate(
        survival = exp(-expected),
        cum_incidence = 1 - survival
      ) %>%
      dplyr::select(cum_incidence) %>%
      dplyr::summarise(cum_incidence = mean(cum_incidence)) %>%
      # Note time, value and variable
      dplyr::mutate(time = x, value = y, variable = variable)
  }
) %>%
  purrr::list_rbind() %>%
  # Calculate risk difference relative to a reference
  {
    df <- .
```

```r
    # Subtract chosen reference value to set linear predictor to 0 at that value
    cum_incidence_ref <- df %>%
      dplyr::filter(value == ref_value) %>%
      dplyr::rename(cum_incidence_ref = cum_incidence) %>%
      dplyr::select(time, cum_incidence_ref)

    df %>%
      dplyr::left_join(cum_incidence_ref, by = "time") %>%
      dplyr::mutate(rd = (cum_incidence - cum_incidence_ref)*100, .keep = "unused")
  }
}


summon_population_risk_difference_boot <- function(data,
                                                   model,
                                                   times,
                                                   variable,
                                                   values,
                                                   ref_value = NULL,
                                                   nboot = 99,
                                                   seed = 1) {
  # Function to calculate population risk difference for a given variable at given time point
  # relative to a specified reference value
  # Confidence intervals calculated using bootstrapping.

  # Set seed
  set.seed(seed)

  # If no reference value then use one in the middle
  if (is.null(ref_value)) {
    pos <- (length(values) / 2) %>% ceiling()
    ref_value <- values %>% purrr::pluck(pos)
  }

  purrr::map_dfr(
    .x = seq_len(nboot),
    .f = function(b) {
      # Variable used in the model
      all_variables <- all.vars(terms(model))

      # Bootstrap sample of the data
      data_boot <- data %>%
```

```r
      # Remove any NAs as Cox doesn't use these
      dplyr::select(tidyselect::all_of(all_variables)) %>%
      dplyr::filter(!dplyr::if_any(.cols = everything(), .fns = is.na)) %>%
      # Sample the df
      dplyr::slice_sample(prop = 1, replace = TRUE)

    # Refit cox model of bootstrapped data
    model_boot <- coxph(formula(model), data = data_boot, model = TRUE)

    # Calculate risk differences
    summon_population_risk_difference(
      data = data_boot,
      model = model_boot,
      times = times,
      variable = variable,
      values = values,
      ref_value = ref_value
    )
  }
) %>%
  dplyr::group_by(time, value) %>%
  # Bootstrapped estimate and confidence intervals
  dplyr::summarise(
    mean_rd = mean(rd),
    conf.low = quantile(rd, prob = 0.025),
    conf.high = quantile(rd, prob = 0.975)
  ) %>%
  dplyr::ungroup() %>%
  dplyr::rename(estimate = mean_rd) %>%
  # Note variable, reference level flag
  dplyr::mutate(
    variable = variable,
    reference_flag = (value == ref_value)
  ) %>%
  # Ordering for plotting
  dplyr::arrange(time, value) %>%
  dplyr::group_by(time) %>%
  dplyr::mutate(ordering = dplyr::row_number()) %>%
  # Tidy confidence intervals
  # Remove confidence intervals for reference level
  dplyr::mutate(
    conf.low = dplyr::case_when(reference_flag == TRUE ~ NA,
```

```r
          .default = conf.low
      ),
      conf.high = dplyr::case_when(reference_flag == TRUE ~ NA,
        .default = conf.high
      )
    ) %>%
    dplyr::mutate(
      conf.interval.tidy = dplyr::case_when(
        (is.na(conf.low) & is.na(conf.high)) ~ "-",
        TRUE ~ paste0(
          sprintf("%.3g", estimate),
          " (",
          sprintf("%.3g", conf.low),
          ", ",
          sprintf("%.3g", conf.high),
          ")"
        )
      )
    ) %>%
    # Significance
    dplyr::mutate(
      significance = dplyr::case_when(
        reference_flag == TRUE ~ "Reference level",
        sign(conf.low) == sign(conf.high) ~ "Significant",
        sign(conf.low) != sign(conf.high) ~ "Not Significant"
      )
    )
}


summon_rd_forest_plot <- function(data, time) {
  # Creating a forest plot of risk difference from the result of
  # summon_population_risk_difference_boot

  # Mask time
  time_point <- time

  data_plot <- data %>%
    dplyr::filter(time == time_point)

  # Forest plot
  plot <- data_plot %>%
```

```r
  ggplot(aes(
    x = estimate,
    y = forcats::as_factor(term_tidy),
    xmin = conf.low,
    xmax = conf.high
  )) +
  geom_point() +
  geom_errorbarh() +
  geom_vline(xintercept = 0, linetype = "dotted") +
  theme_minimal() +
  xlab("Risk Difference (%) and 95% CI") +
  theme(
    # Title
    plot.title = element_text(size = 10),
    plot.subtitle = element_text(size = 8),
    # Axes
    axis.title.y = element_blank()
  )

# RD values
rd <- data_plot %>%
  ggplot() +
  geom_text(aes(
    x = 0,
    y = forcats::as_factor(term_tidy),
    label = conf.interval.tidy
  )) +
  theme_void() +
  theme(plot.title = element_text(hjust = 0.5))

return(list(plot = plot, rd = rd))
}

broom.cols <- c("Term",
                "Estimate",
                "Std error",
                "Statistic",
                "p-value",
                "Lower",
                "Upper")
```

Survival Analysis

Shapes of continuous variables

Using splines for the continuous variables. Testing the splines significance using a likelihood ratio test.

Crohn's disease

Patient reported flare

Before examining dietary exposures, we first assess whether continuous covariates (age, BMI, FC, and diet quality index) exhibit non-linear relationships with flare risk. Natural splines with 2 degrees of freedom are tested using likelihood ratio tests (LRT). Where significant, spline terms are retained in subsequent dietary models to improve model fit and reduce confounding.

Crohn's disease

Patient-reported flare

```
# Just the continuous variables
cox_cd_soft <- coxph(
  Surv(softflare_time, softflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    dqi_tot +
    frailty(SiteNo),
  data = flare.cd.df
)

# Is there any benefit to using spline for:

# age
summon_lrt(cox_cd_soft,
           remove = "age_decade",
           add = "ns(age_decade, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	age_decade	ns(age_decade, df = 2)	-1012.677	0.91	1	0.34

```
# BMI
summon_lrt(cox_cd_soft, remove = "BMI", add = "ns(BMI, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	BMI	ns(BMI, df = 2)	-1013.132	0.001	1	0.976

```
# FC
summon_lrt(cox_cd_soft, remove = "FC", add = "ns(FC, df = 2)")
```
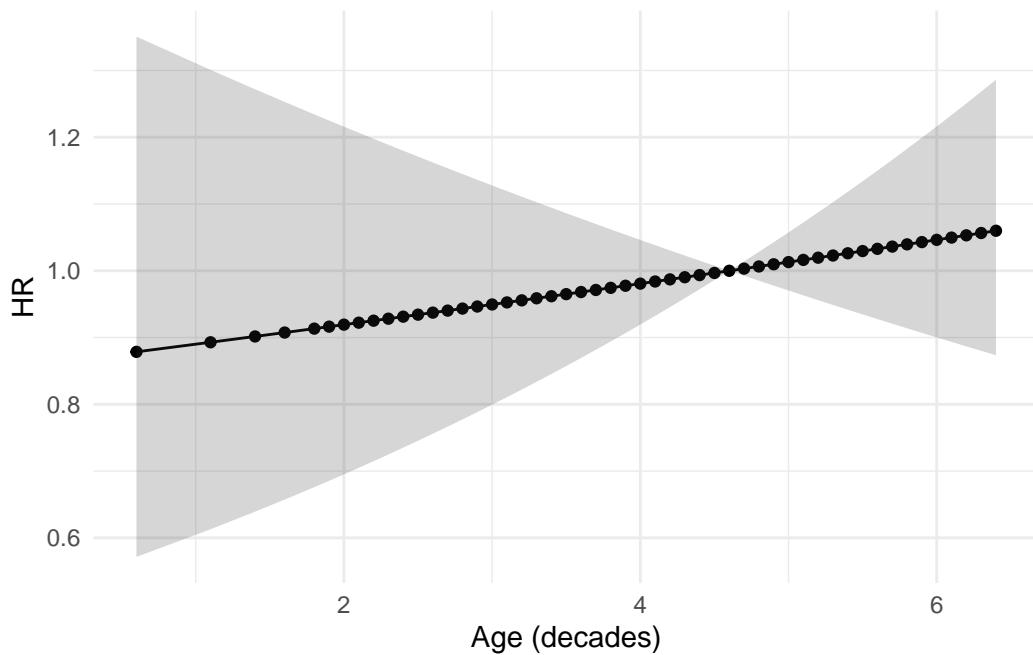
Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	FC	ns(FC, df = 2)	-1013.113	0.038	1	0.845

```
# dqi_tot
summon_lrt(cox_cd_soft,
           remove = "dqi_tot",
           add = "ns(dqi_tot, df = 2)")
```
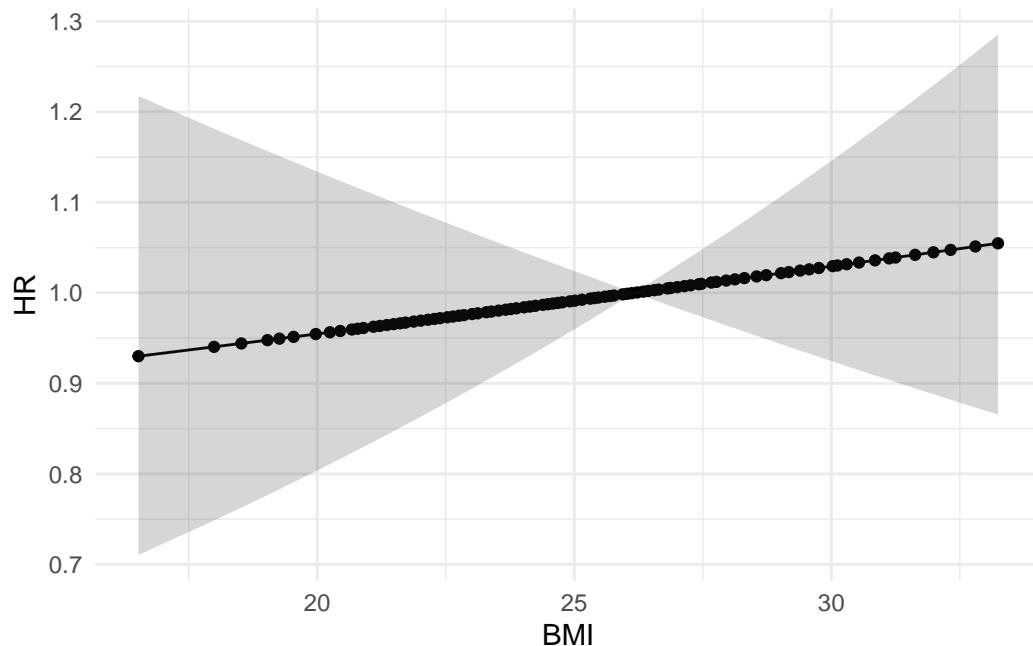
Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	dqi_tot	ns(dqi_tot, df = 2)	-1012.826	0.612	1	0.434

```
# None of the continuous variables benefit from a spline term
```
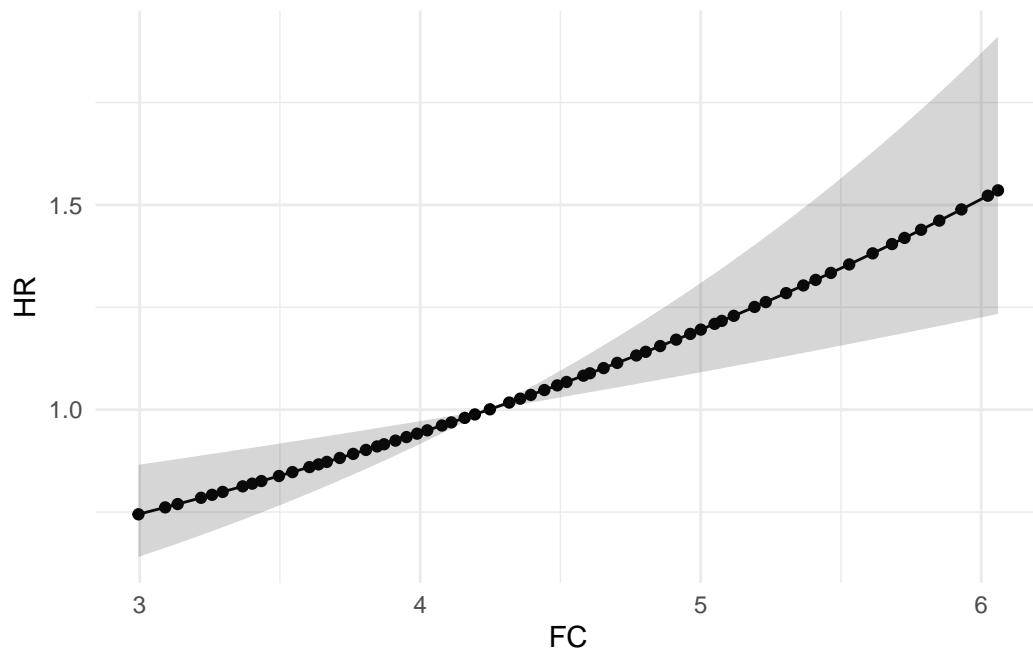
```
# Age
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_cd_soft,
  variable = "age_decade"
) +
  xlab("Age (decades)")
```
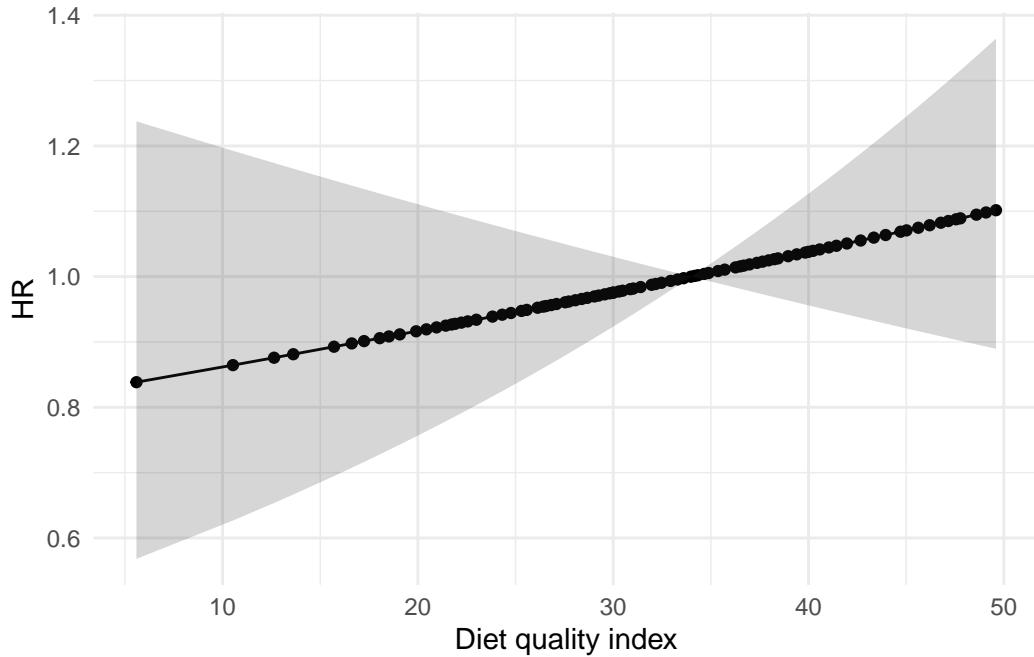
```
# BMI
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_cd_soft,
  variable = "BMI"
)
```

```
# FC
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_cd_soft,
  variable = "FC"
)
```

```
# DQI
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_cd_soft,
  variable = "dqi_tot"
) +
  xlab("Diet quality index")
```

Objective flare

```
# Just the continuous variables

cox_cd_hard <- coxph(
  Surv(hardflare_time, hardflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    dqi_tot +
    frailty(SiteNo),
  data = flare.cd.df
)

# Is there any benefit to using spline for:

# age
summon_lrt(cox_cd_hard, remove = "age_decade", add = "ns(age_decade, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	age_decade	ns(age_decade, df = 2)	-591.537	0.369	1.019	0.551

```
# BMI
summon_lrt(cox_cd_hard, remove = "BMI", add = "ns(BMI, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	BMI	ns(BMI, df = 2)	-591.833	0.221	0.796	0.545

