STA5507 - Final Project

AUTHOR
Joel Carroll, Elise Dixon, Paul Hill

Disclaimer: We didn't use AI tools at all

Project Guidelines

Data set

The dataset `df_who` was derived from the World Health Organization (WHO) Life Expectancy dataset, focusing on health indicators and socioeconomic factors that potentially influence life expectancy in developing countries. After data cleaning, transformation, and filtering, 'df_who' contains a random sample of size 500 from the original dataset, with each observation representing a year-country combination from the subset of "Developing" countries. Although the sample unit in the original dataset is "country," for the purposes of this analysis, we assume that the 500 observations are independent.

Variables in `df_who`:

- **life** y: Life expectancy at birth (in years) - the primary response variable in the analysis, reflecting the average number of years a newborn is expected to live given current health and socioeconomic conditions.
- **GDP** x_1: Gross Domestic Product per capita (in USD) - a measure of a country's economic output per person, indicating overall economic well-being. The variable is represented in its original form, with higher values typically indicating greater economic resources.
- **school** x_2: Average years of schooling - an indicator of educational attainment within a country, with more years of schooling generally associated with better health outcomes.
- **hiv** x_3: HIV/AIDS prevalence (per 1,000 people) - an indicator of the impact of HIV/AIDS within the population. This variable is provided in its raw form, representing the number of cases per 1,000 people.
- **year** x_4: Year (ordered categorical variable) - represents the year the data was collected, enabling temporal trend analysis. This is treated as an ordered factor to respect the time sequence.
- **cat_thinness** x_5: Thinness category (ordered categorical variable) - created based on the "thinness (5-9 years)" indicator. It has two levels, grouping countries into thinness categories defined by specific breaks (0-10 and 10-30). The thinness variable refers to the prevalence (i.e., percentage) of children between 5 and 9 years old with low Body Mass Index (BMI). The thinness variable is typically used to assess malnutrition levels across populations.

Nonparametric regression models

The aim of this final project is to fit nonparametric regression models with life expectancy as the response variable and a selection of covariates that capture key health and socioeconomic factors. The covariates include the logarithm of GDP, average years of schooling, log of HIV/AIDS prevalence, year (treated as an ordered categorical variable), and thinness category. Through these models, we seek to explore and analyze the relationships between life expectancy and the chosen covariates in developing countries.
Consider the following models:

1. Nadaraya-Watson estimator with asymptotic confidence bands.
2. Local constant regression with naive bootstrap confidence bands.
3. Local linear regression with wild bootstrap confidence bands.
4. Generalized additive model (GAM) without including interactions.
5. Bayesian additive regression trees (BART).

Data analysis

In this project, we want to make inferences about

$$E(Y|X_1 = x_1, X_2 = x_2, X_3 = x_3, X_4 = x_4, X_5 = x_5)$$

and for each of the models, you need to provide an answer or a solution for the following items:

1. Comment on the underlying modeling assumptions and whether or not we can check some of them. Check those that can be verified.
2. Using the model outputs, determine how each of the these five predictors relates to the response. Interpret the results in the context of the problem.
3. Choose a combination for $(x_1, x_2, x_3, x_4, x_5)$, and provide a point and interval estimate of

   $$E(Y|X_1 = x_1, X_2 = x_2, X_3 = x_3, X_4 = x_4, X_5 = x_5).$$

   Interpret the result in the context of the problem.
4. Choose a combination for $(x_1, x_2, x_3, x_4, x_5)$ and a value for Δ_3, and provide a point and interval estimate of

   $$E(Y|X_1 = x_1, X_2 = x_2, X_3 = x_3 + \Delta_3, X_4 = x_4, X_5 = x_5)$$

   and

   $$E(Y|X_1 = x_1, X_2 = x_2, X_3 = x_3, X_4 = x_4, X_5 = x_5).$$

Provide an interpretation in terms of how the expected value changes when x_3 is increased by Δ_3 units.

5. Choose a combination for (x_1, x_3, x_4), create a plot that displays a point and interval estimate of

$$x_2 \mapsto E(Y|X_1 = x_1, X_2 = x_2, X_3, X_4 = x_4, X_5 \in [0, 10])$$

and

$$x_2 \mapsto E(Y|X_1 = x_1, X_2 = x_2, X_3, X_4 = x_4, X_5 \in (10, 30]).$$

Interpret the result in the context of the problem.

Hint: When choosing specific values for $(x_1, x_2, x_3, x_4, x_5)$, check that they are not far away from the observed multivariate range.

```
#Require packages
library(np)
library(MVN)
library(nortest)
library(mgcv)
```

```
Warning: package 'mgcv' was built under R version 4.4.2
```

```
library(ggplot2)

#Load and prep data
df_who = read.csv("df_who.csv")[,-1]
df_who[,5:6] = lapply(df_who[,5:6], as.ordered)   #Order year, thinness
df_who[, c(2,4)] = lapply(df_who[,c(2,4)], log)   #Log of GDP, hiv
colnames(df_who)[c(2,4)] <- c("log_GDP", "log_hiv")

#Function to plot local constant/local linear regression model outputs
model_output <- function(bw){

  #Fix quantitative predictors to their medians, and categorical            predictors to the mos
  x = c(median(df_who$log_GDP), median(df_who$school), median(df_who$log_hiv),
        names(sort(table(df_who$year), decreasing = T))[1],
        names(sort(table(df_who$cat_thinness), decreasing = T))[1])

  #Plot mhat vs x[i], i = 1, ..., 5, with other x's fixed
  for(i in 2:6){

    #For quantitative x[i], plot a curve
    if(class(df_who[,i])[1] == "numeric"){
      n = 100

      #Create dataframe of points to evaluate mhat
      x_seq = seq(min(df_who[,i]), max(df_who[,i]),
                  length.out = n)
      eval_pts = matrix(c(rep(NA, n), rep(x[1], n), rep(x[2], n),
```

```
                        rep(x[3], n), rep(x[4], n), rep(x[5], n)),
                      nrow = n)
      colnames(eval_pts) = names(df_who)
      eval_pts[,i] = x_seq
      eval_pts = as.data.frame(eval_pts)
      eval_pts[,2:4] = lapply(eval_pts[,2:4], as.numeric)
      eval_pts[,5:6] = lapply(eval_pts[,5:6], as.ordered)

      #Evaluate mhats
      mhat = npreg(bw, exdat = eval_pts[,-1])
      eval_pts[,1] = mhat$mean

      #Plot
      xlab = names(df_who)[i]
      title = paste("Average Life Expectancy vs", xlab)
      plot(eval_pts[,i], eval_pts$life, xlab = xlab, ylab = "life",
           main = title, type = "l")

    }else{

      #For categorical x[i], plot points
      lvls = levels(df_who[,i])
      n = length(lvls)
      eval_pts = matrix(c(rep(NA, n), rep(x[1], n), rep(x[2], n),
                        rep(x[3], n), rep(x[4], n), rep(x[5], n)),
                      nrow = n)
      colnames(eval_pts) = names(df_who)
      eval_pts[,i] = lvls
      eval_pts = as.data.frame(eval_pts)
      eval_pts[,2:4] = lapply(eval_pts[,2:4], as.numeric)
      eval_pts[,5:6] = lapply(eval_pts[,5:6], as.ordered)

      #Evaluate points
      mhat = npreg(bw, exdat = eval_pts[,-1])
      eval_pts[,1] = mhat$mean

      #Plot
      xlab = names(df_who)[i]
      title = paste("Average Life Expectancy vs", xlab)
      plot(eval_pts[,i], eval_pts$life, xlab = xlab, ylab = "life",
           main = title, type = "p")
    }
  }
}
```

Model 1: Nadaraya-Watson estimator with asymptotic confidence bands

A popular method of nonparametric regression is kernel regression, which utilizes the fact that the relationship between independent predictors $\vec{X} = (X_1, \ldots, X_p)$ and quantitative response Y can be explained by their joint cumulative distribution function F. Contrary to parametric regression, the general form of a nonparametric regression model

$$Y = m(\vec{X}) + \epsilon$$

incorporates a functional component into the parameter space.

Specifically, the conditional probability density function of $f_{Y|\vec{X}=\vec{x}}(y)$ gives us the exact behavior of Y when $\vec{X} = \vec{x} := (x_1, \ldots, x_p)$. This conditional pdf, however, is particularly difficult to estimate for given values of each predictor variable. To simplify the problem, we can instead estimate the conditional mean of Y given specific values of $\vec{X}$, which is a function of $\vec{x}$ defined as

$$m(\vec{x}) := E[Y|\vec{X} = \vec{x}] = \int_{\mathbb{R}} y f_{Y|\vec{X}=\vec{x}}(y)dy = \frac{\int_{\mathbb{R}} y f(\vec{x}, y)dy}{f_{\vec{X}}(\vec{x})}$$

where f is the joint pdf/pmf (depending on whether any of the predictor variables are categorical) of the predictors and the response, and $f_{\vec{X}}$ is the marginal probability function of $\vec{X}$. We can estimate m by replacing f and $f_{\vec{X}}$ with their respective kernel estimates (with given kernels and bandwidths), yielding (after some simplification)

$$\hat{m}(\vec{x}; H) = \sum_{i=1}^{n} \frac{K_H(\vec{x} - \vec{X}_i)}{\sum_{j=1}^{n} K_H(\vec{x} - \vec{X}_j)} Y_i = \sum_{i=1}^{n} W(\vec{X}_i) Y_i$$

where K_H is the joint kernel estimate of $\vec{X}$ with bandwidth matrix H and

$$W(\vec{X}_i) = \frac{K_H(\vec{x} - \vec{X}_i)}{\sum_{j=1}^{n} K_H(\vec{x} - \vec{X}_j)}.$$

This particular estimate of the conditional mean of Y given the predictors is known as the Nadaraya-Watson estimator. We further simplify the problem of estimation by utilizing product kernels so that $K_H(\vec{x} - \vec{X}_i) = \prod_{k=1}^{p} K_{h_k}(x_k - X_{i,k})$ and $H = \text{diag}(h_1^2, \ldots, h_p^2)$. For quantitative predictors X_q, we use the Gaussian kernel, and for ordinal predictors X_d, we use the ordered discrete kernel $I_o(x_d - X_d; h_d) := h_d^{|x_d - X_d|}$. Furthermore, we find the optimal bandwidths $h_1, \ldots, h_p$ that minimize the least-squares cross-validation function

$$CV(h_1, \ldots, h_p) = \frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{m}_{-i}(\vec{X}_i; H))^2$$

where $\hat{m}_{-i}(\vec{x}; H)$ is the Nadaraya-Watson estimator of $m(\vec{x})$ using all the data except for $\vec{X}_i$.