```
# FC
summon_lrt(cox_cd_hard, remove = "FC", add = "ns(FC, df = 2)")
```
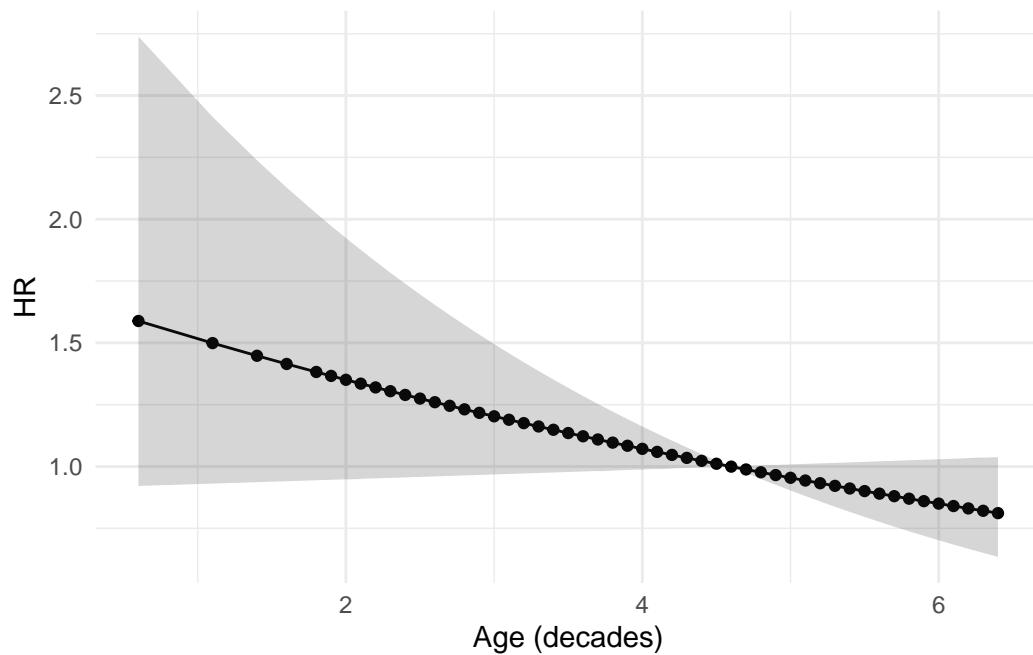
Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	FC	ns(FC, df = 2)	-591.243	0.958	1.282	0.419

```
# dqi_tot
summon_lrt(cox_cd_hard, remove = "dqi_tot", add = "ns(dqi_tot, df = 2)")
```
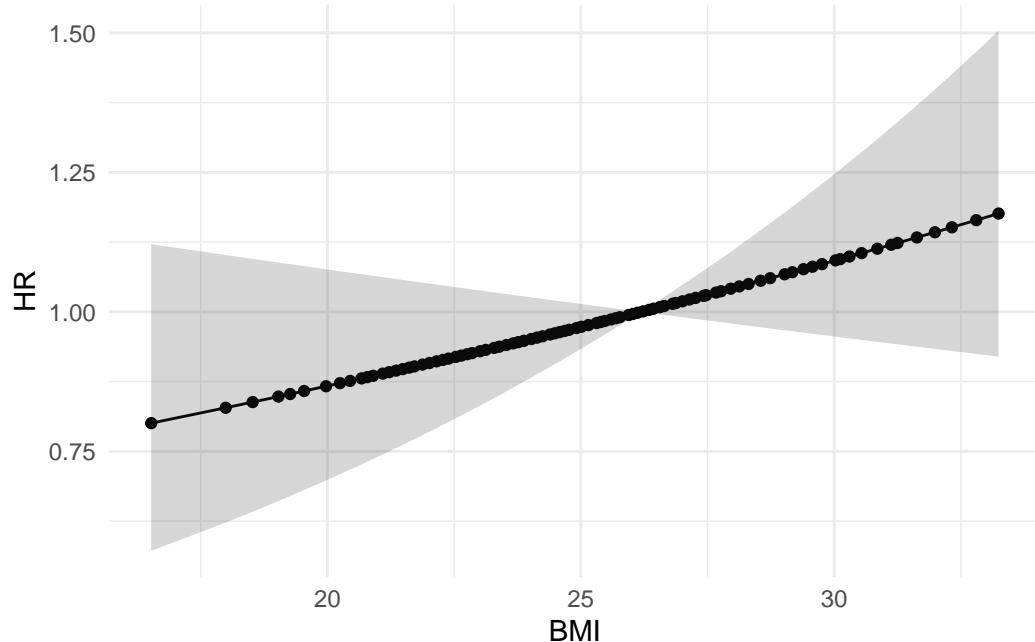
Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	dqi_tot	ns(dqi_tot, df = 2)	-591.489	0.465	1.058	0.518

```
# None of the continuous variables benefit from a spline term
```
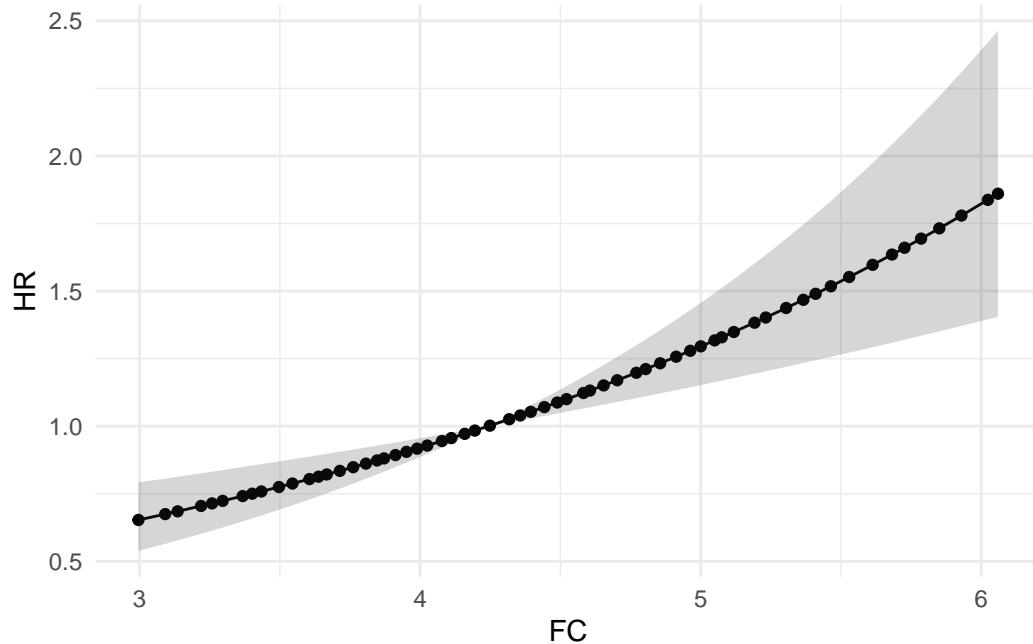
```
# Age
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_cd_hard,
  variable = "age_decade"
) +
  xlab("Age (decades)")
```
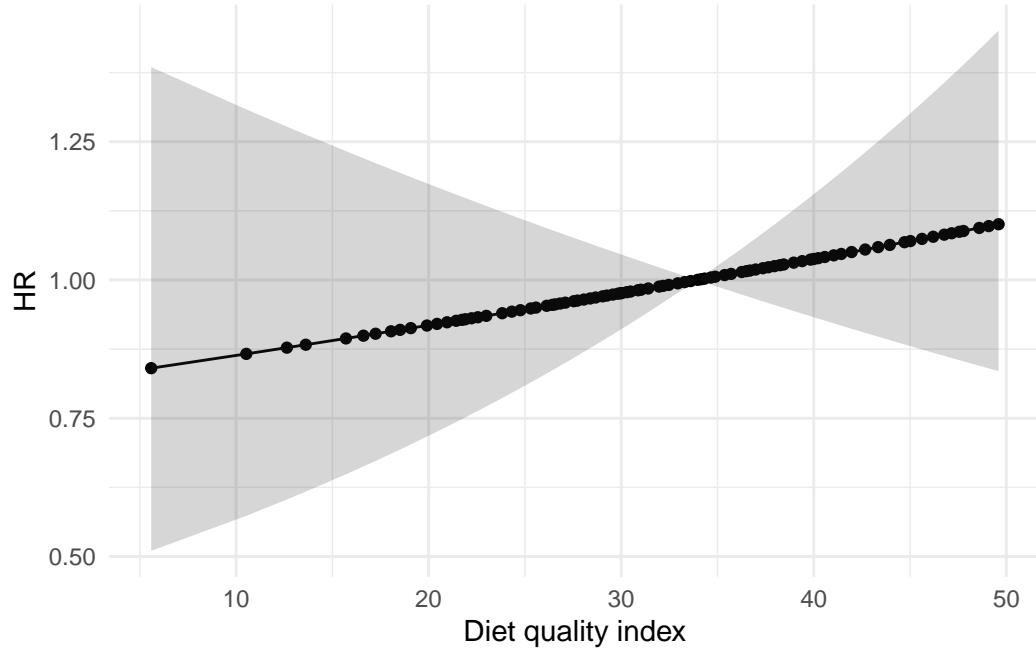
```
# BMI
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_cd_hard,
  variable = "BMI"
)
```

```
# FC
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_cd_hard,
  variable = "FC"
)
```

```
# DQI
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_cd_hard,
  variable = "dqi_tot"
) +
  xlab("Diet quality index")
```

Ulcerative colitis

Patient-reported flare

```r
# Just the continuous variables

cox_uc_soft <- coxph(
  Surv(softflare_time, softflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    dqi_tot +
    frailty(SiteNo),
  data = flare.uc.df
)

# Is there any benefit to using spline for:

# age
summon_lrt(cox_uc_soft, remove = "age_decade", add = "ns(age_decade, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	age_decade	ns(age_decade, df = 2)	-1228.481	0.37	1	0.543

```
# BMI
summon_lrt(cox_uc_soft, remove = "BMI", add = "ns(BMI, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	BMI	ns(BMI, df = 2)	-1228.655	0.022	1	0.881

```
# FC
summon_lrt(cox_uc_soft, remove = "FC", add = "ns(FC, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	FC	ns(FC, df = 2)	-1224.712	7.909	1	0.005

```
# dqi_tot
summon_lrt(cox_uc_soft, remove = "dqi_tot", add = "ns(dqi_tot, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	dqi_tot	ns(dqi_tot, df = 2)	-1227.125	3.083	1	0.079

```
# FC and dqi_tot can get a spline
cox_uc_soft <- coxph(
  Surv(softflare_time, softflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    ns(FC, df = 2) +
    ns(dqi_tot, df = 2) +
    frailty(SiteNo),
  data = flare.uc.df
)

# Test for df = 3
# FC
summon_lrt(cox_uc_soft, remove = "ns(FC, df = 2)", add = "ns(FC, df = 3)")
```
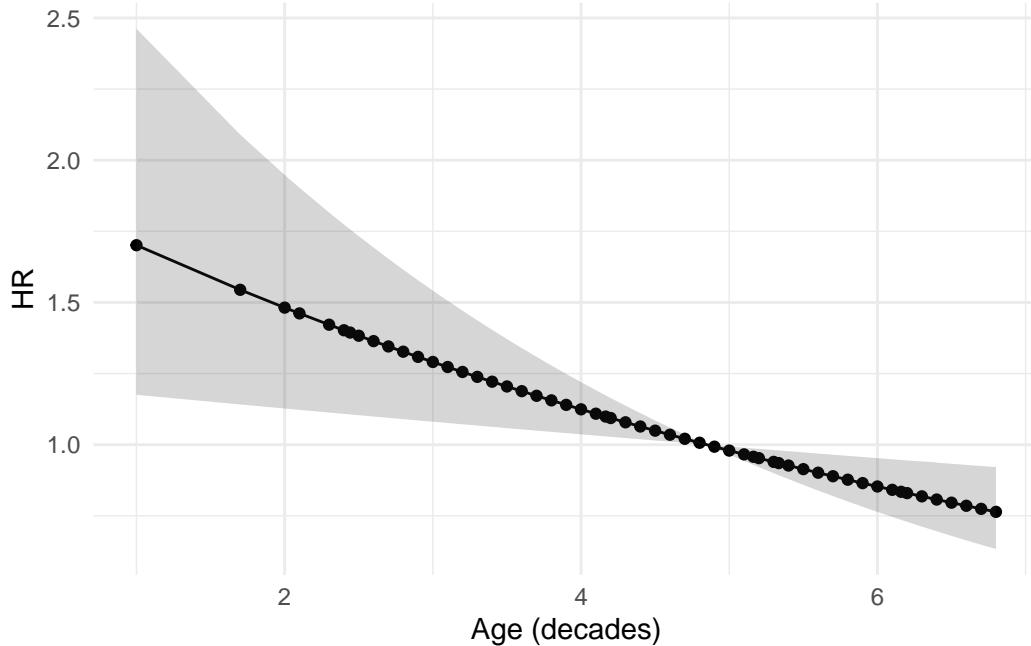
Test	Removled	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	ns(FC, df = 2)	ns(FC, df = 3)	-1223.038	0.091	1	0.762
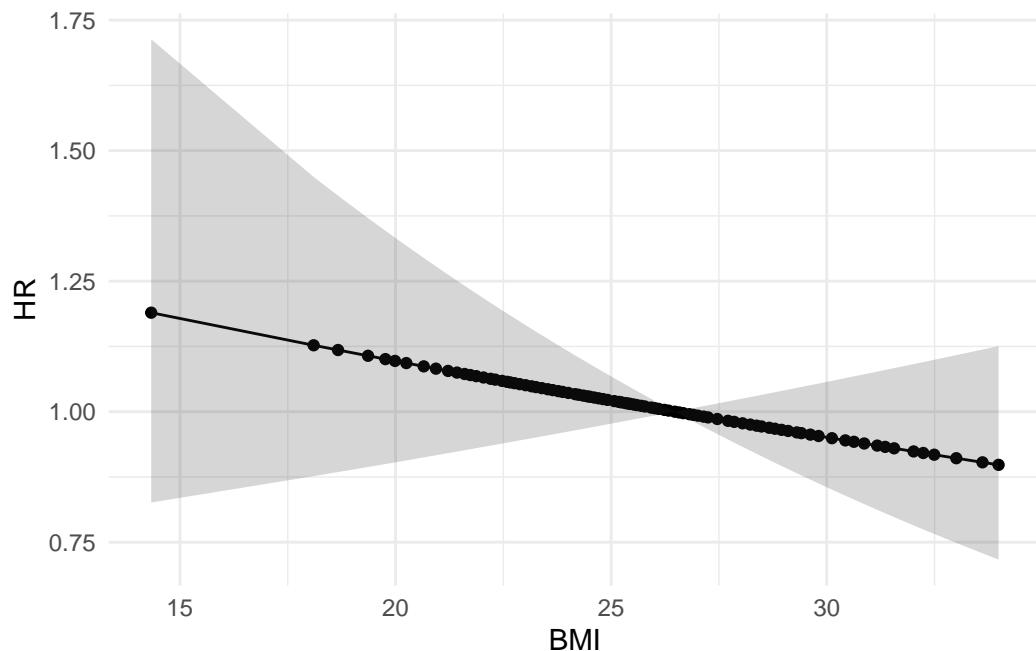
```
# dqi_tot
summon_lrt(cox_uc_soft,
  remove = "ns(dqi_tot, df = 2)",
  add = "ns(dqi_tot, df = 3)"
)
```

Test	Removled	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	ns(dqi_tot, df = 2)	ns(dqi_tot, df = 3)	-1221.851	2.465	1	0.116
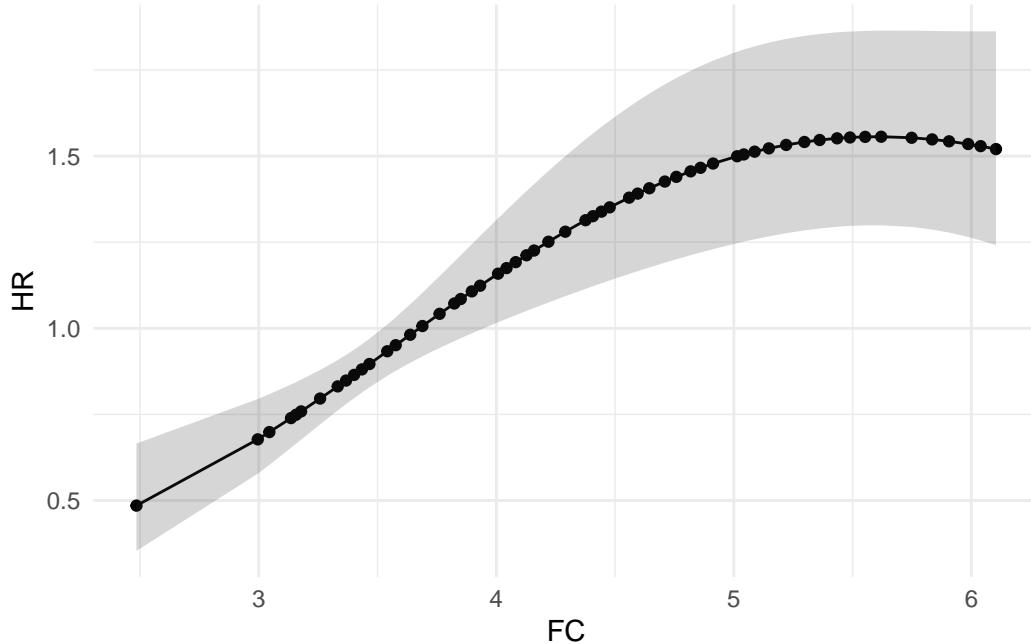
```
# Age
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_uc_soft,
  variable = "age_decade"
) +
  xlab("Age (decades)")
```
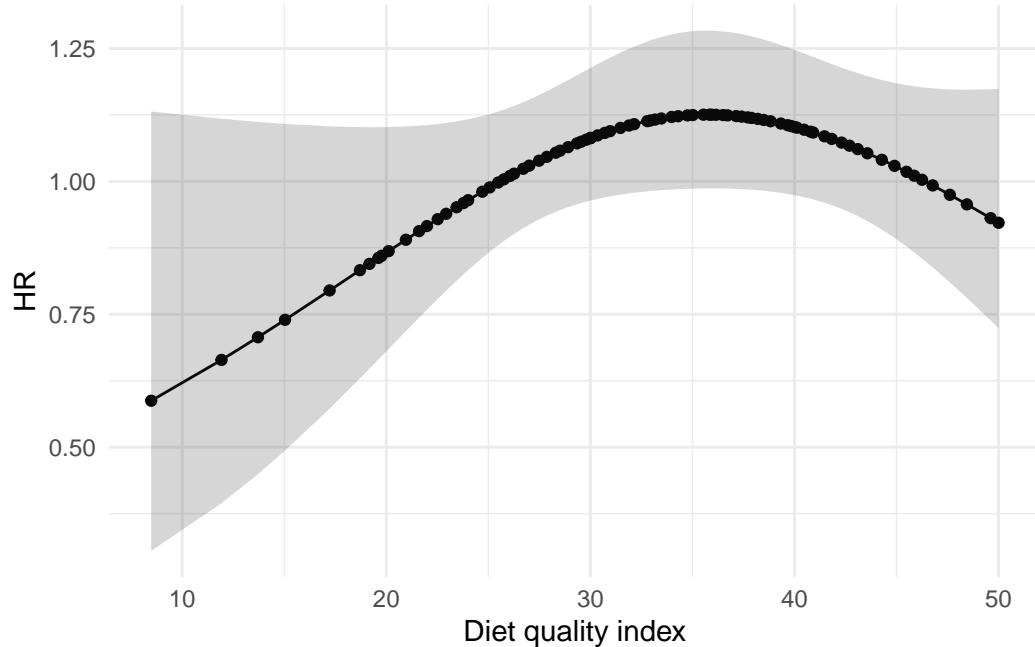
```
# BMI
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_uc_soft,
  variable = "BMI"
)
```



```
# FC
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_uc_soft,
  variable = "FC",
  splineterm = "ns(FC, df = 2)"
)
```

```
# DQI
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_uc_soft,
  variable = "dqi_tot",
  splineterm = "ns(dqi_tot, df = 2)"
) +
  xlab("Diet quality index")
```

```
#### Objective flare
```

```
# Just the continuous variables

cox_uc_hard <- coxph(
  Surv(hardflare_time, hardflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    dqi_tot +
    frailty(SiteNo),
  data = flare.uc.df
)

# Is there any benefit to using spline for:

# age
summon_lrt(cox_uc_hard, remove = "age_decade", add = "ns(age_decade, df = 2)")
```

28

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	age__decade	ns(age__decade, df = 2)	-679.06	0.662	1.004	0.417

```
# BMI
summon_lrt(cox_uc_hard, remove = "BMI", add = "ns(BMI, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	BMI	ns(BMI, df = 2)	-679.35	0.083	1	0.773

```
# FC
summon_lrt(cox_uc_hard, remove = "FC", add = "ns(FC, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	FC	ns(FC, df = 2)	-674.362	10.059	1	0.002

```
# dqi_tot
summon_lrt(cox_uc_hard, remove = "dqi_tot", add = "ns(dqi_tot, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	dqi__tot	ns(dqi__tot, df = 2)	-679.074	0.635	1	0.426

```
# FC benefits from a spline
cox_uc_hard <- coxph(
  Surv(hardflare_time, hardflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    ns(FC, df = 2) +
    dqi_tot +
    frailty(SiteNo),
  data = flare.uc.df
)


# FC
summon_lrt(cox_uc_hard, remove = "ns(FC, df = 2)", add = "ns(FC, df = 3)")
```
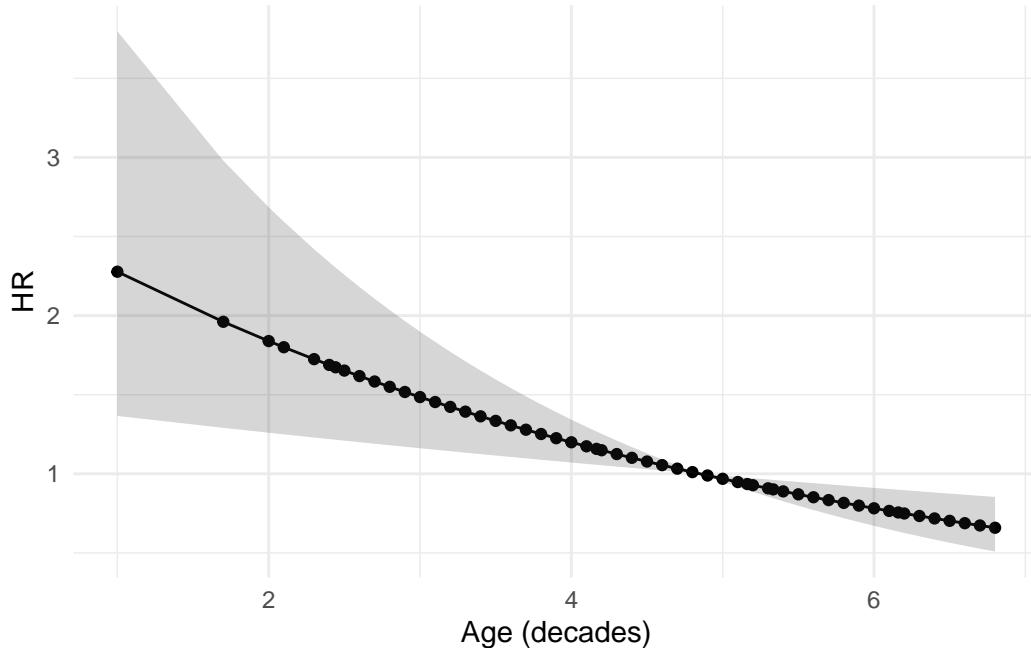
Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	ns(FC, df = 2)	ns(FC, df = 3)	-673.943	0.838	1	0.36
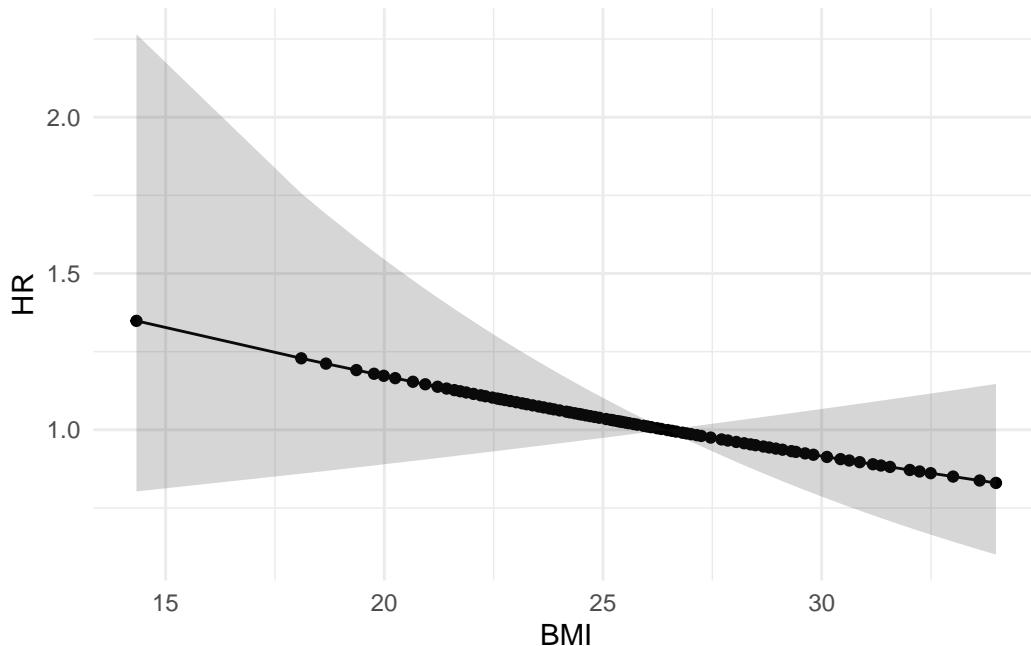
```
# Test for df = 3. df = 2 is sufficient for FC
# FC
summon_lrt(cox_uc_hard, remove = "ns(FC, df = 2)", add = "ns(FC, df = 3)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	ns(FC, df = 2)	ns(FC, df = 3)	-673.943	0.838	1	0.36
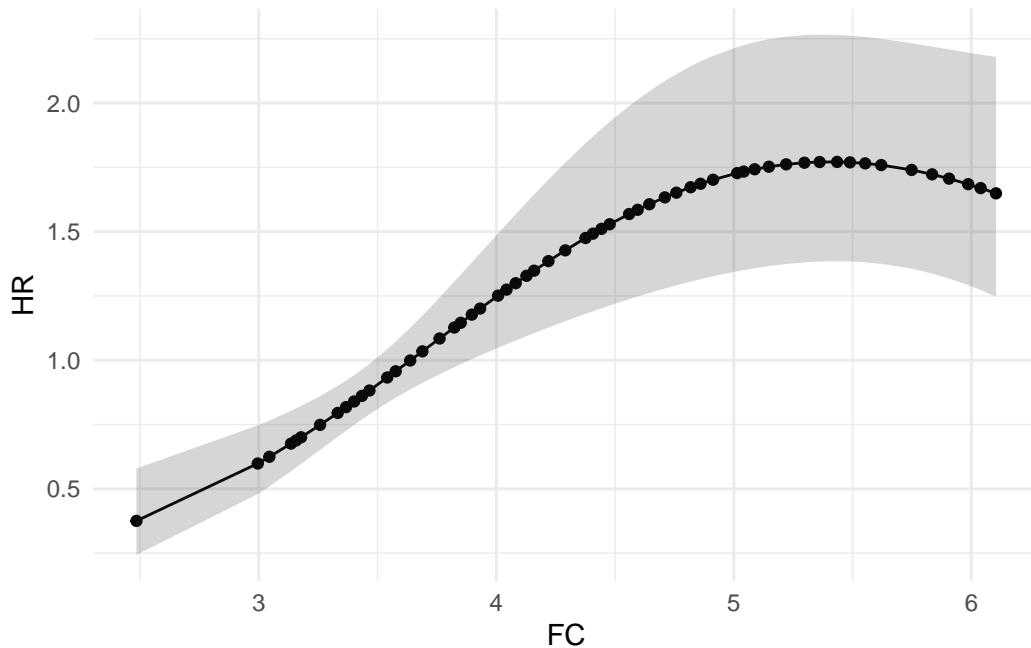
```
# Age
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_uc_hard,
  variable = "age_decade"
) +
  xlab("Age (decades)")
```

```
# BMI
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_uc_hard,
  variable = "BMI"
)
```
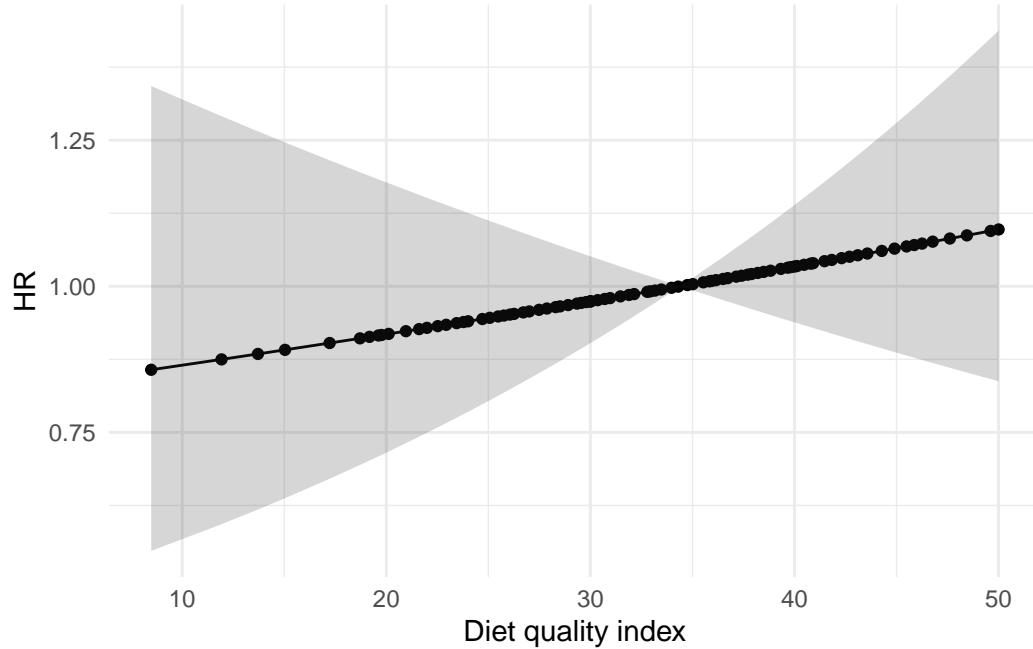


```
# FC
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_uc_hard,
  variable = "FC",
  splineterm = "ns(FC, df = 2)"
)
```

```
# DQI
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_uc_hard,
  variable = "dqi_tot"

) +
  xlab("Diet quality index")
```

Total meat protein

This section examines the association between total meat protein intake (grams per day) and time to flare. Cox models are adjusted for sex, Index of Multiple Deprivation (IMD), age, BMI, FC (continuous), and diet quality index, with a frailty term for study site. We first test whether the relationship is non-linear using splines, then present hazard ratios and continuous HR curves.

Crohn's disease

Patient reported flare

```
cox_meat_cd_soft <- coxph(
  Surv(softflare_time, softflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    Meat_sum +
    dqi_tot +
```

```
    frailty(SiteNo),
  data = flare.cd.df
)


# Do we need a spline? No
summon_lrt(cox_meat_cd_soft, remove = "Meat_sum", add = "ns(Meat_sum, df = 2)")
```

Test	Removed	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	Meat_sum	ns(Meat_sum, df = 2)	-1013.017	0.179	1	0.672

```
cox_meat_cd_soft %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```
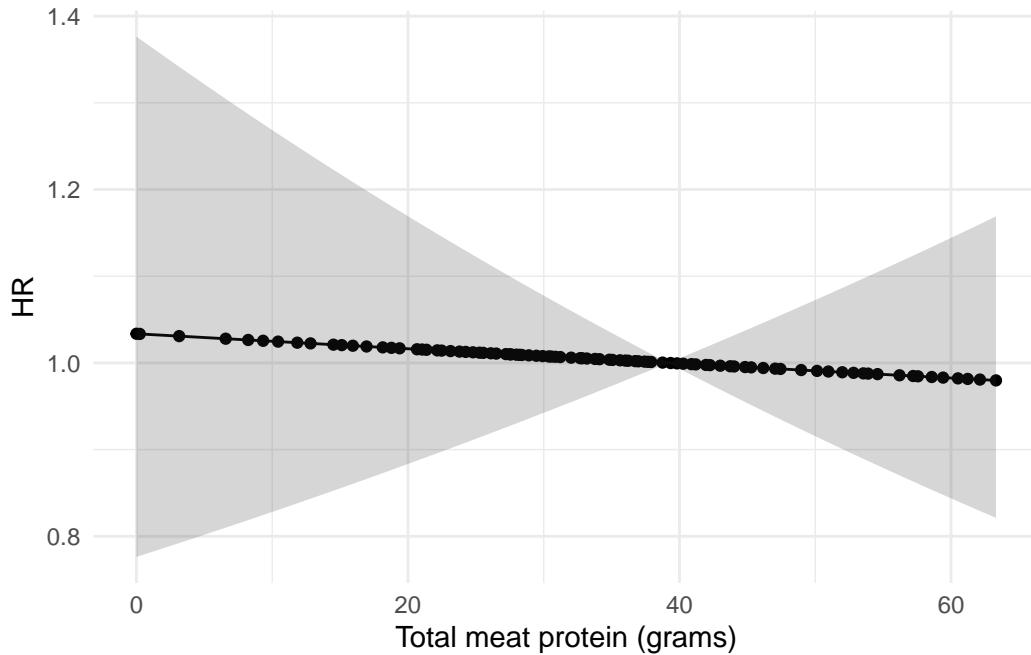
Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.873	0.170	13.561	0.000	1.341	2.617
IMD2	0.704	0.333	1.115	0.291	0.367	1.351
IMD3	0.911	0.306	0.093	0.760	0.500	1.658
IMD4	0.824	0.304	0.404	0.525	0.454	1.496
IMD5	0.954	0.284	0.027	0.869	0.547	1.666
age_decade	1.032	0.055	0.338	0.561	0.927	1.150
BMI	1.008	0.014	0.298	0.585	0.980	1.037
FC	1.267	0.061	14.821	0.000	1.123	1.429
Meat_sum	0.999	0.004	0.051	0.821	0.992	1.006
dqi_tot	1.006	0.007	0.698	0.403	0.992	1.020
frailty(SiteNo)	NA	NA	0.000	0.915	NA	NA

```
# Meat sum
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_meat_cd_soft,
  variable = "Meat_sum"

) +
  xlab("Total meat protein (grams)")
```