Under certain regularity assumptions, the asymptotic distribution of $\hat{m}(\vec{x}; H)$ as $nh \to \infty$ is normal with mean and variance

$$E[\hat{m}(\vec{x};H)] = m(\vec{x}) + \sum_{k=1}^{p}\left\{h_k\sigma^2_{K_{h_k}}\left(\frac{\partial^2 m}{\partial x_k^2} + \frac{2\frac{\partial m}{\partial x_k}\frac{\partial f_{\vec{X}}}{\partial x_k}}{f_{\vec{X}}(\vec{x})}\right) + o_p(h_k)\right\},$$

$$\mathrm{Var}(\hat{m}(\vec{x};H)) = \frac{\sigma^2(\vec{x})\prod_{k=1}^{p}R(K_{h_k})}{n\sqrt{|H|}f(\vec{x})} + o_p\left(\left(n\sqrt{|H|}\right)^{-1}\right),$$

respectively. Note that if $h_1, \ldots, h_p$ are small and m is smooth (over the continuous predictors), then the Nadaraya-Watson estimator is close to unbiased. Thus, we can construct a confidence interval for m with standard error

$$\hat{s}(\vec{x}) = \sqrt{\frac{\hat{\sigma}^2(\vec{x})\prod_{k=1}^{p}R(K_{h_k})}{n\sqrt{|H|}\hat{f}_{\vec{X}}(\vec{x};H)}}$$

where $\hat{f}_{\vec{X}}(\vec{x};H)$ is the kernel density estimate of $f_{\vec{X}}(\vec{x})$ and $\sigma^2(x_k)$ is estimated by

$$\hat{\sigma}^2(\vec{x}) = \frac{\sum_{i=1}^{n}K_H(\vec{x}-\vec{X}_i)(Y_i-\hat{m}(\vec{x};H))^2}{\sum_{i=1}^{n}K_{h_k}(x_k - X_{i,k})}$$

to avoid assuming conditional homoscedicity of the residuals. Ultimately, the $(1-\alpha/2)$ confidence interval for m is given by $\hat{m}(\vec{x};H) \pm z_{1-\alpha/2}\hat{s}(x)$.

Assumptions

The assumptions for a nonparametric model using the Nadaraya-Watson estimator are:

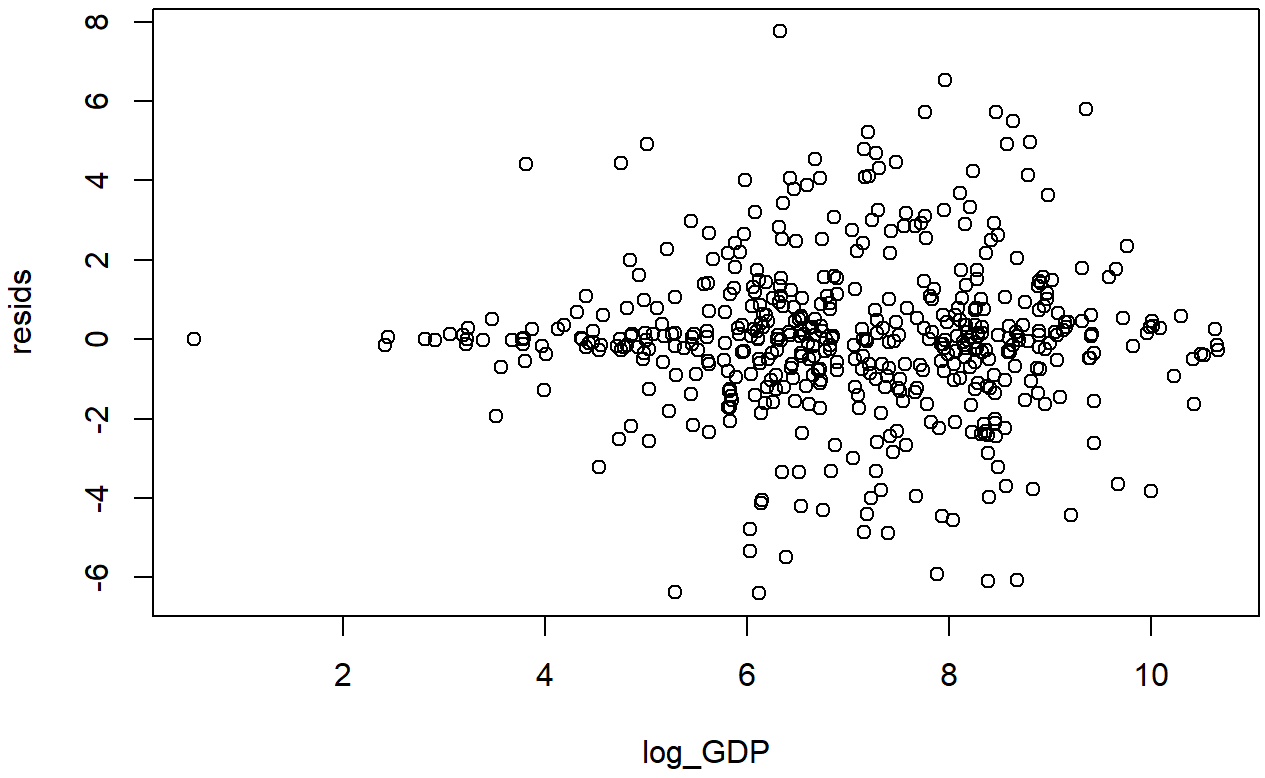
1. Around point $\vec{X}$ the mean of Y can be locally approximated by a constant
2. Errors in estimating Y are independent and identically Normally distributed with mean 0 and constant variance
3. The unknown function m is twice differentiable with respect to the quantitative predictors
4. $|H|$ is small while $n|H|$ is large

Assumption (1) is assumed by the premise of the model, and assumption (3) cannot be verified since m is unknown. Computed bandwidths are relatively small and n is large, reasonably satisfying assumption (4), and requirements of assumption (2) are investigated below.

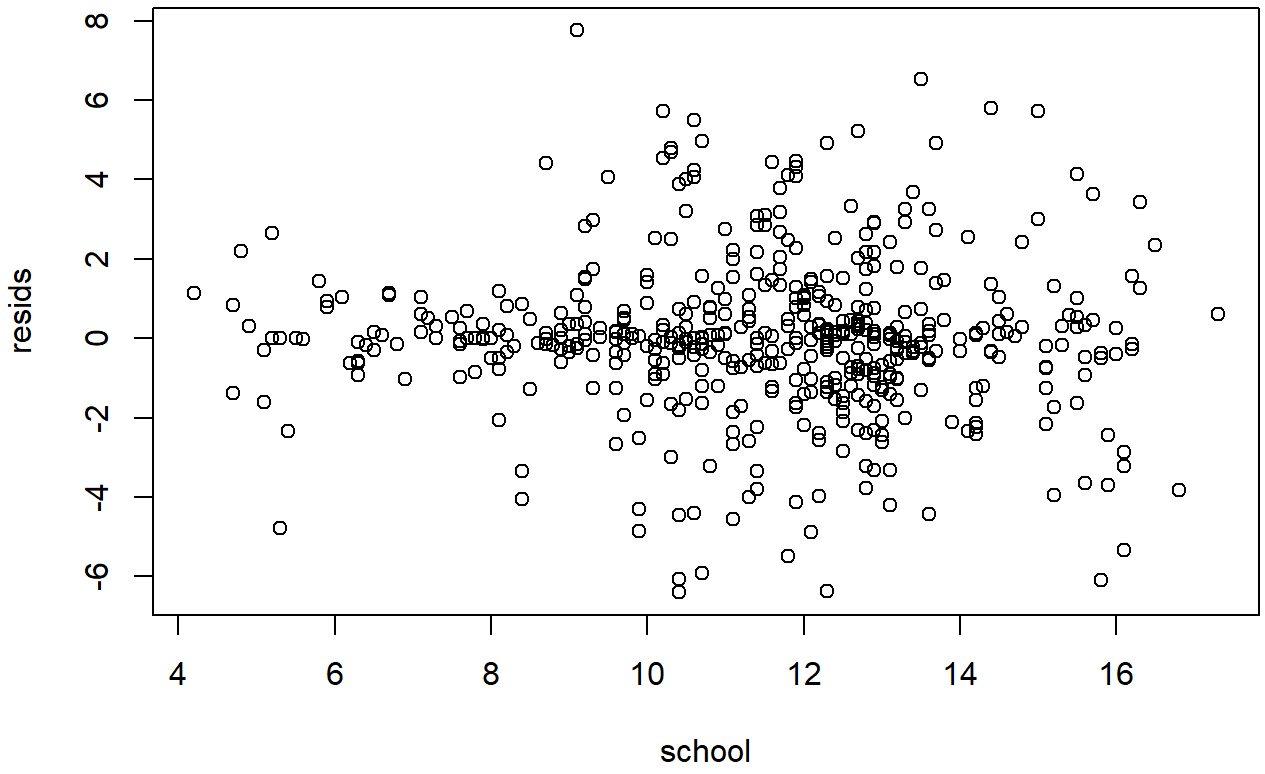
```
#Compute least-squares cross-validation bandwidth for NW estimator
bw = npregbw(formula = life ~ log_GDP + school + log_hiv + year + cat_thinness, data = df_who)
```

```
#Check residuals are normal with constant variance
attach(df_who)

mhat = npreg(bw)$mean
```

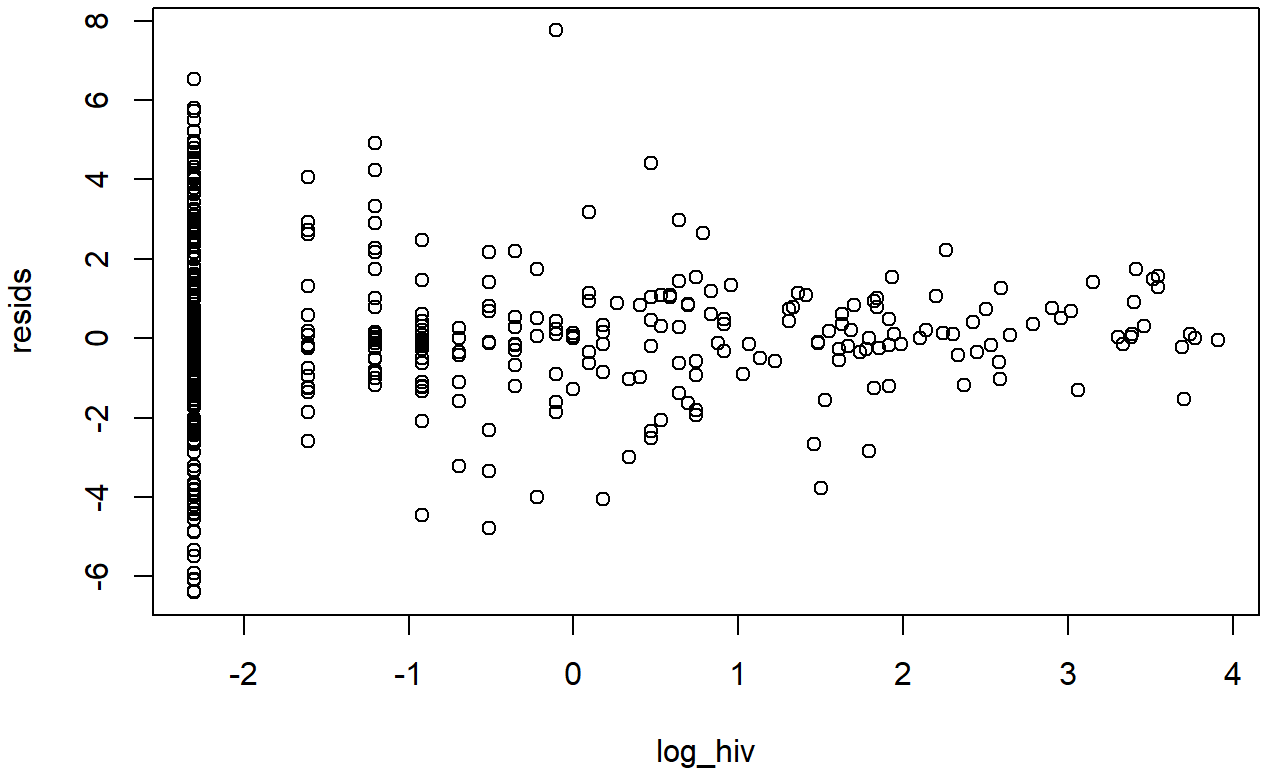
```r
resids = mhat - life

#Check residual plots
plot(log_GDP, resids)
```