Objective flare

```
cox_meat_cd_hard <- coxph(
  Surv(hardflare_time, hardflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    Meat_sum +
    dqi_tot +
    frailty(SiteNo),
  data = flare.cd.df
)

# Do we need a spline? No
summon_lrt(cox_meat_cd_hard, remove = "Meat_sum", add = "ns(Meat_sum, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	Meat_sum	ns(Meat_sum, df = 2)	-591.46	0.327	1.03	0.58

```
cox_meat_cd_hard %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```
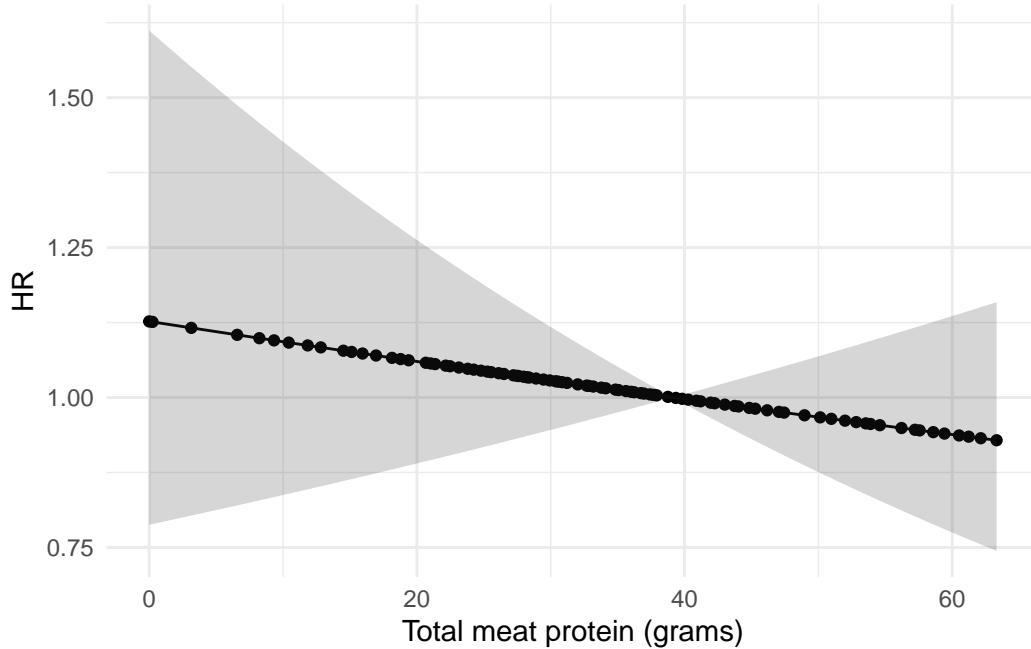
Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.274	0.210	1.327	0.249	0.844	1.922
IMD2	0.601	0.437	1.355	0.244	0.255	1.416
IMD3	0.759	0.407	0.459	0.498	0.342	1.684
IMD4	0.730	0.399	0.618	0.432	0.334	1.598
IMD5	0.716	0.376	0.789	0.374	0.342	1.497
age_decade	0.889	0.069	2.850	0.091	0.776	1.019
BMI	1.024	0.018	1.753	0.186	0.989	1.060
FC	1.411	0.079	19.058	0.000	1.209	1.647
Meat_sum	0.997	0.005	0.428	0.513	0.988	1.006
dqi_tot	1.005	0.009	0.356	0.551	0.988	1.023
frailty(SiteNo)	NA	NA	7.459	0.205	NA	NA

```
# Meat sum
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_meat_cd_hard,
  variable = "Meat_sum"
) +
  xlab("Total meat protein (grams)")
```

Ulcerative colitis

Patient reported flare

```
cox_meat_uc_soft <- coxph(
  Surv(softflare_time, softflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    ns(FC, df = 2) +
    Meat_sum +
    ns(dqi_tot, df = 2) +
    frailty(SiteNo),
  data = flare.uc.df
)

# Do we need a spline? No
summon_lrt(cox_meat_uc_soft, remove = "Meat_sum", add = "ns(Meat_sum, df = 2)")
```
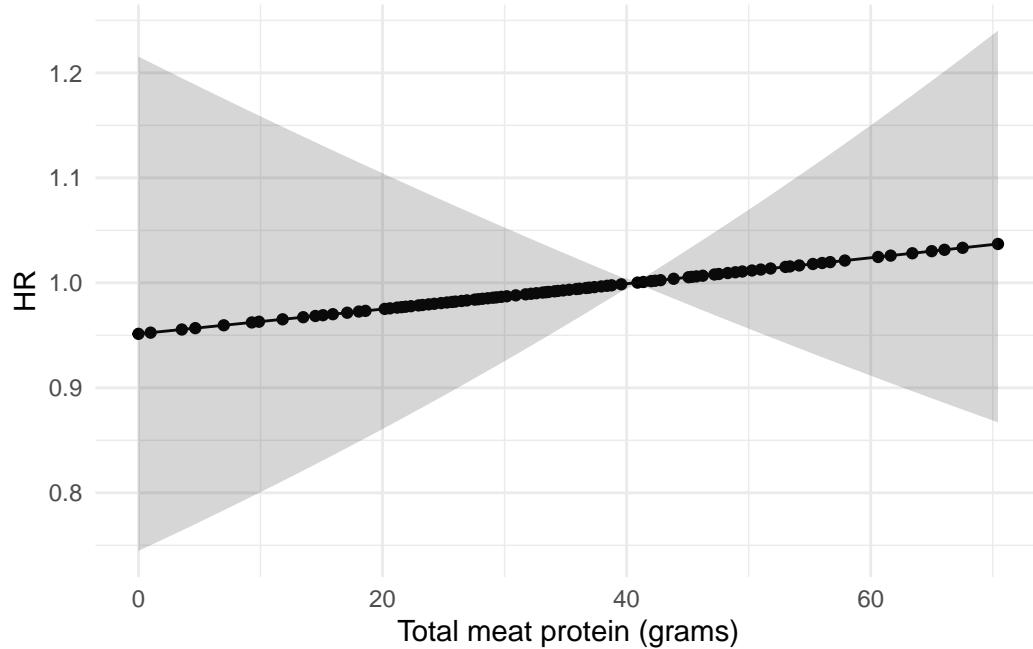
Test	Removled	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	Meat_sum	ns(Meat_sum, df = 2)	-1222.919	0.171	1	0.679

```
cox_meat_uc_soft %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```

Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.639	0.149	10.968	0.001	1.224	2.196
IMD2	1.323	0.332	0.711	0.399	0.690	2.538
IMD3	1.114	0.324	0.112	0.738	0.591	2.102
IMD4	1.301	0.306	0.738	0.390	0.714	2.369
IMD5	1.180	0.302	0.301	0.583	0.653	2.131
age_decade	0.872	0.049	7.866	0.005	0.792	0.959
BMI	0.985	0.015	0.921	0.337	0.956	1.016
ns(FC, df = 2)1	7.646	0.426	22.820	0.000	3.319	17.616
ns(FC, df = 2)2	0.856	0.578	0.073	0.788	0.275	2.658
Meat_sum	1.001	0.003	0.159	0.690	0.995	1.007
ns(dqi_tot, df = 2)1	2.971	0.698	2.431	0.119	0.756	11.679
ns(dqi_tot, df = 2)2	0.797	0.336	0.457	0.499	0.412	1.539
frailty(SiteNo)	NA	NA	0.000	0.913	NA	NA

```
# Meat sum
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_meat_uc_soft,
  variable = "Meat_sum"
) +
  xlab("Total meat protein (grams)")
```

Objective flare

```r
cox_meat_uc_hard <- coxph(
  Surv(hardflare_time, hardflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    ns(FC, df = 2) +
    Meat_sum +
    dqi_tot +
    frailty(SiteNo),
  data = flare.uc.df
)

# Do we need a spline? No
summon_lrt(cox_meat_uc_hard, remove = "Meat_sum", add = "ns(Meat_sum, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	Meat_sum	ns(Meat_sum, df = 2)	-671.444	0.038	1.002	0.846

```
cox_meat_uc_hard %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```
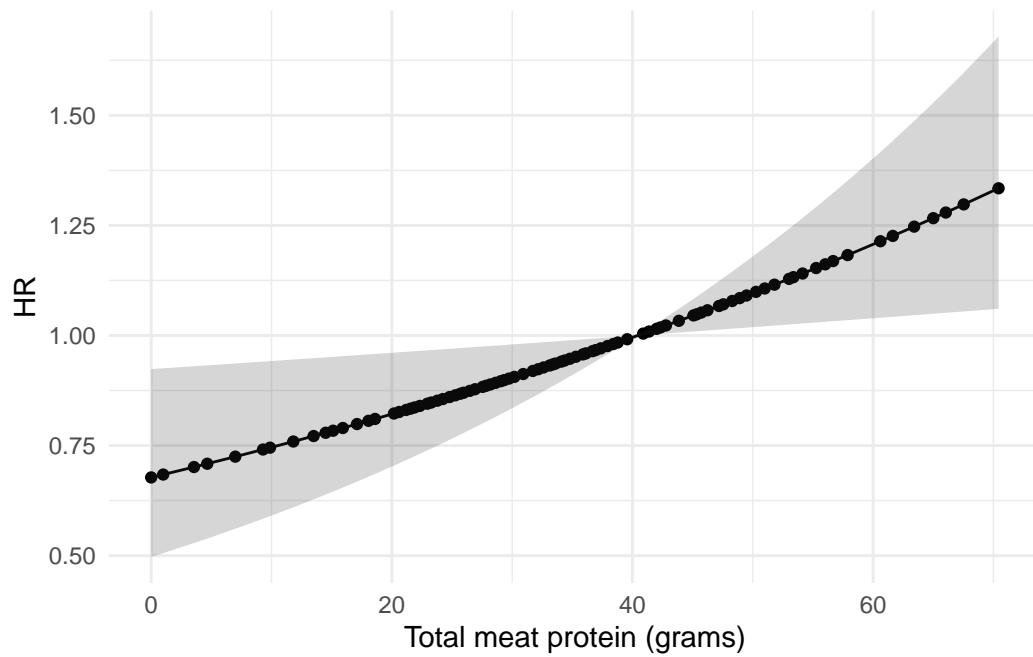
Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.236	0.195	1.184	0.277	0.844	1.812
IMD2	1.535	0.493	0.755	0.385	0.584	4.035
IMD3	1.361	0.480	0.412	0.521	0.531	3.488
IMD4	2.428	0.448	3.922	0.048	1.009	5.842
IMD5	1.582	0.449	1.046	0.306	0.657	3.811
age_decade	0.816	0.067	9.082	0.003	0.715	0.931
BMI	0.972	0.022	1.715	0.190	0.931	1.014
ns(FC, df = 2)1	12.562	0.584	18.782	0.000	4.000	39.455
ns(FC, df = 2)2	0.685	0.799	0.224	0.636	0.143	3.281
Meat_sum	1.010	0.004	6.062	0.014	1.002	1.017
dqi_tot	1.011	0.009	1.516	0.218	0.993	1.030
frailty(SiteNo)	NA	NA	0.159	0.423	NA	NA
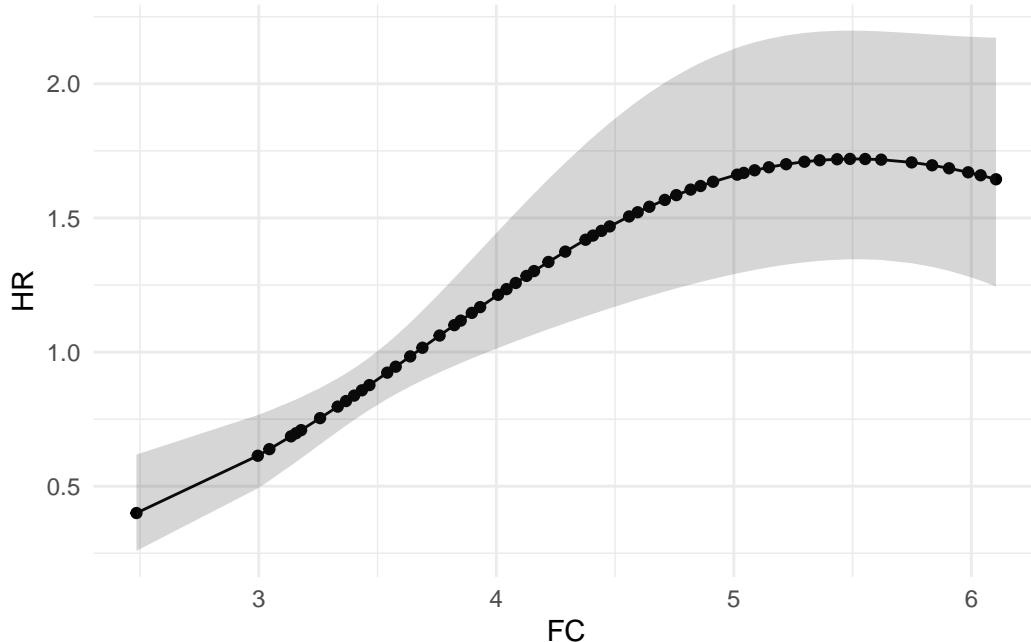
```
# Meat sum
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_meat_uc_hard,
  variable = "Meat_sum"
) +
  xlab("Total meat protein (grams)")
```

```
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_meat_uc_hard,
  variable = "FC",
  splineterm = "ns(FC, df = 2)"
)
```

Dietary Fibre

This section examines the association between dietary fibre intake (grams per day) and time to flare, using the same covariate adjustment strategy as the meat protein analysis.

Crohn's disease

Patient-reported flare

```
cox_fibre_cd_soft <- coxph(
  Surv(softflare_time, softflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    fibre +
    dqi_tot +
    frailty(SiteNo),
  data = flare.cd.df
)
```
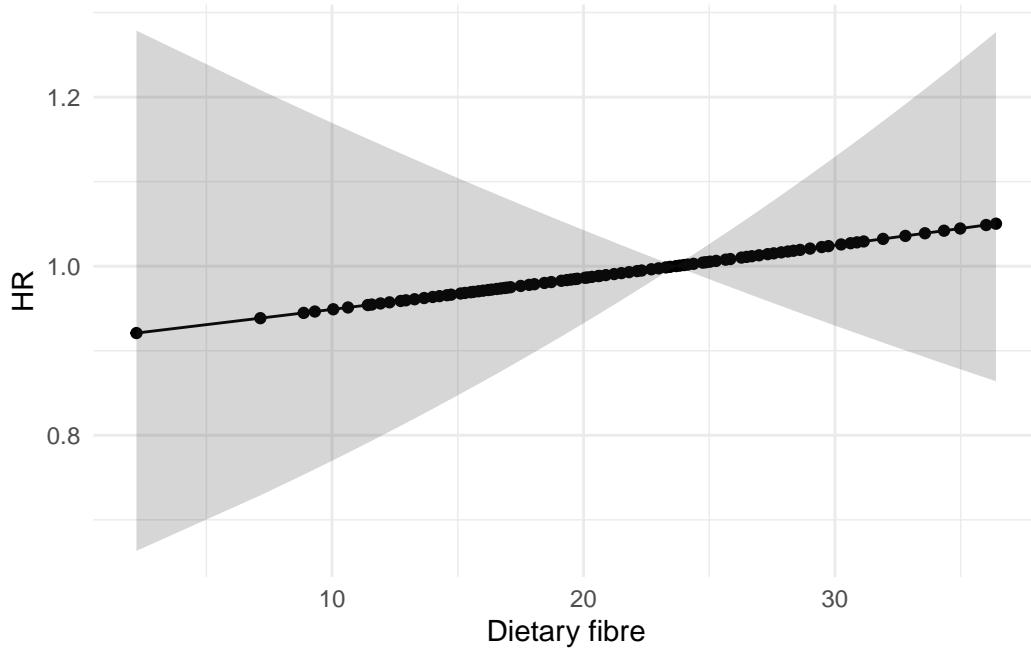
```
# Do we need a spline? No
summon_lrt(cox_fibre_cd_soft, remove = "fibre", add = "ns(fibre, df = 2)")
```

Test	Removled	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	fibre	ns(fibre, df = 2)	-1013.013	0.002	1	0.965