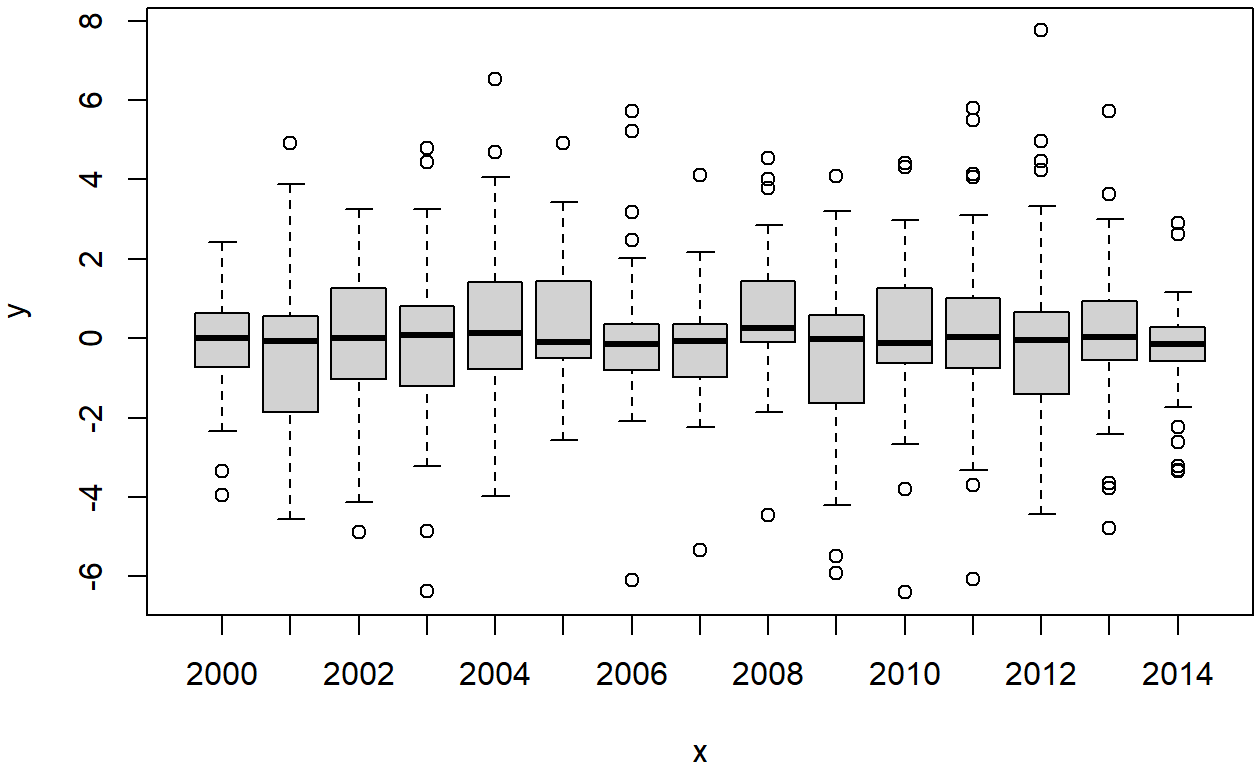

```r
plot(school, resids)
```



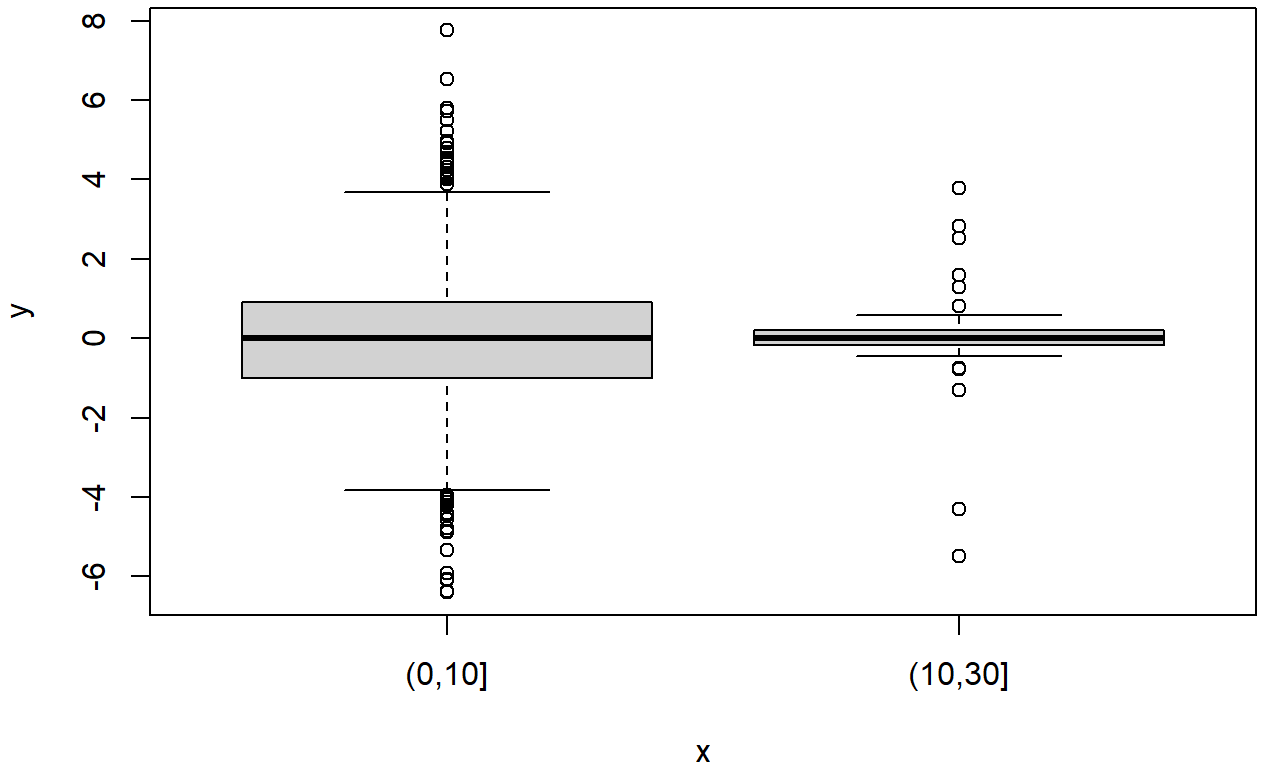

```
plot(log_hiv, resids)
```




```
plot(year, resids)
```




```
plot(cat_thinness, resids)
```




```
#Check if residuals are multivariate normal given values of quantitative predictors
MVN::mvn(data = cbind(resids, df_who[,2:4]))$multivariateNormality
```

```
           Test       HZ p value MVN
1 Henze-Zirkler 6.511403       0  NO
```

```
#Check if residuals are multivariate normal given levels of categorical predictors
for(i in levels(year)){
  LF = lillie.test(resids[which(year==i)])
  print(paste("Lilliefors test of residuals for year", i, "p-value =",
              LF$p.value))
}
```

```
[1] "Lilliefors test of residuals for year 2000 p-value = 0.212914443063246"
[1] "Lilliefors test of residuals for year 2001 p-value = 0.432467692154963"
[1] "Lilliefors test of residuals for year 2002 p-value = 0.124076101276831"
[1] "Lilliefors test of residuals for year 2003 p-value = 0.245145510323155"
[1] "Lilliefors test of residuals for year 2004 p-value = 0.0178585760693962"
[1] "Lilliefors test of residuals for year 2005 p-value = 0.00185579497061089"
[1] "Lilliefors test of residuals for year 2006 p-value = 0.00044043428718303"
[1] "Lilliefors test of residuals for year 2007 p-value = 0.117046573335022"
[1] "Lilliefors test of residuals for year 2008 p-value = 0.0372272729709904"
[1] "Lilliefors test of residuals for year 2009 p-value = 0.00111773334501625"
```

```
[1] "Lilliefors test of residuals for year 2010 p-value = 0.224350772392553"
[1] "Lilliefors test of residuals for year 2011 p-value = 0.0282175279350592"
[1] "Lilliefors test of residuals for year 2012 p-value = 0.00386431707650081"
[1] "Lilliefors test of residuals for year 2013 p-value = 0.0204726629821187"
[1] "Lilliefors test of residuals for year 2014 p-value = 0.00192917496618653"
```

```
for(i in levels(cat_thinness)){
  LF = lillie.test(resids[which(cat_thinness==i)])
  print(paste("Lilliefors test of residuals for cat_thinness", i,
              "p-value =", LF$p.value))
}
```

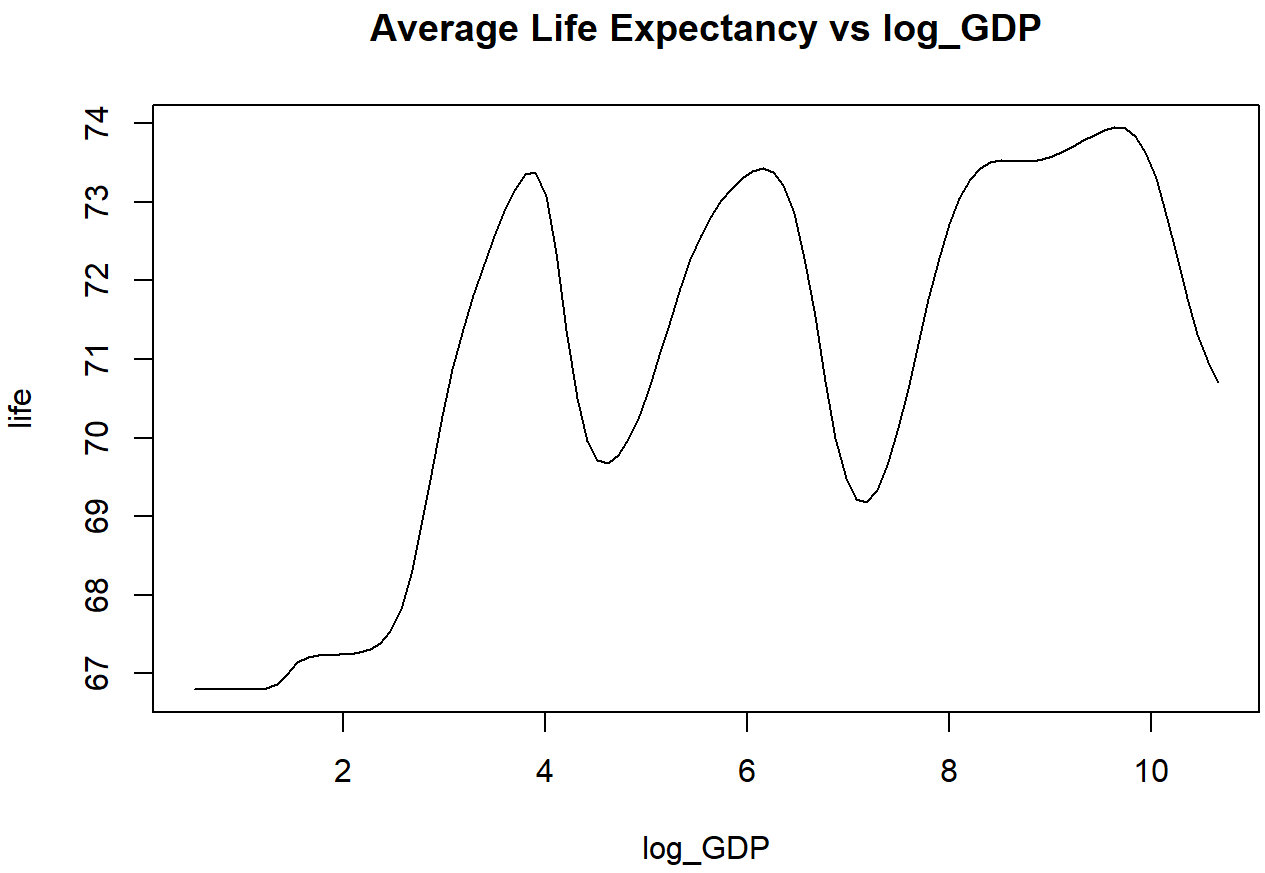
```
[1] "Lilliefors test of residuals for cat_thinness (0,10] p-value = 3.30443354504891e-09"
[1] "Lilliefors test of residuals for cat_thinness (10,30] p-value = 2.300266788819e-10"
```

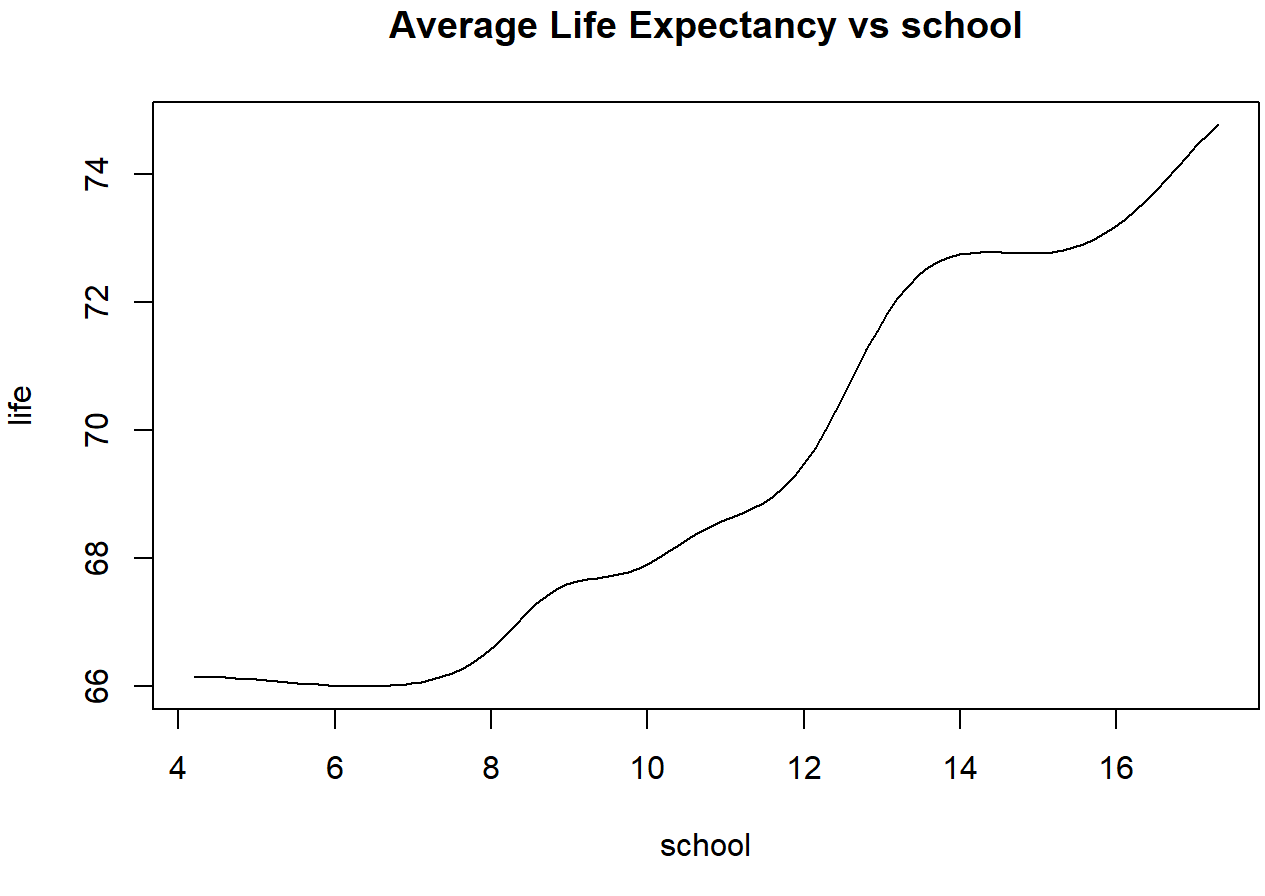
```
detach(df_who)
```

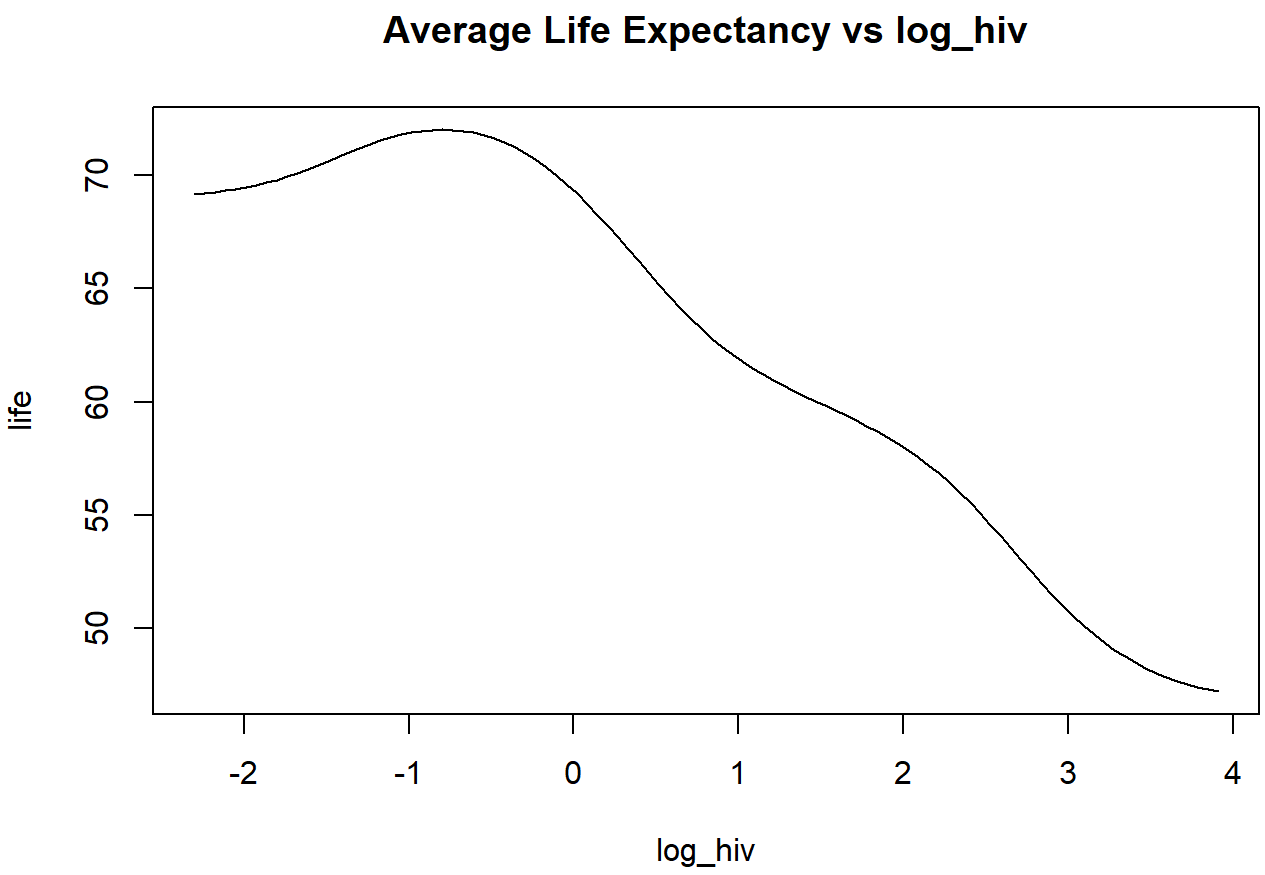
Clearly the verifiable assumptions are not met. Results from the Lilliefors tests suggest the predictors are not normal, and residual plots show nonconstant variance to some degree for all predictors.