```
cox_fibre_cd_soft %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```

Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.897	0.171	14.013	0.000	1.357	2.653
IMD2	0.712	0.332	1.053	0.305	0.372	1.363
IMD3	0.910	0.306	0.095	0.758	0.500	1.657
IMD4	0.826	0.304	0.394	0.530	0.456	1.499
IMD5	0.962	0.284	0.018	0.892	0.552	1.678
age_decade	1.034	0.055	0.371	0.542	0.928	1.152
BMI	1.007	0.014	0.268	0.605	0.980	1.036
FC	1.268	0.062	14.852	0.000	1.124	1.431
fibre	1.004	0.008	0.242	0.623	0.989	1.019
dqi_tot	1.005	0.008	0.376	0.540	0.990	1.020
frailty(SiteNo)	NA	NA	0.000	0.914	NA	NA

```
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_fibre_cd_soft,
  variable = "fibre"
) +
  xlab("Dietary fibre")
```

Objective Flare

```
cox_fibre_cd_hard <- coxph(
  Surv(hardflare_time, hardflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    fibre +
    dqi_tot +
    frailty(SiteNo),
  data = flare.cd.df
)

summon_lrt(cox_fibre_cd_hard, remove = "fibre", add = "ns(fibre, df = 2)")
```

Test	Removed	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	fibre	ns(fibre, df = 2)	-590.177	2.044	1.014	0.155

```
cox_fibre_cd_hard %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```
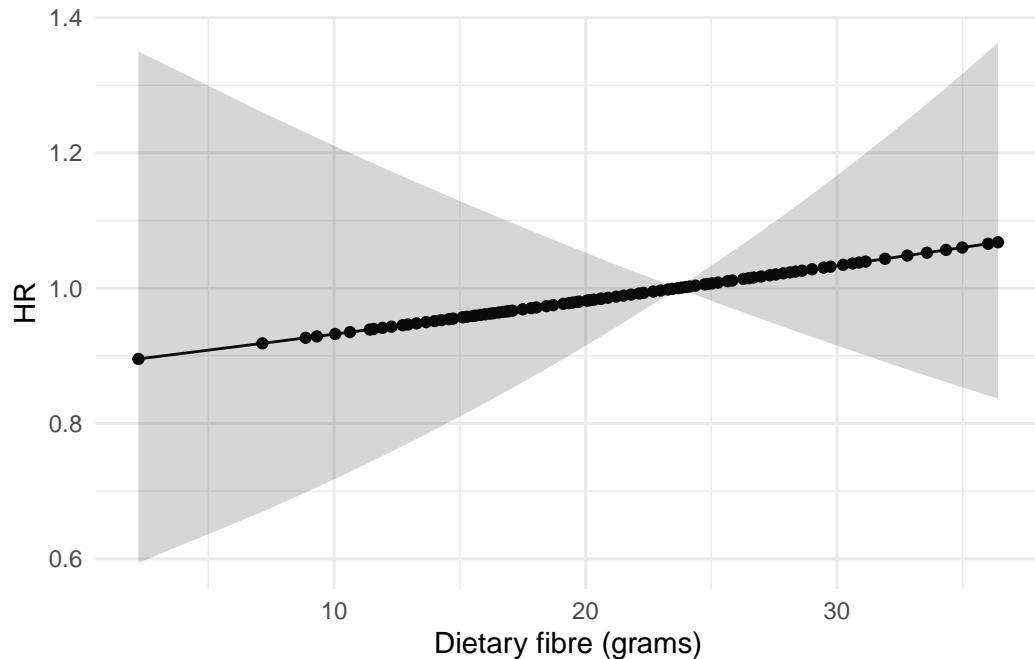
Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.310	0.212	1.625	0.202	0.865	1.983
IMD2	0.651	0.432	0.983	0.321	0.279	1.520
IMD3	0.798	0.407	0.309	0.578	0.360	1.770
IMD4	0.788	0.396	0.363	0.547	0.363	1.712
IMD5	0.765	0.374	0.514	0.474	0.367	1.592
age_decade	0.892	0.070	2.693	0.101	0.778	1.022
BMI	1.023	0.018	1.647	0.199	0.988	1.059
FC	1.410	0.079	18.897	0.000	1.208	1.647
fibre	1.005	0.010	0.278	0.598	0.986	1.025
dqi_tot	1.004	0.010	0.164	0.686	0.985	1.024
frailty(SiteNo)	NA	NA	8.196	0.194	NA	NA

```
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_fibre_cd_hard,
  variable = "fibre"
) +
  xlab("Dietary fibre (grams)")
```

Ulcerative colitis

Patient-reported flare

```
cox_fibre_uc_soft <- coxph(
  Surv(softflare_time, softflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    ns(FC, df = 2) +
    fibre +
    ns(dqi_tot, df = 2) +
    frailty(SiteNo),
  data = flare.uc.df
)

summon_lrt(cox_fibre_uc_soft, remove = "fibre", add = "ns(fibre, df = 2)")
```
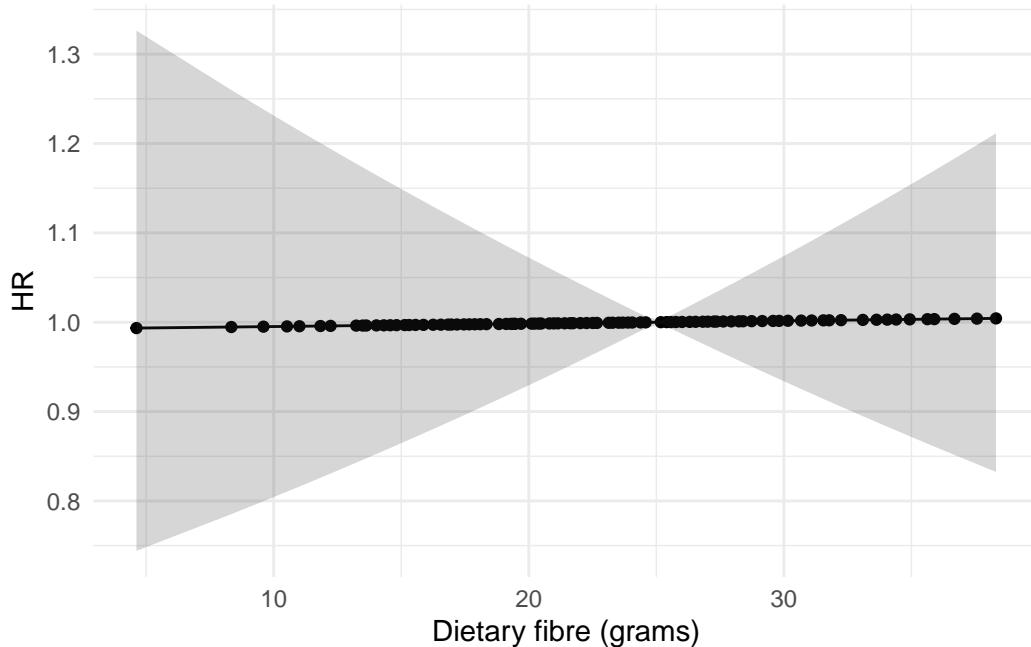
Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	fibre	ns(fibre, df = 2)	-1223.058	0.048	1	0.827

```
cox_fibre_uc_soft %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```

Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.629	0.149	10.737	0.001	1.217	2.181
IMD2	1.302	0.330	0.641	0.423	0.682	2.484
IMD3	1.106	0.324	0.097	0.755	0.587	2.086
IMD4	1.284	0.304	0.674	0.412	0.707	2.329
IMD5	1.171	0.302	0.273	0.601	0.648	2.116
age_decade	0.871	0.049	7.889	0.005	0.791	0.959
BMI	0.986	0.015	0.871	0.351	0.957	1.016
ns(FC, df = 2)1	7.708	0.426	23.033	0.000	3.348	17.749
ns(FC, df = 2)2	0.837	0.582	0.093	0.760	0.267	2.620
fibre	1.000	0.007	0.002	0.964	0.986	1.015
ns(dqi_tot, df = 2)1	2.840	0.710	2.161	0.142	0.706	11.421
ns(dqi_tot, df = 2)2	0.774	0.349	0.539	0.463	0.391	1.534
frailty(SiteNo)	NA	NA	0.000	0.914	NA	NA

```
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_fibre_uc_soft,
  variable = "fibre"
) +
  xlab("Dietary fibre (grams)")
```

Objective Flare

```r
cox_fibre_uc_hard <- coxph(
  Surv(hardflare_time, hardflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    ns(FC, df = 2) +
    fibre +
    dqi_tot +
    frailty(SiteNo),
  data = flare.uc.df
)

summon_lrt(cox_fibre_uc_hard, remove = "fibre", add = "ns(fibre, df = 2)")
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	fibre	ns(fibre, df = 2)	-674.129	0.424	1	0.515

```
cox_fibre_uc_hard %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```
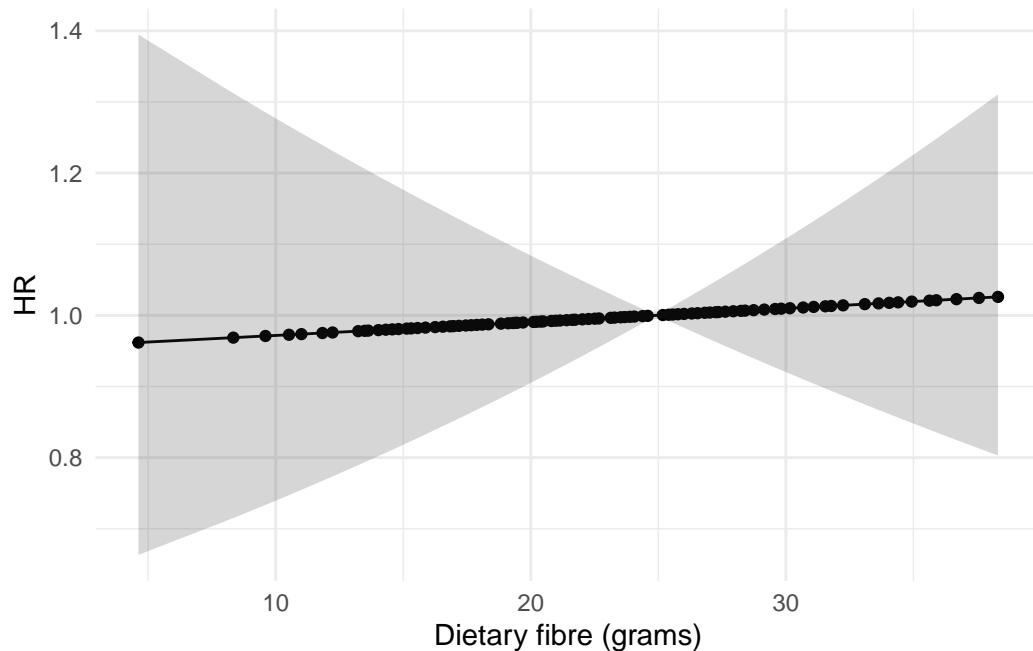
Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.164	0.194	0.617	0.432	0.797	1.702
IMD2	1.380	0.489	0.434	0.510	0.529	3.595
IMD3	1.240	0.477	0.204	0.652	0.487	3.157
IMD4	2.164	0.442	3.057	0.080	0.911	5.143
IMD5	1.469	0.446	0.745	0.388	0.613	3.519
age_decade	0.808	0.068	9.879	0.002	0.707	0.923
BMI	0.975	0.022	1.293	0.256	0.934	1.018
ns(FC, df = 2)1	14.266	0.585	20.638	0.000	4.532	44.905
ns(FC, df = 2)2	0.538	0.795	0.609	0.435	0.113	2.554
fibre	1.002	0.009	0.042	0.837	0.984	1.020
dqi_tot	1.005	0.009	0.318	0.573	0.987	1.024
frailty(SiteNo)	NA	NA	0.000	0.921	NA	NA

```
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_fibre_uc_hard,
  variable = "fibre"
) +
  xlab("Dietary fibre (grams)")
```

Polyunsaturated fatty acids

This section examines the association between PUFA intake (as percentage of total energy) and time to flare. Note that while the study protocol specified n-6 PUFAs, the available FFQ data captures total PUFA (both n-3 and n-6).

Crohn's disease

Patient-reported flare

```
cox_pufa_cd_soft <- coxph(
  Surv(softflare_time, softflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    PUFA_percEng +
    dqi_tot +
    frailty(SiteNo),
  data = flare.cd.df
```

```
)

# Do we need a spline? Close to 0.05, lets add one
summon_lrt(cox_pufa_cd_soft,
  remove = "PUFA_percEng",
  add = "ns(PUFA_percEng, df = 2)"
)
```

Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	PUFA_percEng	ns(PUFA_percEng, df = 2)	-1011.464	3.29	1	0.07

```
cox_pufa_cd_soft <- coxph(
  Surv(softflare_time, softflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    ns(PUFA_percEng, df = 2) +
    dqi_tot +
    frailty(SiteNo),
  data = flare.cd.df
)

# df = 2 is sufficient
summon_lrt(cox_pufa_cd_soft,
  remove = "ns(PUFA_percEng, df = 2)",
  add = "ns(PUFA_percEng, df = 3)"
)
```

Test	Removed	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	ns(PUFA_percEng, df = 2)	ns(PUFA_percEng, df = 3)	-1011.482	0.036	1	0.85

```
cox_pufa_cd_soft %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```
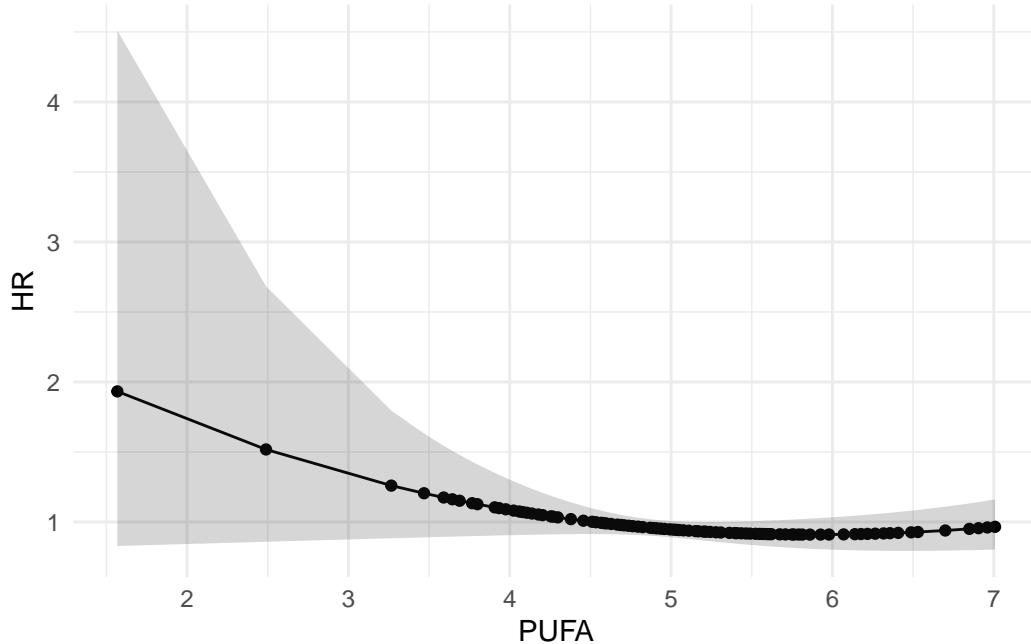
Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.916	0.170	14.588	0.000	1.373	2.676
IMD2	0.697	0.332	1.185	0.276	0.364	1.335
IMD3	0.919	0.306	0.076	0.782	0.504	1.674
IMD4	0.840	0.304	0.330	0.566	0.462	1.525
IMD5	0.973	0.285	0.009	0.924	0.557	1.701
age_decade	1.028	0.055	0.259	0.611	0.923	1.145
BMI	1.010	0.014	0.535	0.464	0.983	1.039
FC	1.263	0.062	14.160	0.000	1.118	1.426
ns(PUFA_percEng, df = 2)1	0.417	0.858	1.041	0.308	0.077	2.240
ns(PUFA_percEng, df = 2)2	3.814	0.778	2.963	0.085	0.831	17.507
dqi_tot	1.005	0.007	0.453	0.501	0.991	1.019
frailty(SiteNo)	NA	NA	0.000	0.914	NA	NA