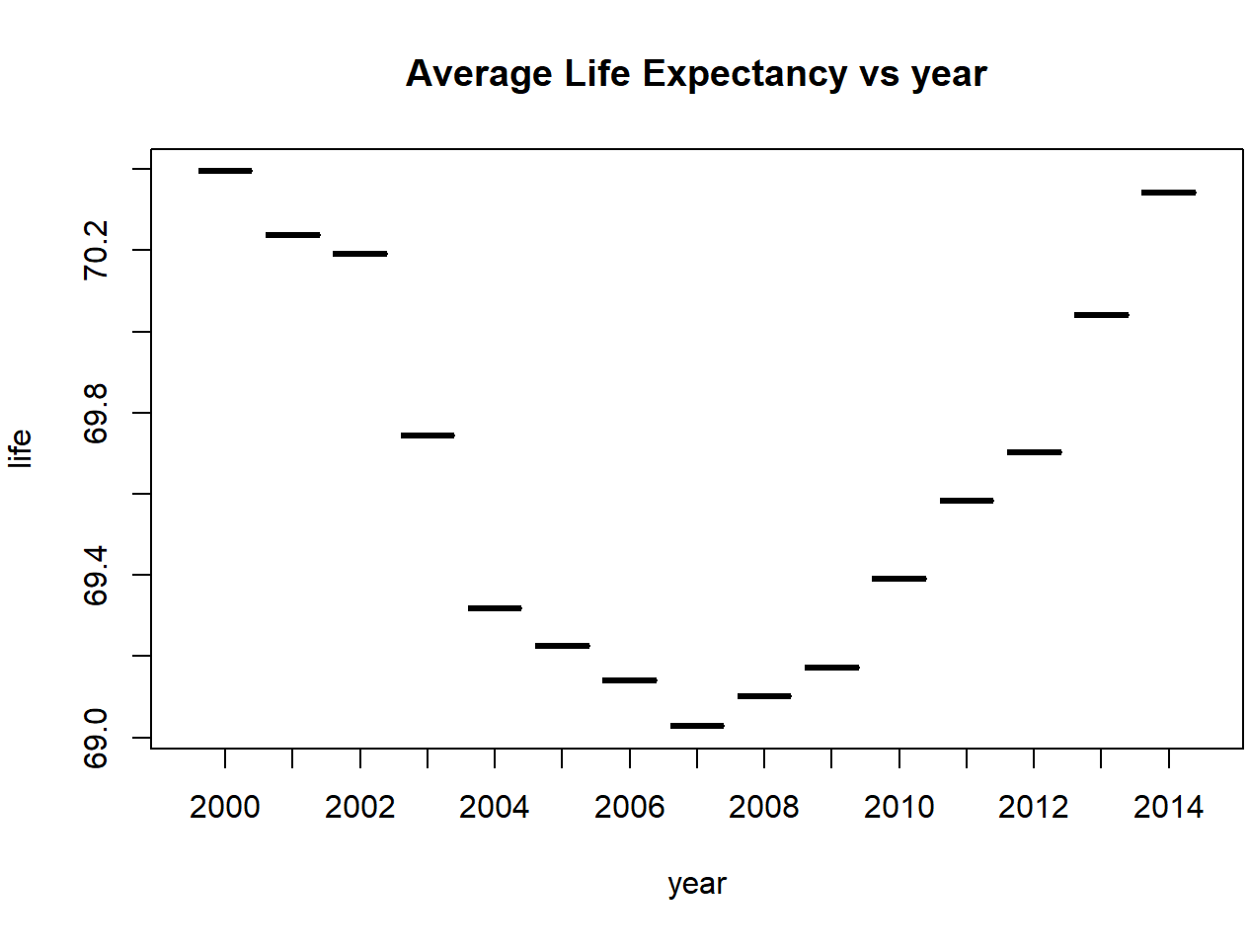
Model output and interpretation

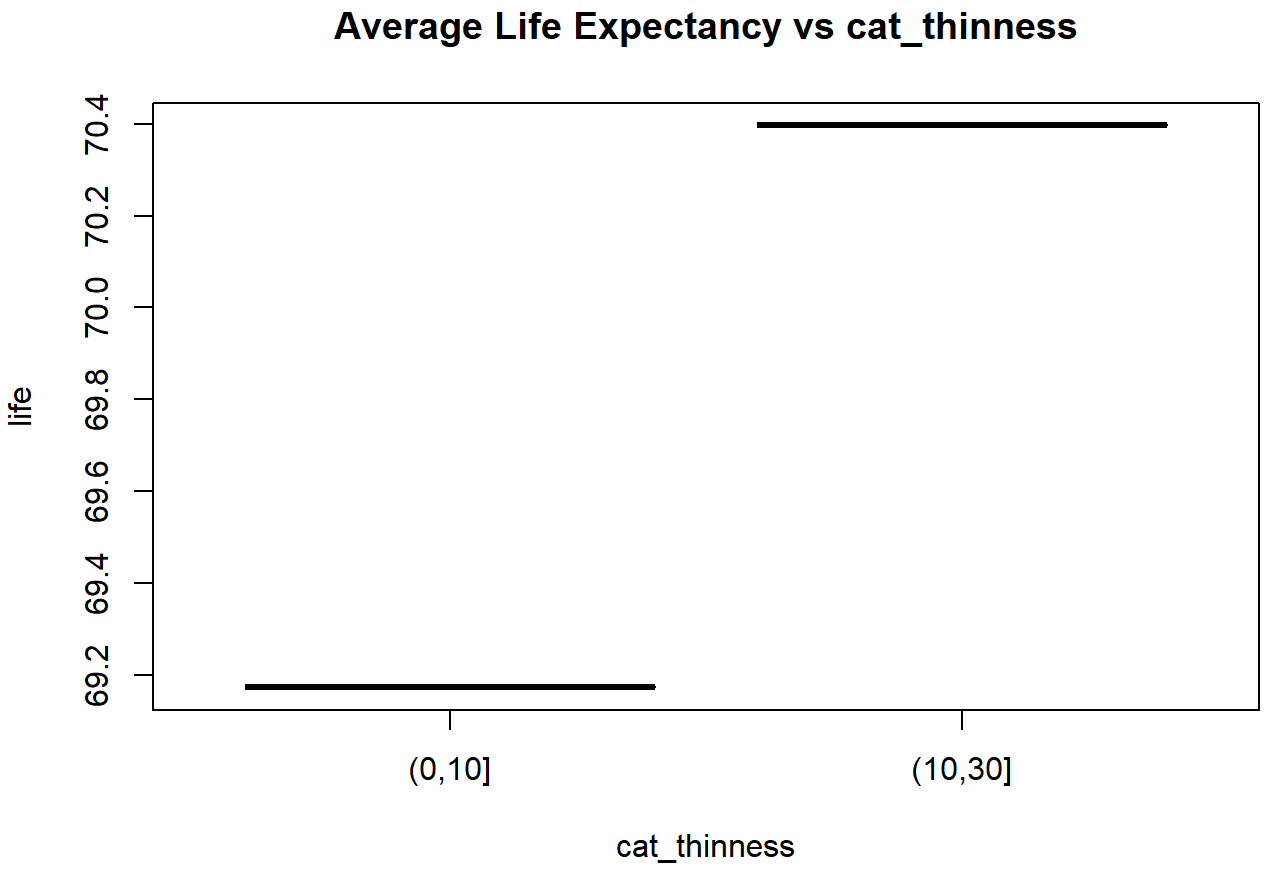
```
model_output(bw)
```


Average Life Expectancy vs log_GDP
life
67
68
69
70
71
72
73
74
2
4
6
8
10
log_GDP


Average Life Expectancy vs school
life
66
68
70
72
74
4
6
8
10
12
14
16
school


Average Life Expectancy vs log_hiv
life
50
55
60
65
70
-2
-1
0
1
2
3
4
log_hiv


Average Life Expectancy vs year
life
69.0
69.4
69.8
70.2
2000
2002
2004
2006
2008
2010
2012
2014
year



Interpretation: The life expectancy of developing countries fluctuates drastically with respect to the log of GDP. There is a clear positive trend between life expectancy and education, while the log transformed HIV has a negative relationship. Life expectancy dips by year around 2007, perhaps a result of the global economic and housing crisis. Contrary to our intuition, the model suggests life expectancy increases slightly when a larger percentage of the population are malnourished.

Point and interval estimates

Point and interval estimate of

$$E(Y|X_1 = x_1, X_2 = x_2, X_3 = x_3, X_4 = x_4, X_5 = x_5)$$

and

$$E(Y|X_1 = x_1, X_2 = x_2, X_3 = x_3 + \Delta_3, X_4 = x_4, X_5 = x_5)$$

when $(x_1, x_2, x_3, x_4, x_5) = (3000, 10.7, 0.1, 2009, (0, 10])$ and $\Delta_3 = 1$:

```
#Choose combination of predictor values and change in hiv
x = c(3000, 10.7, 0.1, 2009, "(0,10]")
d = 1

#Create data frame of points for which the conditional mean will be estimated
```

```
eval_pts = data.frame(life_pred = NA, GDP = rep(log(as.numeric(x[1])), 2),
                      school = rep(x[2], 2),
                      hiv = log(as.numeric(x[3]) + c(0, d)),
                      year = rep(x[4], 2), cat_thinness = rep(x[5], 2))
eval_pts[,2:5] <- lapply(eval_pts[,2:5], as.numeric)
eval_pts[,5:6] <- lapply(eval_pts[,5:6], as.ordered)

#Compute and display mhat
mhat = npreg(bw, exdat = eval_pts[,-1])
eval_pts[,1] <- mhat$mean
eval_pts
```

```
  life_pred      GDP school         hiv year cat_thinness
1  72.37661 8.006368   10.7 -2.30258509 2009       (0,10]
2  69.09870 8.006368   10.7  0.09531018 2009       (0,10]
```

```
#Compute and display 95% confidence interval bounds for mhat
alpha = 0.95
z = qnorm(0.5 + 0.5*alpha)

LB = mhat$mean - z*mhat$merr
UB = mhat$mean + z*mhat$merr

CI1 = paste("95% CI for GDP = ", x[1], ", school = ", x[2], ", hiv = ", x[3], ", year = ", x[4],
CI2 = paste("95% CI for GDP = ", x[1], ", school = ", x[2], ", hiv = ", as.numeric(x[3]) + d, ",
print(CI1)
```

```
[1] "95% CI for GDP = 3000, school = 10.7, hiv = 0.1, year = 2009, cat_thinness = (0,10]: (71.01,
73.75)"
```

```
print(CI2)
```

```
[1] "95% CI for GDP = 3000, school = 10.7, hiv = 1.1, year = 2009, cat_thinness = (0,10]: (60.62,
77.57)"
```

Interpretation: Our model predicts that the life expectancy of a country with a per-capita GDP of $3000, an average of 10.7 years of schooling, and an HIV prevalence of 0.1 per 1000 people in the year 2009 with 0-10% of the population between ages 5-9 having a low BMI, is 71.48 years with 95% confidence interval bounds of (70.04, 72.93). Increasing the log of HIV prevalence by 1 results in a predicted life expectancy of 67.23 years with a 95% confidence interval of (64.11, 70.35).

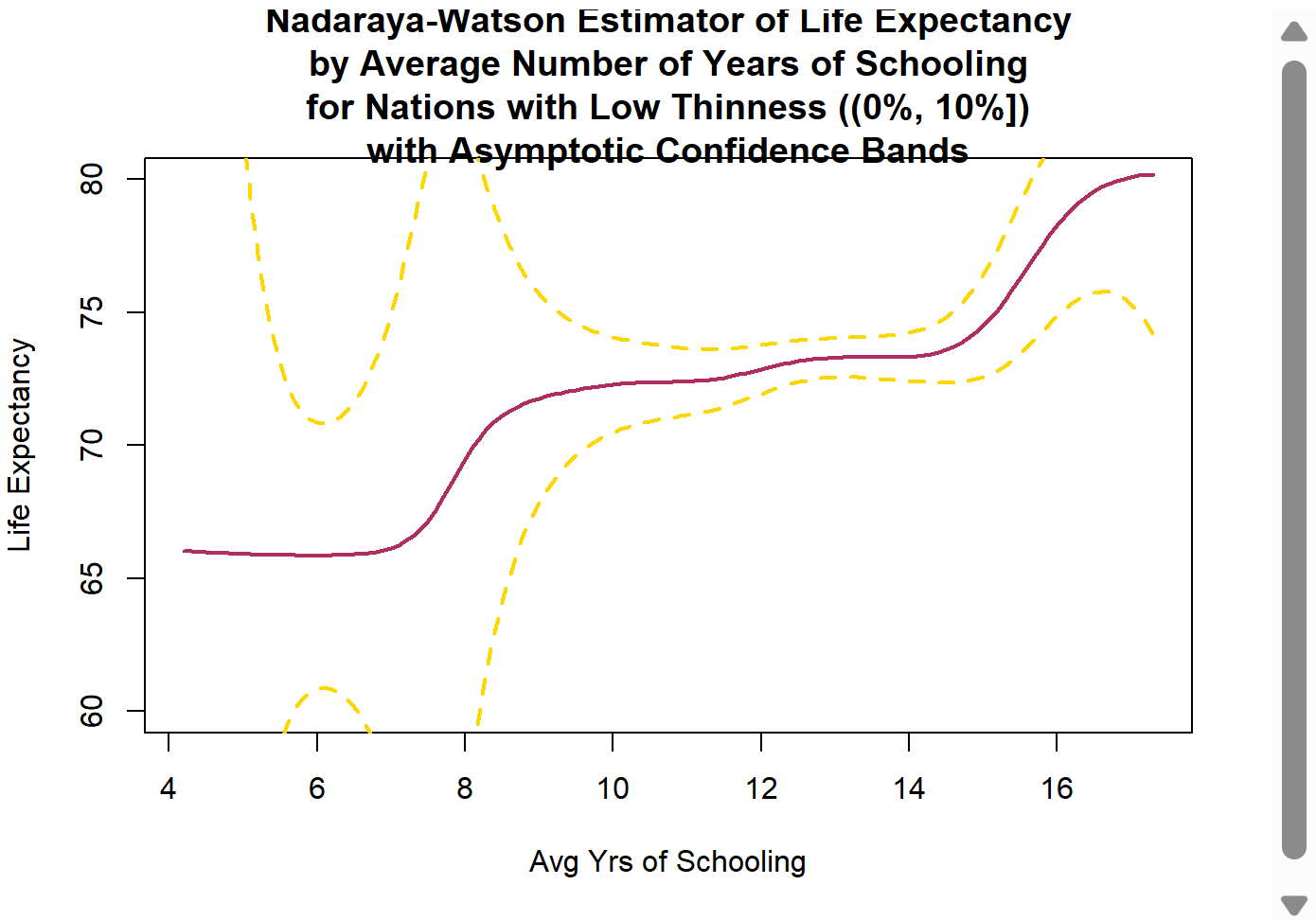
Let $(x_1, x_3, x_4) = (3000, 0.1, 2009)$, create a plot that displays a point and interval estimate of:

$$x_2 \mapsto E(Y \mid X_1 = x_1, X_2 = x_2, X_3, X_4 = x_4, X_5 \in [0, 10])$$

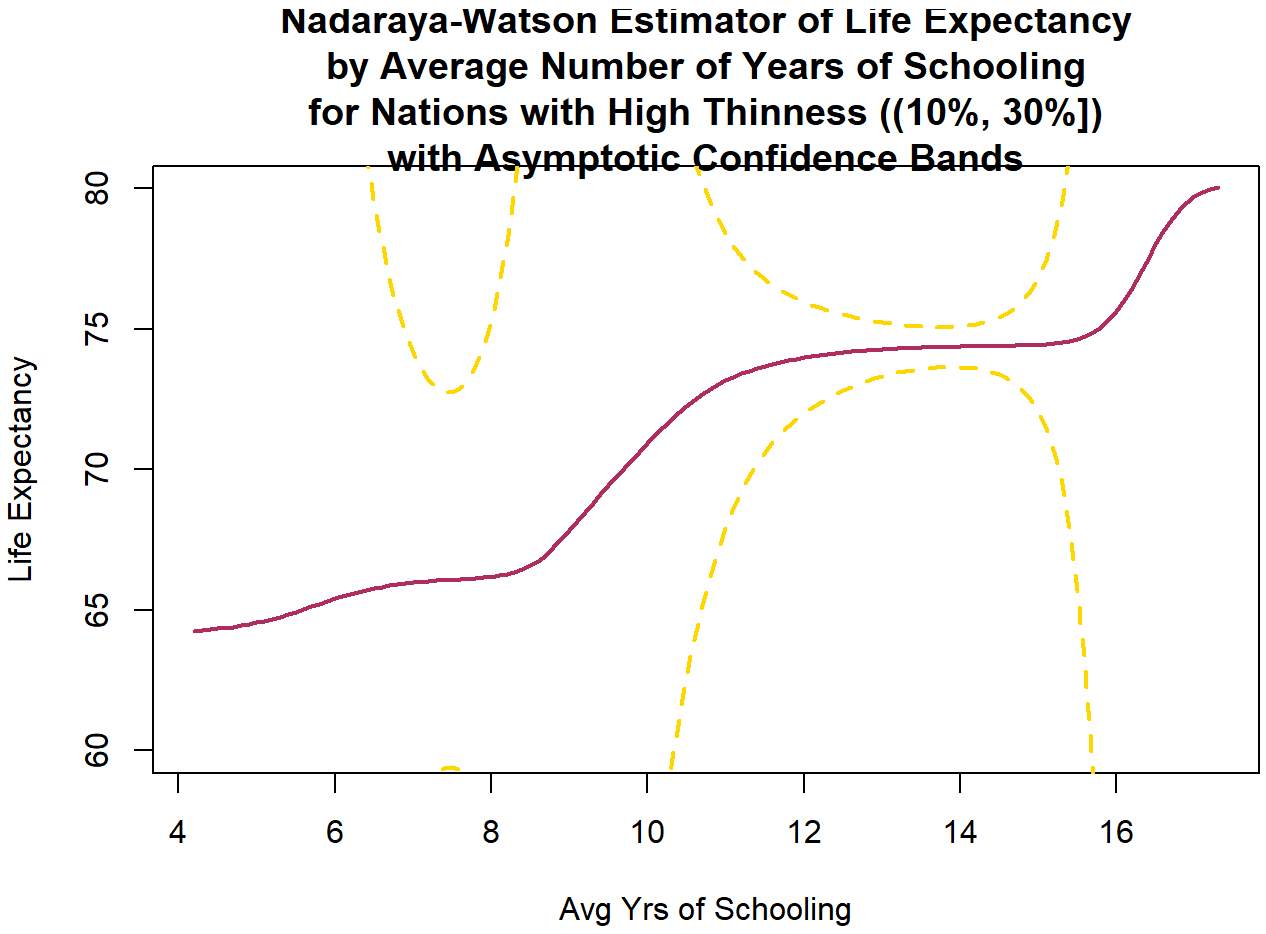
and

$$x_2 \mapsto E(Y \mid X_1 = x_1, X_2 = x_2, X_3, X_4 = x_4, X_5 \in (10, 30]).$$

```r
#Plot mhat vs x2 for fixed x1, x3, and x4 and both levels of x5
x2 = seq(min(df_who$school), max(df_who$school), by = 0.1)
n = length(x2)

#Create data frame of points for which the conditional mean will be estimated
eval_pts = data.frame(life_pred = NA, GDP = rep(log(as.numeric(x[1])), 2*n),
                      school = rep(x2, 2), hiv = rep(log(as.numeric(x[3])), 2*n),
                      year = rep(x[4], 2*n), cat_thinness = rep(c("(0,10]", "(10,30]"), c(n, n)))
eval_pts[,2:5] <- lapply(eval_pts[,2:5], as.numeric)
eval_pts[,5:6] <- lapply(eval_pts[,5:6], as.ordered)

#Compute mhat
mhat = npreg(bw, exdat = eval_pts[,-1])
eval_pts[,1] <- mhat$mean

#Compute 95% confidence bands for mhat
LB = mhat$mean - z*mhat$merr
UB = mhat$mean + z*mhat$merr

#Display plots
plot(x2, eval_pts$life_pred[1:n], type = "l", col = "maroon", lwd = 2, xlab = "Avg Yrs of Schooli
lines(x2, LB[1:n], col = "gold", lwd = 2, lty = 2)
lines(x2, UB[1:n], col = "gold", lwd = 2, lty = 2)
```



```
plot(x2, eval_pts$life_pred[(n+1):(2*n)], type = "l", col = "maroon", lwd = 2, xlab = "Avg Yrs of
lines(x2, LB[(n+1):(2*n)], col = "gold", lwd = 2, lty = 2)
lines(x2, UB[(n+1):(2*n)], col = "gold", lwd = 2, lty = 2)
```



Interpretation: For both levels of thinness, confidence bands vary wildly around 4-9 years of schooling, presumably resulting from these areas having only few observations compared to the other regions of the plots. Regardless, we see a clear upward trend in life expectancy per year of school for each category.

Model 2: Local constant regression with Naive Bootstrap Confidence Bands

The Nadaraya-Watson estimator is also known as the local constant regression estimator. Rather than rely on the asymptotic distribution of $\hat{m}$, which assumes certain regularity conditions, we can instead estimate its distribution via bootstrapping.

Note that $\hat{m}$ is a function which depends on the data $X = (\vec{X}_1, \ldots, \vec{X}_n)^T$ (where $\vec{X}_i$ is the vector of values of the predictors for the i^{th} observation) and Y. In general, one can obtain a realization of $\hat{m}$ by sampling observations $\vec{X}_1^*, \ldots, \vec{X}_n^*, \vec{Y}^*$ from the joint distribution F of $X_1, \ldots, X_p, Y$ and computing

$$\hat{m}(\vec{x}^*; H) = \sum_{i=1}^{n} W(\vec{X}_i^*) Y_i^* \ .$$

Doing so across multiple samples would result in a simulated distribution of $\hat{m}$. However, since F is often unknown in practice, this approach is unrealistic. A nonparametric solution to this problem is to instead sample $\vec{X}_1^*, \ldots, \vec{X}_n^*, \vec{Y}^*$ from the empirical joint distribution

$$\hat{F}_n(\vec{x}, y) := \sum_{i=1}^{n} \{\mathbb{I}(Y \leq y) \prod_{j=1}^{p} \mathbb{I}(X_j \leq x_j)\}$$