```
# Significance - need to use LRT
summon_lrt(cox_pufa_cd_soft, remove = "ns(PUFA_percEng, df = 2)")
```

Test	Removed	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	ns(PUFA_percEng, df = 2)	NA	-1013.132	3.338	2	0.188

```
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_pufa_cd_soft,
  variable = "PUFA_percEng",
  splineterm = "ns(PUFA_percEng, df = 2)"
) +
  xlab("PUFA")
```

Objective Flare

```r
cox_pufa_cd_hard <- coxph(
  Surv(hardflare_time, hardflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    PUFA_percEng +
    dqi_tot +
    frailty(SiteNo),
  data = flare.cd.df
)

# Do we need a spline? No
summon_lrt(cox_pufa_cd_hard,
  remove = "PUFA_percEng",
  add = "ns(PUFA_percEng, df = 2)"
)
```
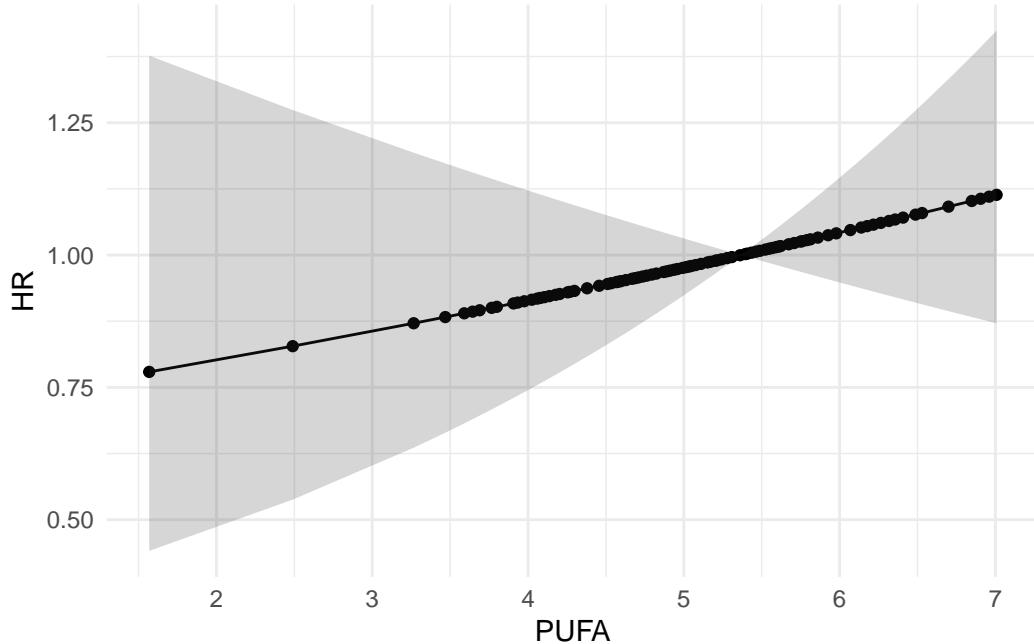
| Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	PUFA_percEng	s(PUFA_percEng, df = 2)	-590.441	0.698	0.891	0.362

```
cox_pufa_cd_hard %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```

Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.298	0.210	1.544	0.214	0.860	1.957
IMD2	0.657	0.431	0.952	0.329	0.282	1.528
IMD3	0.822	0.409	0.231	0.631	0.369	1.830
IMD4	0.811	0.398	0.279	0.597	0.372	1.767
IMD5	0.778	0.374	0.450	0.502	0.374	1.619
age_decade	0.895	0.070	2.530	0.112	0.780	1.026
BMI	1.024	0.018	1.837	0.175	0.989	1.061
FC	1.396	0.079	17.689	0.000	1.195	1.632
PUFA_percEng	1.068	0.076	0.738	0.390	0.919	1.240
dqi_tot	1.006	0.009	0.401	0.526	0.988	1.024
frailty(SiteNo)	NA	NA	8.580	0.183	NA	NA

```
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_pufa_cd_hard,
  variable = "PUFA_percEng"
) +
  xlab("PUFA")
```

Ulcerative colitis

Patient-reported flare

```
cox_pufa_uc_soft <- coxph(
  Surv(softflare_time, softflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    ns(FC, df = 2) +
    PUFA_percEng +
    ns(dqi_tot, df = 2) +
    frailty(SiteNo),
  data = flare.uc.df
)

# Do we need a spline? No
summon_lrt(cox_pufa_uc_soft,
  remove = "PUFA_percEng",
  add = "ns(PUFA_percEng, df = 2)"
)
```
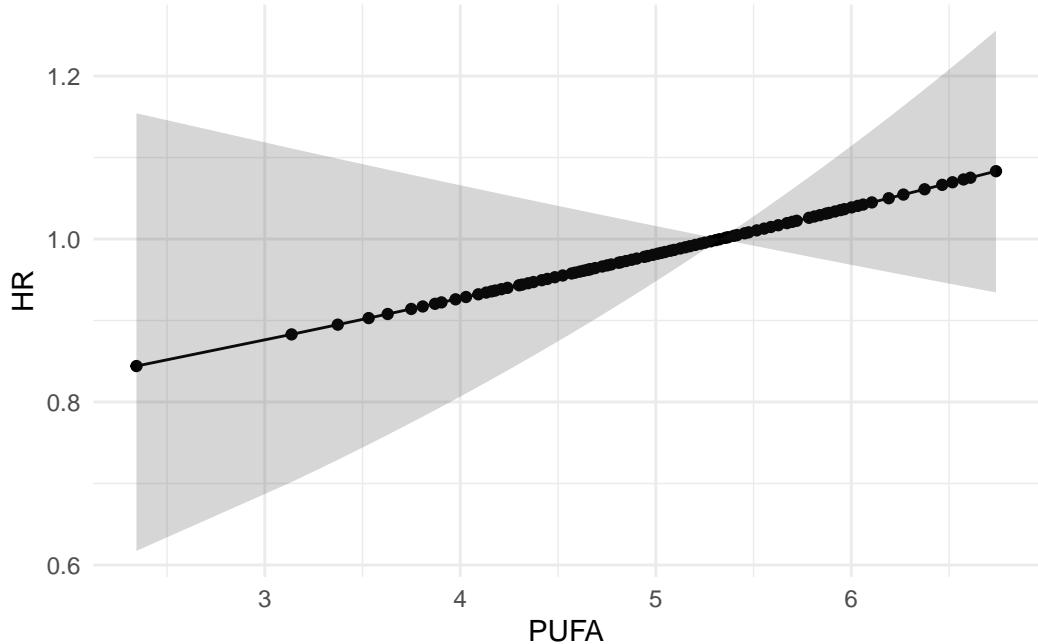
Test	Remolved	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	PUFA_percEng	ns(PUFA_percEng, df = 2)	-1222.426	0.221	1	0.638

```
cox_pufa_uc_soft %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```

Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.628	0.148	10.807	0.001	1.218	2.178
IMD2	1.340	0.331	0.783	0.376	0.701	2.561
IMD3	1.112	0.323	0.108	0.743	0.590	2.096
IMD4	1.301	0.304	0.748	0.387	0.717	2.360
IMD5	1.185	0.301	0.316	0.574	0.657	2.137
age_decade	0.877	0.050	6.954	0.008	0.796	0.967
BMI	0.988	0.015	0.653	0.419	0.958	1.018
ns(FC, df = 2)1	8.000	0.428	23.648	0.000	3.460	18.497
ns(FC, df = 2)2	0.839	0.577	0.092	0.761	0.271	2.600
PUFA_percEng	1.058	0.053	1.126	0.289	0.953	1.175
ns(dqi_tot, df = 2)1	2.779	0.692	2.184	0.139	0.716	10.779
ns(dqi_tot, df = 2)2	0.782	0.331	0.549	0.459	0.409	1.497
frailty(SiteNo)	NA	NA	0.000	0.914	NA	NA

```
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_pufa_uc_soft,
  variable = "PUFA_percEng"
) +
  xlab("PUFA")
```

Objective Flare

```r
cox_pufa_uc_hard <- coxph(
  Surv(hardflare_time, hardflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    ns(FC, df = 2) +
    PUFA_percEng +
    dqi_tot +
    frailty(SiteNo),
  data = flare.uc.df
)

# Do we need a spline? No
summon_lrt(cox_pufa_uc_hard,
  remove = "PUFA_percEng",
  add = "ns(PUFA_percEng, df = 2)"
)
```
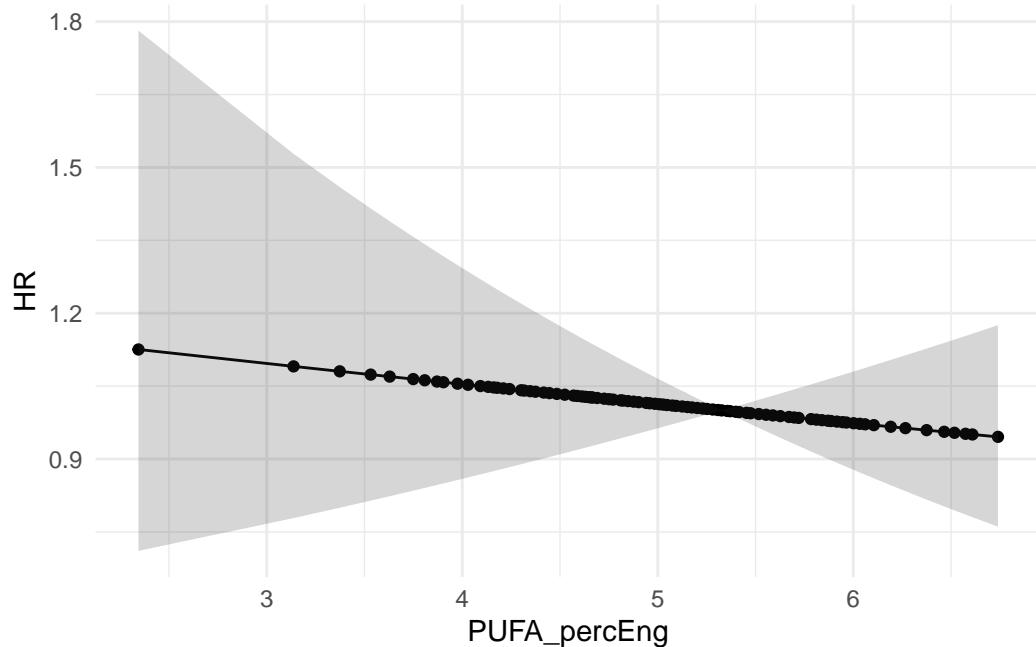
Test	Removled	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	PUFA_percEng	ns(PUFA_percEng, df = 2)	-673.44	1.585	1	0.208

```
cox_pufa_uc_hard %>%
  broom::tidy(exp = TRUE, conf.int = TRUE)  %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```

Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.161	0.193	0.600	0.439	0.796	1.695
IMD2	1.369	0.489	0.412	0.521	0.525	3.569
IMD3	1.241	0.476	0.205	0.651	0.488	3.157
IMD4	2.155	0.442	3.018	0.082	0.906	5.122
IMD5	1.471	0.446	0.750	0.387	0.614	3.524
age_decade	0.804	0.068	10.233	0.001	0.703	0.919
BMI	0.974	0.022	1.393	0.238	0.933	1.017
ns(FC, df = 2)1	14.159	0.585	20.538	0.000	4.500	44.548
ns(FC, df = 2)2	0.551	0.791	0.567	0.452	0.117	2.599
PUFA_percEng	0.961	0.079	0.254	0.614	0.824	1.121
dqi_tot	1.006	0.009	0.440	0.507	0.989	1.023
frailty(SiteNo)	NA	NA	0.000	0.921	NA	NA

```
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_pufa_uc_hard,
  variable = "PUFA_percEng"
)
```

Ultra-processed food

This section examines the association between ultra-processed food consumption (percentage of daily energy intake from UPF) and time to flare. UPF is defined according to the NOVA classification system as foods that have undergone substantial industrial processing.

Crohn's disease

Patient-reported flare

```
cox_upf_cd_soft <- coxph(
  Surv(softflare_time, softflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    UPF_perc +
    dqi_tot +
    frailty(SiteNo),
  data = flare.cd.df
```
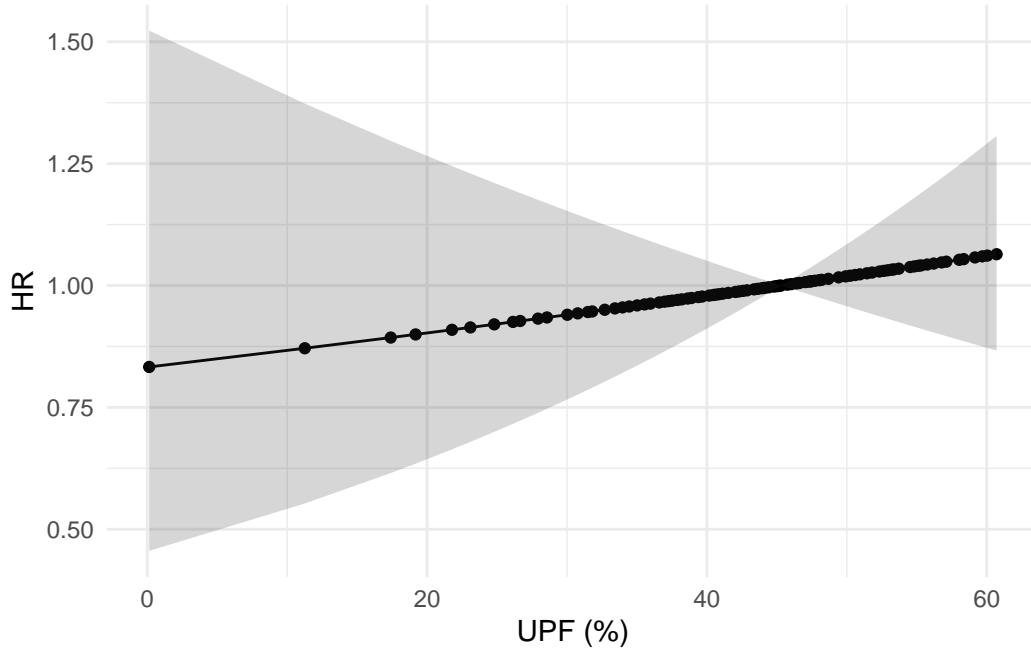
```
)

# Do we need a spline? No
summon_lrt(cox_upf_cd_soft, remove = "UPF_perc", add = "ns(UPF_perc, df = 2)")
```

Test	Removed	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	UPF_perc	ns(UPF_perc, df = 2)	-1012.525	0.861	1	0.354

```
cox_upf_cd_soft %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```

Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.903	0.171	14.097	0.000	1.360	2.663
IMD2	0.703	0.332	1.126	0.289	0.367	1.347
IMD3	0.903	0.306	0.112	0.738	0.496	1.644
IMD4	0.824	0.304	0.405	0.525	0.454	1.495
IMD5	0.956	0.284	0.026	0.873	0.548	1.666
age_decade	1.038	0.055	0.449	0.503	0.931	1.157
BMI	1.006	0.014	0.191	0.662	0.978	1.035
FC	1.270	0.062	15.053	0.000	1.125	1.433
UPF_perc	1.004	0.007	0.353	0.553	0.991	1.018
dqi_tot	1.007	0.007	1.017	0.313	0.993	1.022
frailty(SiteNo)	NA	NA	0.000	0.915	NA	NA

```
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_upf_cd_soft,
  variable = "UPF_perc"
) +
  xlab("UPF (%)")
```

Objective Flare

```r
cox_upf_cd_hard <- coxph(
  Surv(hardflare_time, hardflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    FC +
    UPF_perc +
    dqi_tot +
    frailty(SiteNo),
  data = flare.cd.df
)

# Do we need a spline? No
summon_lrt(cox_upf_cd_hard, remove = "UPF_perc", add = "ns(UPF_perc, df = 2)")
```

Test	Removed	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	UPF_perc	ns(UPF_perc, df = 2)	-591.249	0.634	1.122	0.471