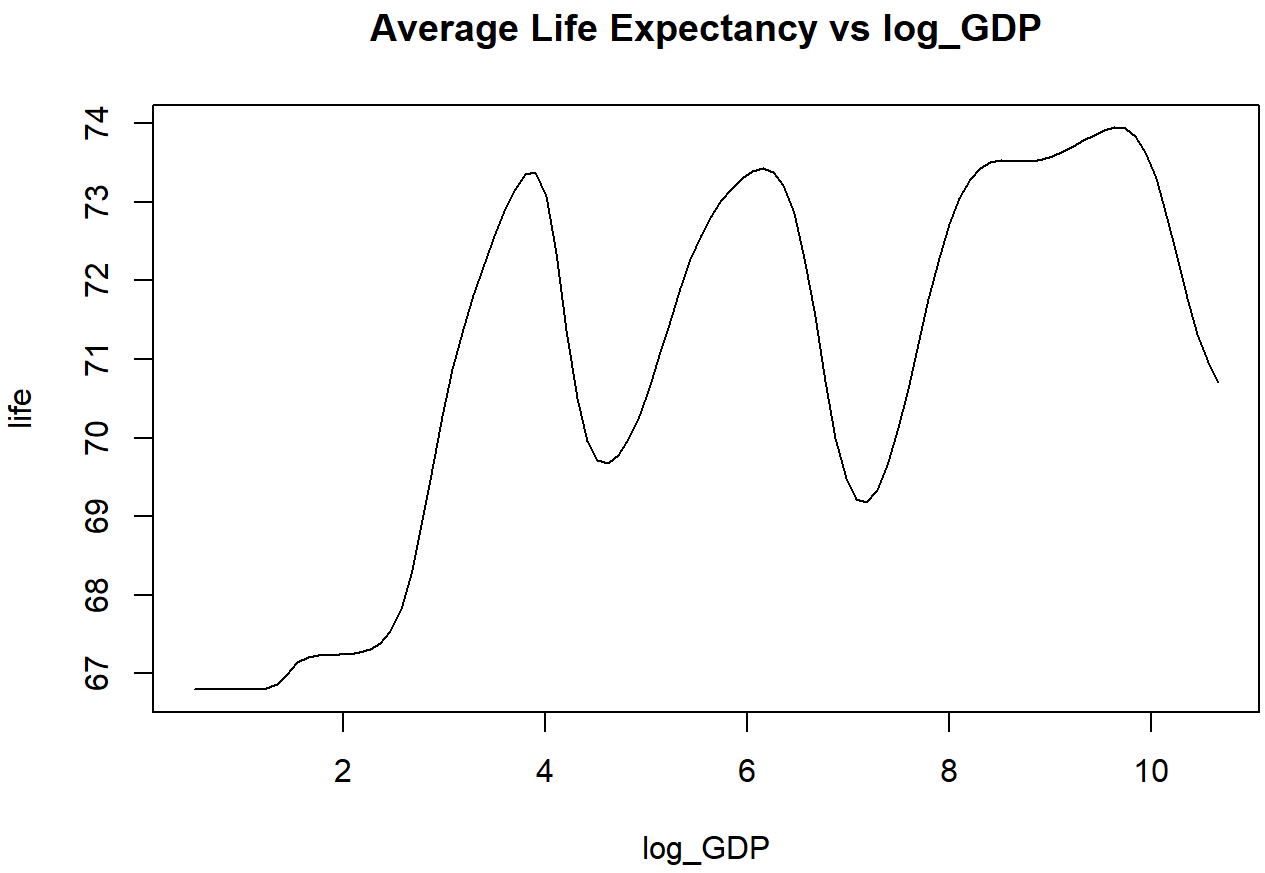
(note that this is the same as sampling observations with replacement from the original data). Once we obtain B samples and compute $\hat{m}_b$ for samples $b = 1, \ldots, B$, we can obtain interval bounds from the quantiles of $\hat{m}_1, \ldots, \hat{m}_B$. Specifically, the lower and upper bounds of the $(1 - \alpha) \times 100$ percentile interval of $\hat{m}$ are the $(\alpha/2)^{th}$ and $(1 - \alpha/2)^{th}$ percentiles of $\hat{m}_1, \ldots, \hat{m}_B$, respectively.

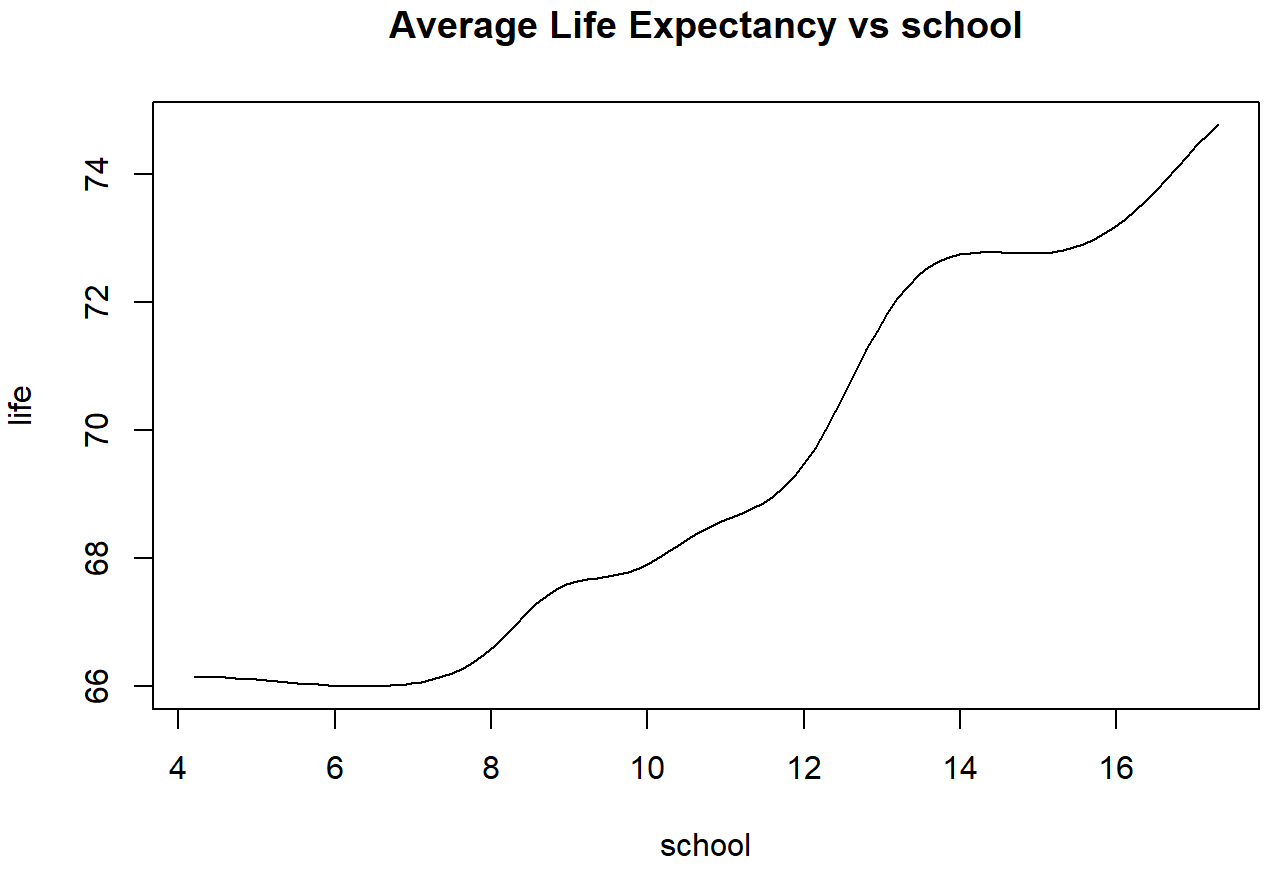
Assumptions

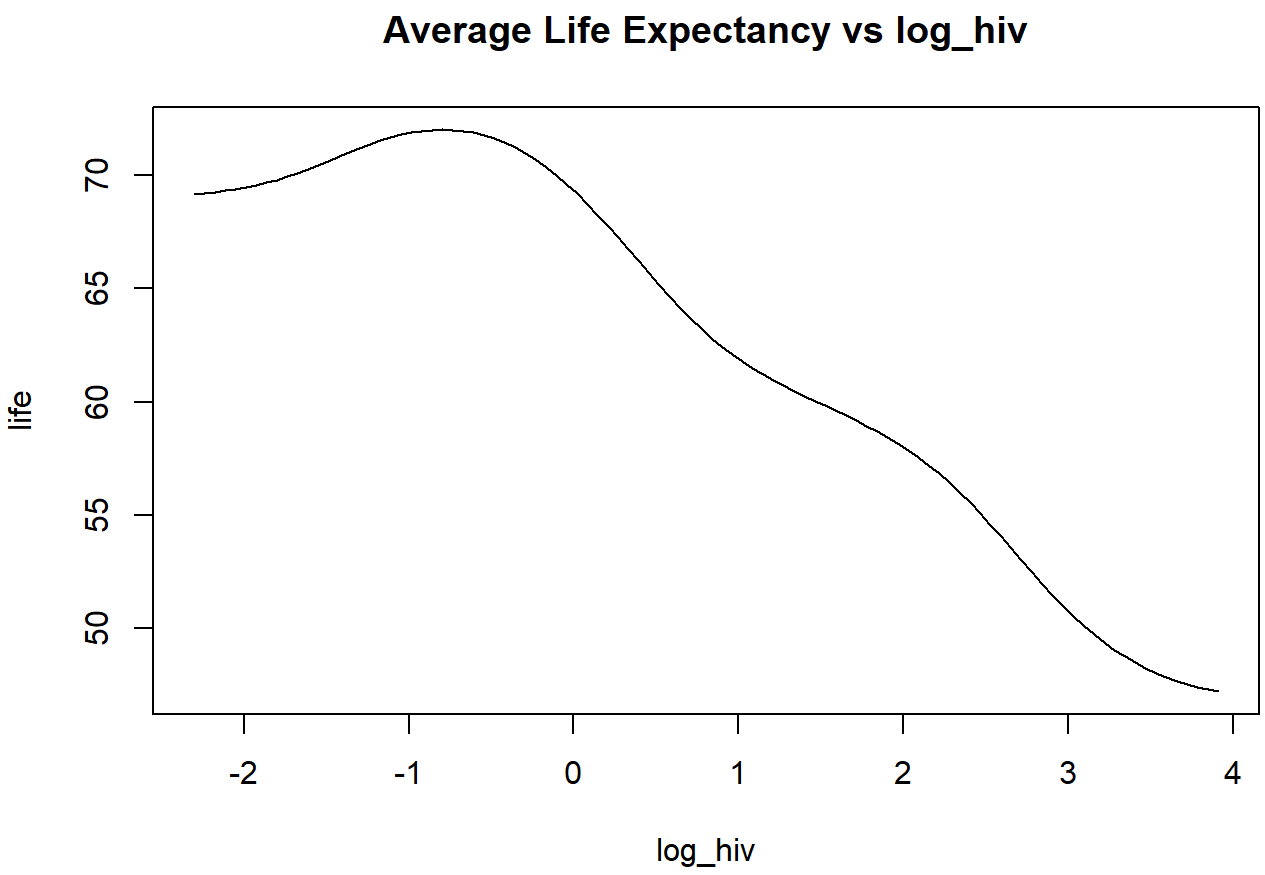
This model imposes the same assumptions as model 1, except that there are no regularity conditions on m and there is no requirement that m is twice differentiable with respect to the quantitative predictors; these were needed for the asymptotic distribution of $\hat{m}$, but are no longer necessary when we estimate the distribution using bootstrap.