```r
cox_upf_cd_hard %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```
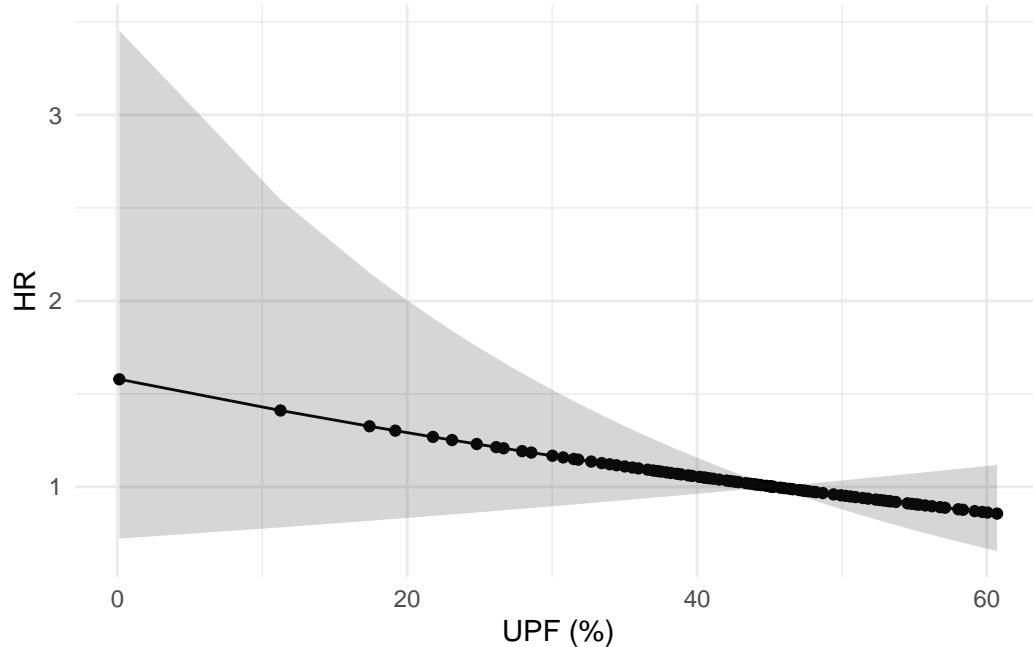
Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.247	0.210	1.102	0.294	0.826	1.884
IMD2	0.651	0.428	1.006	0.316	0.281	1.507
IMD3	0.785	0.404	0.359	0.549	0.356	1.732
IMD4	0.770	0.392	0.444	0.505	0.357	1.660
IMD5	0.746	0.370	0.629	0.428	0.361	1.539
age_decade	0.881	0.070	3.287	0.070	0.767	1.010
BMI	1.027	0.018	2.181	0.140	0.991	1.063
FC	1.394	0.079	17.515	0.000	1.193	1.629
UPF_perc	0.990	0.009	1.305	0.253	0.973	1.007
dqi_tot	1.003	0.009	0.099	0.753	0.985	1.022
frailty(SiteNo)	NA	NA	7.059	0.206	NA	NA

```r
plot_continuous_hr(
  data = flare.cd.df,
  model = cox_upf_cd_hard,
  variable = "UPF_perc"
) +
  xlab("UPF (%)")
```

Ulcerative colitis

Patient-reported flare

```r
cox_upf_uc_soft <- coxph(
  Surv(softflare_time, softflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    ns(FC, df = 2) +
    UPF_perc +
    ns(dqi_tot, df = 2) +
    frailty(SiteNo),
  data = flare.uc.df
)

# Do we need a spline? No
summon_lrt(cox_upf_uc_soft, remove = "UPF_perc", add = "ns(UPF_perc, df = 2)")
```
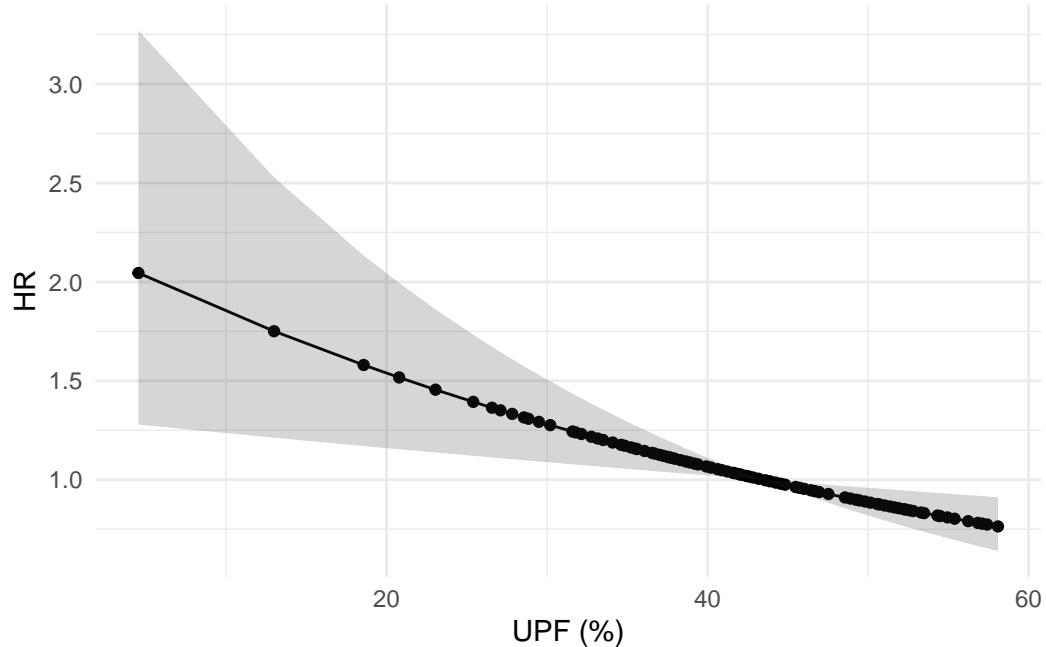
Test	Removed	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	UPF_perc	ns(UPF_perc, df = 2)	-1218.611	0.001	1	0.98

```r
cox_upf_uc_soft %>%
  broom::tidy(exp = TRUE, conf.int = TRUE) %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```

Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.649	0.149	11.284	0.001	1.232	2.208
IMD2	1.305	0.329	0.655	0.418	0.685	2.486
IMD3	1.062	0.323	0.035	0.852	0.564	2.002
IMD4	1.276	0.304	0.645	0.422	0.704	2.315
IMD5	1.114	0.302	0.127	0.721	0.617	2.011
age_decade	0.871	0.049	8.120	0.004	0.791	0.958
BMI	0.988	0.015	0.572	0.449	0.959	1.019
ns(FC, df = 2)1	8.988	0.428	26.291	0.000	3.883	20.807
ns(FC, df = 2)2	0.817	0.580	0.121	0.728	0.262	2.545
UPF_perc	0.982	0.006	8.922	0.003	0.970	0.994
ns(dqi_tot, df = 2)1	1.999	0.703	0.971	0.324	0.504	7.930
ns(dqi_tot, df = 2)2	0.671	0.334	1.430	0.232	0.348	1.291
frailty(SiteNo)	NA	NA	0.000	0.912	NA	NA

```r
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_upf_uc_soft,
  variable = "UPF_perc"
) +
  xlab("UPF (%)")
```

Objective Flare

```
cox_upf_uc_hard <- coxph(
  Surv(hardflare_time, hardflare) ~
    Sex +
    IMD +
    age_decade +
    BMI +
    ns(FC, df = 2) +
    UPF_perc +
    dqi_tot +
    frailty(SiteNo),
  data = flare.uc.df
)

# Do we need a spline? No
summon_lrt(cox_upf_uc_hard, remove = "UPF_perc", add = "ns(UPF_perc, df = 2)")
```

Test	Removed	Added	Log-likelihood	Statistic	DF	p-vaue
LRT	UPF_perc	ns(UPF_perc, df = 2)	-673.452	0.326	1	0.568

```
cox_upf_uc_hard %>%
  broom::tidy(exp = TRUE, conf.int = TRUE)  %>%
  knitr::kable(col.names = broom.cols, digits = 3)
```
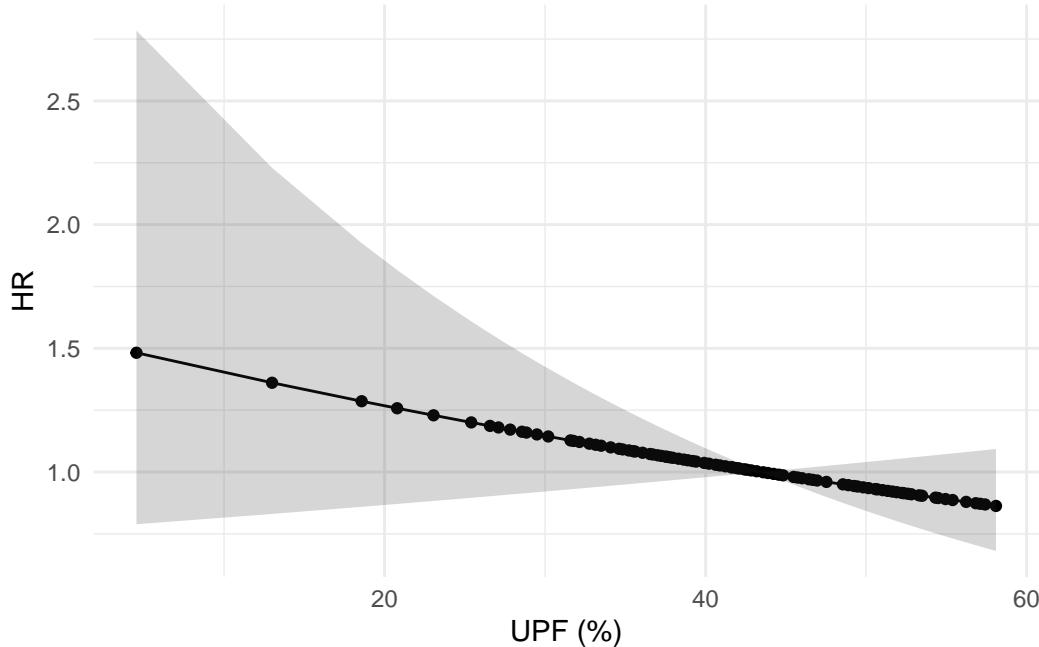
Term	Estimate	Std error	Statistic	p-value	Lower	Upper
SexFemale	1.151	0.193	0.527	0.468	0.788	1.680
IMD2	1.344	0.488	0.367	0.545	0.516	3.501
IMD3	1.199	0.478	0.144	0.705	0.470	3.056
IMD4	2.140	0.442	2.968	0.085	0.901	5.086
IMD5	1.418	0.447	0.610	0.435	0.590	3.403
age_decade	0.808	0.067	10.036	0.002	0.707	0.922
BMI	0.976	0.022	1.258	0.262	0.935	1.018
ns(FC, df = 2)1	14.861	0.585	21.281	0.000	4.722	46.775
ns(FC, df = 2)2	0.542	0.792	0.599	0.439	0.115	2.559
UPF_perc	0.990	0.008	1.495	0.221	0.974	1.006
dqi_tot	1.004	0.009	0.183	0.668	0.986	1.022
frailty(SiteNo)	NA	NA	0.000	0.921	NA	NA

```
plot_continuous_hr(
  data = flare.uc.df,
  model = cox_upf_uc_hard,
  variable = "UPF_perc"
) +
  xlab("UPF (%)")
```

Results

Risk difference

```
# How many bootstraps
nboot <- 199
```

Risk differences quantify the absolute change in cumulative incidence of flare at a specific time point (1 year) associated with different levels of each dietary exposure. These are population-averaged estimates obtained by:

1. Predicting individual-level survival for each participant at specified dietary exposure values
2. Averaging across the population to obtain mean cumulative incidence
3. Calculating differences relative to a reference value (median)
4. Bootstrapping to obtain 95% confidence intervals

Results are presented as forest plots with corresponding risk difference values and confidence intervals.

Crohn's disease

Patient-reported flare

```r
# Values for calculation risk difference
rd_values_meat_cd <- quantile(
  flare.cd.df$Meat_sum,
  probs = c(0, 0.25, 0.5, 0.75, 0.95),
  na.rm = TRUE
  ) %>%
  round()

rd_values_fibre_cd <- quantile(
  flare.cd.df$fibre,
  probs = c(0, 0.25, 0.5, 0.75, 0.95),
  na.rm = TRUE
  ) %>%
  round()

rd_values_pufa_cd <- quantile(
  flare.cd.df$PUFA_percEng,
  probs = c(0, 0.25, 0.5, 0.75, 0.95),
  na.rm = TRUE
  ) %>%
  round(digits = 1)

rd_values_upf_cd <- quantile(
  flare.cd.df$UPF_perc,
  probs = c(0, 0.25, 0.5, 0.75, 0.95),
  na.rm = TRUE
  ) %>%
  round()

# Calculate risk difference for these values
data_rd_cd_soft_meat <- summon_population_risk_difference_boot(
  data = flare.cd.df,
  model = cox_meat_cd_soft,
  times = c(365),
  variable = "Meat_sum",
  values = rd_values_meat_cd,
  ref_value = NULL,
  nboot = nboot
) %>%
```

```
  dplyr::mutate(
    term_tidy = paste0("Total meat protein: ", value)
  )

data_rd_cd_soft_fibre <- summon_population_risk_difference_boot(
  data = flare.cd.df,
  model = cox_fibre_cd_soft,
  times = c(365),
  variable = "fibre",
  values = rd_values_fibre_cd,
  ref_value = NULL,
  nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("Dietary fibre: ", value)
  )

data_rd_cd_soft_pufa <- summon_population_risk_difference_boot(
  data = flare.cd.df,
  model = cox_pufa_cd_soft,
  times = c(365),
  variable = "PUFA_percEng",
  values = rd_values_pufa_cd,
  ref_value = NULL,
  nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("PUFA: ", value)
  )

data_rd_cd_soft_upf <- summon_population_risk_difference_boot(
  data = flare.cd.df,
  model = cox_upf_cd_soft,
  times = c(365),
  variable = "UPF_perc",
  values = rd_values_upf_cd,
  ref_value = NULL,
  nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("UPF (%): ", value)
  )
```

```
# Creating combined forest plot
# 1-year risk difference
plot_rd_cd_soft_meat_1y <- summon_rd_forest_plot(data_rd_cd_soft_meat,
  time = 365
)

plot_rd_cd_soft_fibre_1y <- summon_rd_forest_plot(data_rd_cd_soft_fibre,
  time = 365
)

plot_rd_cd_soft_pufa_1y <- summon_rd_forest_plot(data_rd_cd_soft_pufa,
  time = 365
)

plot_rd_cd_soft_upf_1y <- summon_rd_forest_plot(data_rd_cd_soft_upf,
  time = 365
)
```
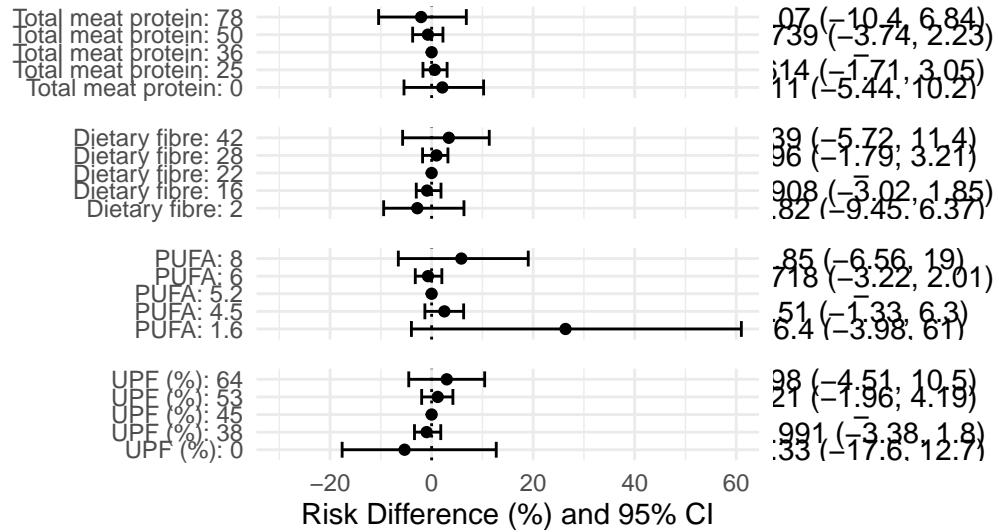
```
# Final plots using patchwork

# 1-year risk difference
plot_rd_cd_soft_meat_1y$plot + plot_rd_cd_soft_meat_1y$rd +
  plot_rd_cd_soft_fibre_1y$plot + plot_rd_cd_soft_fibre_1y$rd +
  plot_rd_cd_soft_pufa_1y$plot + plot_rd_cd_soft_pufa_1y$rd +
  plot_rd_cd_soft_upf_1y$plot + plot_rd_cd_soft_upf_1y$rd +
  patchwork::plot_layout(
    ncol = 2,
    guides = "collect",
    axes = "collect",
    width = c(2, 1)
  ) +
  patchwork::plot_annotation(
    title = "1-year risk difference",
    subtitle = "Crohn's, patient reported flare."
  ) &
  coord_cartesian(xlim = c(-30, 60))
```

1–year risk difference

Crohn's, patient reported flare.


```
Total meat protein: 78          .07 (–10.4, 6.84)
Total meat protein: 50          739 (–3.74, 2.23)
Total meat protein: 36          
Total meat protein: 25          ;14 (–1.71, 3.05)
Total meat protein: 0           11 (–5.44, 10.2)

Dietary fibre: 42               39 (–5.72, 11.4)
Dietary fibre: 28               96 (–1.79, 3.21)
Dietary fibre: 22               
Dietary fibre: 16               908 (–3.02, 1.85)
Dietary fibre: 2                .82 (–9.45, 6.37)

PUFA: 8                         85 (–6.56, 19)
PUFA: 6                         718 (–3.22, 2.01)
PUFA: 5.2                       
PUFA: 4.5                       51 (–1.33, 6.3)
PUFA: 1.6                       6.4 (–3.98, 61)

UPF (%): 64                     98 (–4.51, 10.5)
UPF (%): 53                     21 (–1.96, 4.19)
UPF (%): 45                     
UPF (%): 38                     .991 (–3.38, 1.8)
UPF (%): 0                      .33 (–17.6, 12.7)
```

Risk Difference (%) and 95% CI

(x-axis: –20, 0, 20, 40, 60)

Objective Flare

```r
# Calculate risk difference for these values
data_rd_cd_hard_meat <- summon_population_risk_difference_boot(
  data = flare.cd.df,
  model = cox_meat_cd_hard,
  times = c(365),
  variable = "Meat_sum",
  values = rd_values_meat_cd,
  ref_value = NULL,
  nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("Total meat protein: ", value)
  )

data_rd_cd_hard_fibre <- summon_population_risk_difference_boot(
  data = flare.cd.df,
  model = cox_fibre_cd_hard,
  times = c(365),
  variable = "fibre",
  values = rd_values_fibre_cd,
```

71

```r
  ref_value = NULL,
  nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("Dietary fibre: ", value)
  )

data_rd_cd_hard_pufa <- summon_population_risk_difference_boot(
  data = flare.cd.df,
  model = cox_pufa_cd_hard,
  times = c(365),
  variable = "PUFA_percEng",
  values = rd_values_pufa_cd,
  ref_value = NULL,
  nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("PUFA: ", value)
  )

data_rd_cd_hard_upf <- summon_population_risk_difference_boot(
  data = flare.cd.df,
  model = cox_upf_cd_hard,
  times = c(365),
  variable = "UPF_perc",
  values = rd_values_upf_cd,
  ref_value = NULL,
  nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("UPF (%): ", value)
  )


# Creating combined forest plot
# 1-year risk difference
plot_rd_cd_hard_meat_1y <- summon_rd_forest_plot(data_rd_cd_hard_meat,
  time = 365
)

plot_rd_cd_hard_fibre_1y <- summon_rd_forest_plot(data_rd_cd_hard_fibre,
  time = 365
```

```
)

plot_rd_cd_hard_pufa_1y <- summon_rd_forest_plot(data_rd_cd_hard_pufa,
  time = 365
)

plot_rd_cd_hard_upf_1y <- summon_rd_forest_plot(data_rd_cd_hard_upf,
  time = 365
)
```
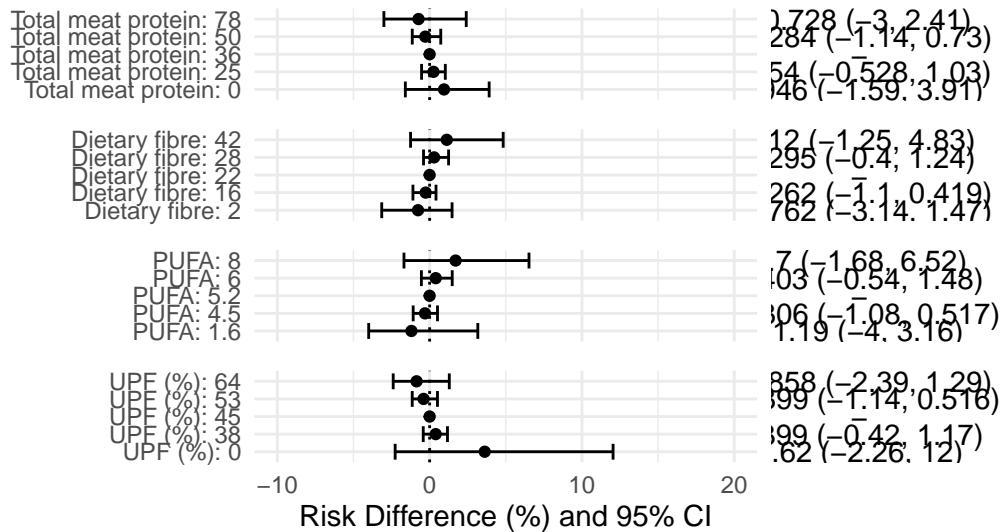
```
# Final plots using patchwork

# 1-year risk difference
plot_rd_cd_hard_meat_1y$plot + plot_rd_cd_hard_meat_1y$rd +
  plot_rd_cd_hard_fibre_1y$plot + plot_rd_cd_hard_fibre_1y$rd +
  plot_rd_cd_hard_pufa_1y$plot + plot_rd_cd_hard_pufa_1y$rd +
  plot_rd_cd_hard_upf_1y$plot + plot_rd_cd_hard_upf_1y$rd +
  patchwork::plot_layout(
    ncol = 2,
    guides = "collect",
    axes = "collect",
    width = c(2, 1)
  ) +
  patchwork::plot_annotation(
    title = "1-year risk difference",
    subtitle = "Crohn's, objective flare."
  ) &
  coord_cartesian(xlim = c(-10, 20))
```

1–year risk difference

Crohn's, objective flare.

		Risk Difference
Total meat protein: 78		0.728 (−3, 2.41)
Total meat protein: 50		0.284 (−1.14, 0.73)
Total meat protein: 36		0.54 (−0.528, 1.03)
Total meat protein: 25		
Total meat protein: 0		1.46 (−1.59, 3.91)
Dietary fibre: 42		1.12 (−1.25, 4.83)
Dietary fibre: 28		0.295 (−0.4, 1.24)
Dietary fibre: 22		
Dietary fibre: 16		0.262 (−1.1, 0.419)
Dietary fibre: 2		0.762 (−3.14, 1.47)
PUFA: 8		2.7 (−1.68, 6.52)
PUFA: 6		0.403 (−0.54, 1.48)
PUFA: 5.2		
PUFA: 4.5		0.306 (−1.08, 0.517)
PUFA: 1.6		1.19 (−4, 3.16)
UPF (%): 64		0.858 (−2.39, 1.29)
UPF (%): 53		0.599 (−1.14, 0.516)
UPF (%): 45		
UPF (%): 38		0.399 (−0.42, 1.17)
UPF (%): 0		3.62 (−2.26, 12)

Risk Difference (%) and 95% CI

Ulcerative colitis

Patient-reported flare

```
# Values for calculation risk difference
rd_values_meat_uc <- quantile(flare.uc.df$Meat_sum,
  probs = c(0, 0.25, 0.5, 0.75, 0.95),
  na.rm = TRUE
) %>%
  round()

rd_values_fibre_uc <- quantile(flare.uc.df$fibre,
  probs = c(0, 0.25, 0.5, 0.75, 0.95),
  na.rm = TRUE
) %>%
  round()

rd_values_pufa_uc <- quantile(flare.uc.df$PUFA_percEng,
  probs = c(0, 0.25, 0.5, 0.75, 0.95),
  na.rm = TRUE
) %>%
  round(digits = 1)
```

```r
rd_values_upf_uc <- quantile(flare.uc.df$UPF_perc,
  probs = c(0, 0.25, 0.5, 0.75, 0.95),
  na.rm = TRUE
) %>%
  round()

# Calculate risk difference for these values
data_rd_uc_soft_meat <- summon_population_risk_difference_boot(
  data = flare.uc.df,
  model = cox_meat_uc_soft,
  times = c(365),
  variable = "Meat_sum",
  values = rd_values_meat_uc,
  ref_value = NULL,
  nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("Total meat protein: ", value)
  )

data_rd_uc_soft_fibre <- summon_population_risk_difference_boot(
  data = flare.uc.df,
  model = cox_fibre_uc_soft,
  times = c(365),
  variable = "fibre",
  values = rd_values_fibre_uc,
  ref_value = NULL,
  nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("Dietary fibre: ", value)
  )

data_rd_uc_soft_pufa <- summon_population_risk_difference_boot(
  data = flare.uc.df,
  model = cox_pufa_uc_soft,
  times = c(365),
  variable = "PUFA_percEng",
  values = rd_values_pufa_uc,
  ref_value = NULL,
  nboot = nboot
) %>%
```

```r
  dplyr::mutate(
    term_tidy = paste0("PUFA: ", value)
  )


data_rd_uc_soft_upf <- summon_population_risk_difference_boot(
  data = flare.uc.df,
  model = cox_upf_uc_soft,
  times = c(365),
  variable = "UPF_perc",
  values = rd_values_upf_uc,
  ref_value = NULL,
  nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("UPF (%): ", value)
  )


# Creating combined forest plot
# 1-year risk difference
plot_rd_uc_soft_meat_1y <- summon_rd_forest_plot(data_rd_uc_soft_meat,
  time = 365
)


plot_rd_uc_soft_fibre_1y <- summon_rd_forest_plot(data_rd_uc_soft_fibre,
  time = 365
)


plot_rd_uc_soft_pufa_1y <- summon_rd_forest_plot(data_rd_uc_soft_pufa,
  time = 365
)


plot_rd_uc_soft_upf_1y <- summon_rd_forest_plot(data_rd_uc_soft_upf,
  time = 365
)
```

```r
# Final plots using patchwork

# 1-year risk difference
plot_rd_uc_soft_meat_1y$plot + plot_rd_uc_soft_meat_1y$rd +
  plot_rd_uc_soft_fibre_1y$plot + plot_rd_uc_soft_fibre_1y$rd +
  plot_rd_uc_soft_pufa_1y$plot + plot_rd_uc_soft_pufa_1y$rd +
  plot_rd_uc_soft_upf_1y$plot + plot_rd_uc_soft_upf_1y$rd +
```
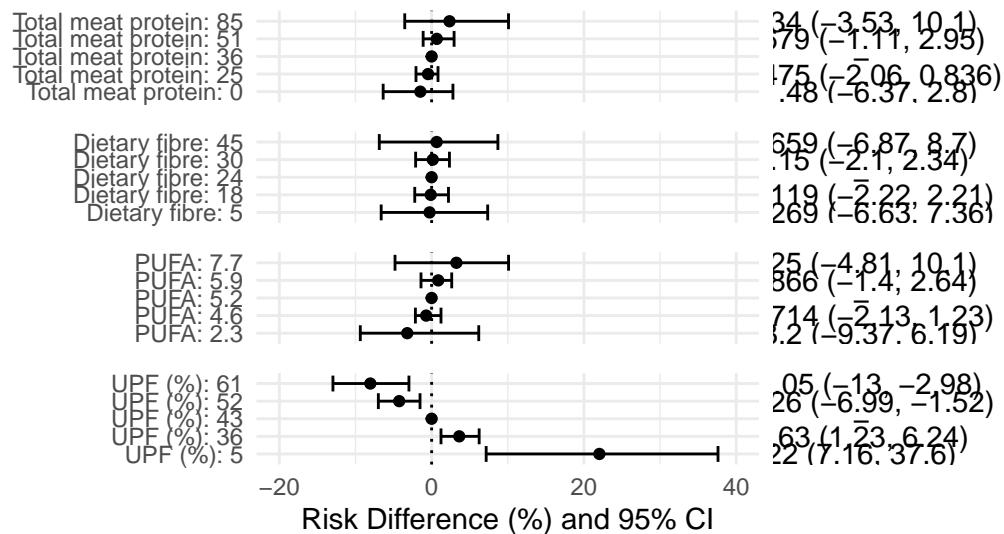
```
patchwork::plot_layout(
  ncol = 2,
  guides = "collect",
  axes = "collect",
  width = c(2, 1)
) +
patchwork::plot_annotation(
  title = "1-year risk difference",
  subtitle = "Ulcerative colitis, patient reported flare."
) &
coord_cartesian(xlim = c(-20, 40))
```

1–year risk difference
Ulcerative colitis, patient reported flare.

Objective Flare

```
# Calculate risk difference for these values
data_rd_uc_hard_meat <- summon_population_risk_difference_boot(
  data = flare.uc.df,
  model = cox_meat_uc_hard,
  times = c(365),
  variable = "Meat_sum",
  values = rd_values_meat_uc,
```

```r
    ref_value = NULL,
    nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("Total meat protein: ", value)
  )

data_rd_uc_hard_fibre <- summon_population_risk_difference_boot(
  data = flare.uc.df,
  model = cox_fibre_uc_hard,
  times = c(365),
  variable = "fibre",
  values = rd_values_fibre_uc,
  ref_value = NULL,
  nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("Dietary fibre: ", value)
  )

data_rd_uc_hard_pufa <- summon_population_risk_difference_boot(
  data = flare.uc.df,
  model = cox_pufa_uc_hard,
  times = c(365),
  variable = "PUFA_percEng",
  values = rd_values_pufa_uc,
  ref_value = NULL,
  nboot = nboot
) %>%
  dplyr::mutate(
    term_tidy = paste0("PUFA: ", value)
  )

data_rd_uc_hard_upf <- summon_population_risk_difference_boot(
  data = flare.uc.df,
  model = cox_upf_uc_hard,
  times = c(365),
  variable = "UPF_perc",
  values = rd_values_upf_uc,
  ref_value = NULL,
  nboot = nboot
) %>%
```

```r
  dplyr::mutate(
    term_tidy = paste0("UPF (%): ", value)
  )


# Creating combined forest plot
# 1-year risk difference
plot_rd_uc_hard_meat_1y <- summon_rd_forest_plot(data_rd_uc_hard_meat,
  time = 365
)

plot_rd_uc_hard_fibre_1y <- summon_rd_forest_plot(data_rd_uc_hard_fibre,
  time = 365
)

plot_rd_uc_hard_pufa_1y <- summon_rd_forest_plot(data_rd_uc_hard_pufa,
  time = 365
)

plot_rd_uc_hard_upf_1y <- summon_rd_forest_plot(data_rd_uc_hard_upf,
  time = 365
)
```

```r
# Final plots using patchwork

# 1-year risk difference
plot_rd_uc_hard_meat_1y$plot + plot_rd_uc_hard_meat_1y$rd +
  plot_rd_uc_hard_fibre_1y$plot + plot_rd_uc_hard_fibre_1y$rd +
  plot_rd_uc_hard_pufa_1y$plot + plot_rd_uc_hard_pufa_1y$rd +
  plot_rd_uc_hard_upf_1y$plot + plot_rd_uc_hard_upf_1y$rd +
  patchwork::plot_layout(
    ncol = 2,
    guides = "collect",
    axes = "collect",
    width = c(2, 1)
  ) +
  patchwork::plot_annotation(
    title = "1-year risk difference",
    subtitle = "Ulcerative colitis, objective flare."
  ) &
  coord_cartesian(xlim = c(-10, 20))
```

1–year risk difference

Ulcerative colitis, objective flare.

	Risk Difference (%) and 95% CI
Total meat protein: 85	5.15 (1.4, 9.74)
Total meat protein: 51	1.3 (0.407, 2.27)
Total meat protein: 36	
Total meat protein: 25	0.22 (−1.35, −0.288)
Total meat protein: 0	5 (−3.83, −0.909)
Dietary fibre: 45	35 (−2.07, 4.13)
Dietary fibre: 30	24 (−0.664, 1.03)
Dietary fibre: 24	
Dietary fibre: 18	01 (−0.921, 0.729)
Dietary fibre: 5	248 (−2.59, 2.56)
PUFA: 7.7	702 (−3.22, 2.44)
PUFA: 5.9	33 (−1.08, 0.628)
PUFA: 5.2	
PUFA: 4.6	24 (−0.506, 1.05)
PUFA: 2.3	32 (−2.2, 6.49)
UPF (%): 61	33 (−3.27, 0.603)
UPF (%): 52	11 (−1.83, 0.296)
UPF (%): 43	
UPF (%): 36	29 (−0.224, 1.69)
UPF (%): 5	42 (−1.15, 13.3)

Risk Difference (%) and 95% CI