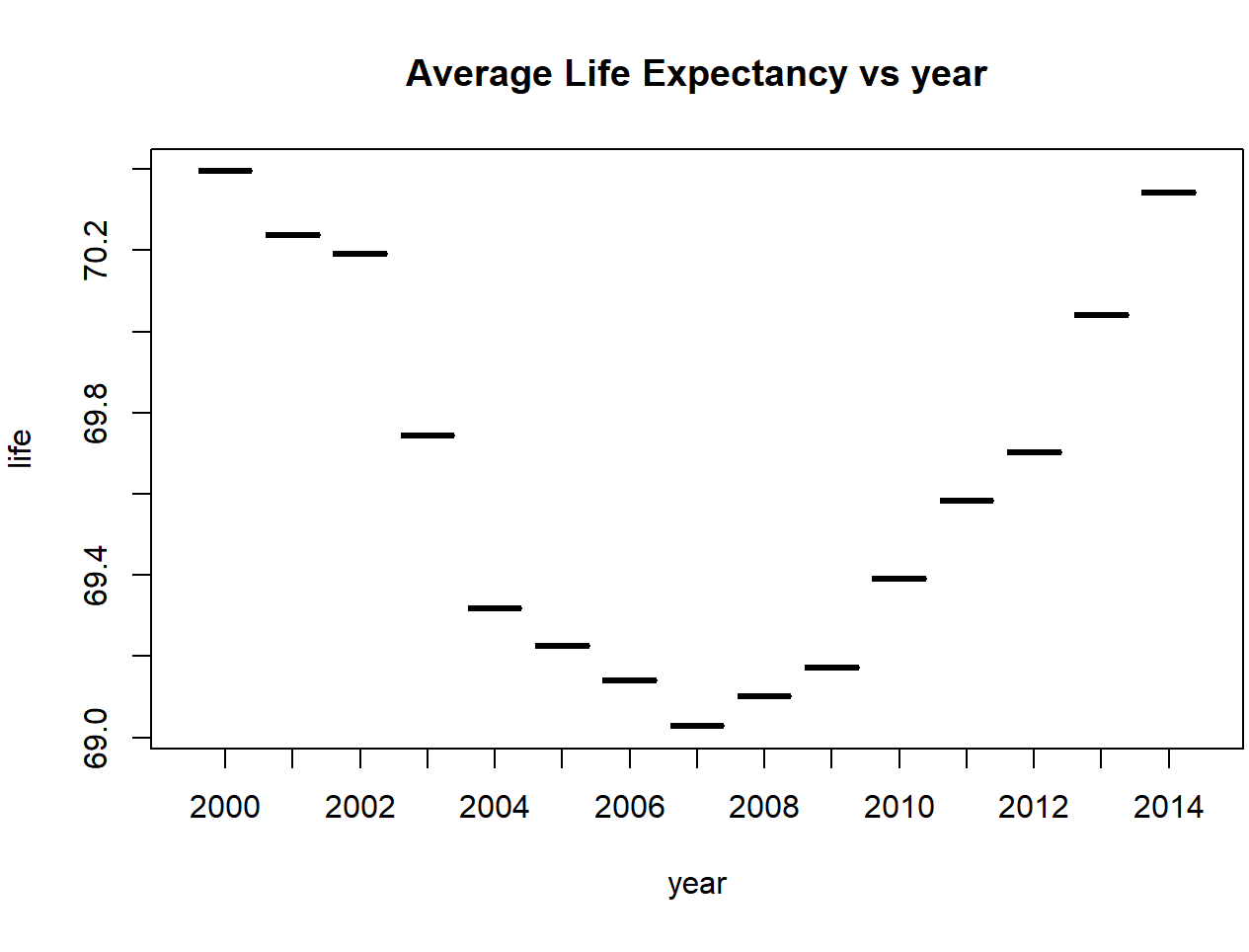
Model output and interpretation

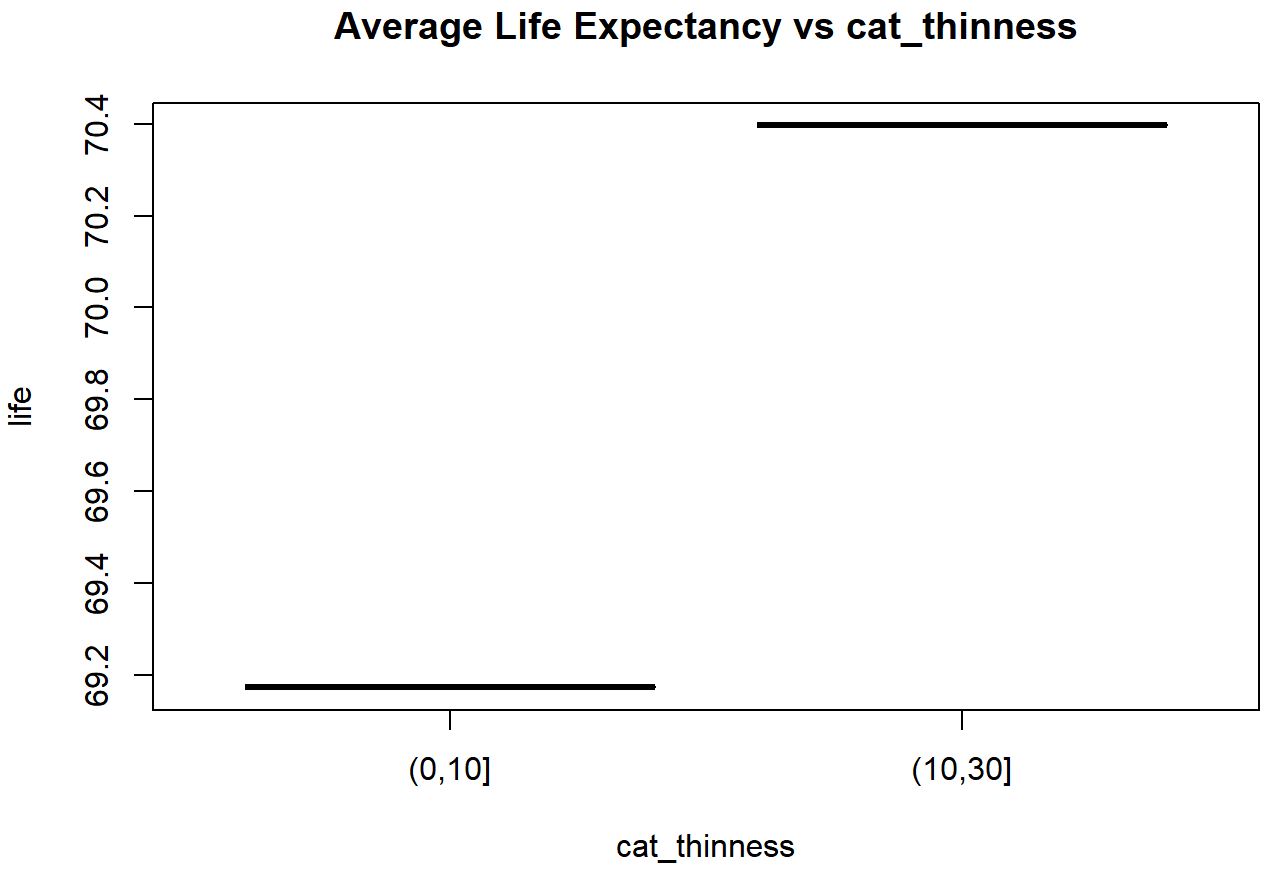
```
model_output(bw)
```


Average Life Expectancy vs log_GDP
life
67
68
69
70
71
72
73
74
2
4
6
8
10
log_GDP


Average Life Expectancy vs school
life
66
68
70
72
74
4
6
8
10
12
14
16
school


Average Life Expectancy vs log_hiv
life
50
55
60
65
70
-2
-1
0
1
2
3
4
log_hiv


Average Life Expectancy vs year
life
69.0
69.4
69.8
70.2
2000
2002
2004
2006
2008
2010
2012
2014
year



Interpretation: Since the local constant regression estimator is the same as the Nadaraya-Watson estimator, plot outputs and interpretations are the same as before.

Point and interval estimates

Point and interval estimate of

$$E(Y|X_1 = x_1, X_2 = x_2, X_3 = x_3, X_4 = x_4, X_5 = x_5)$$

and

$$E(Y|X_1 = x_1, X_2 = x_2, X_3 = x_3 + \Delta_3, X_4 = x_4, X_5 = x_5)$$

when $(x_1, x_2, x_3, x_4, x_5) = (3000, 10.7, 0.1, 2009, (0, 10])$ and $\Delta_3 = 1$:

```
#NW estimator naive bootstrap function
LCR_boot <- function(B, n, eval_df){
  T_n = matrix(NA, nrow = nrow(eval_df), ncol = B)
  for(b in 1:B){
    #Sample observations
    samp = sample(1:nrow(df_who), n, replace = T)
    df_samp = df_who[samp,]

    #Obtain estimates based on sampled data
```

```
    life_pred = npreg(bw, txdat = df_samp[,-1], tydat = df_samp[,1],
                      exdat = eval_df[,-1])$mean
    T_n[,b] = life_pred
  }
  return(T_n)
}

#Percentile interval bounds function
perc_int <- function(T_n, conf_level = 0.95){
  #Obtain quantiles
  alpha_half = 0.5 - 0.5*conf_level
  quants = c(alpha_half, 1 - alpha_half)

  #Obtain bounds based on quantiles
  n_bounds = nrow(T_n)
  bounds = matrix(NA, nrow = n_bounds, ncol = 2)
  for(i in 1:n_bounds){
    bounds[i,] = quantile(T_n[i,], quants)
  }
  return(bounds)
}

#Set seed
set.seed(20241118)

#Create data frame of points for which the conditional mean will be estimated
eval_pts <- data.frame(life_pred = NA, log_GDP = rep(log(as.numeric(x[1])), 2),
                       school = rep(x[2], 2),
                       log_hiv = log(as.numeric(x[3]) + c(0, d)),
                       year = rep(x[4], 2),
                       cat_thinness = rep(x[5], 2))
eval_pts[,2:5] <- lapply(eval_pts[,2:5], as.numeric)
eval_pts[,5:6] <- lapply(eval_pts[,5:6], as.ordered)

#Initialize variables
B = 1000
n = nrow(df_who)

#Perform naive bootstrap
mhat_n = LCR_boot(B, n, eval_pts)
eval_pts[,1] = rowMeans(mhat_n)
eval_pts
```

```
  life_pred  log_GDP school     log_hiv year cat_thinness
1  72.29645 8.006368   10.7 -2.30258509 2009       (0,10]
2  69.27508 8.006368   10.7  0.09531018 2009       (0,10]
```

```
#Compute and display 95% percentile interval bounds for mhat
bounds = perc_int(mhat_n)
LB = bounds[,1]
```

```
UB = bounds[,2]

PI1 = paste("95% PI for GDP = ", x[1], ", school = ", x[2],
            ", hiv = ", x[3], ", year = ", x[4], ", cat_thinness = ",
            x[5], ": (", floor(LB[1]*100)/100, ", ",
            ceiling(UB[1]*100)/100, ")", sep = "")
PI2 = paste("95% PI for GDP = ", x[1], ", school = ", x[2], ", hiv = ",
            as.numeric(x[3]) + d, ", year = ", x[4],
            ", cat_thinness = ", x[5], ": (",
            floor(LB[2]*100)/100, ", ",
            ceiling(UB[2]*100)/100, ")", sep = "")
print(PI1)
```

```
[1] "95% PI for GDP = 3000, school = 10.7, hiv = 0.1, year = 2009, cat_thinness = (0,10]: (70.83,
73.79)"
```

```
print(PI2)
```

```
[1] "95% PI for GDP = 3000, school = 10.7, hiv = 1.1, year = 2009, cat_thinness = (0,10]: (66.48,
73.06)"
```

Interpretation: life expectancy for a country with a per-capita GDP of $3000, an average of 10.7 years of schooling, and an HIV prevalence of 0.1 per 1000 people in the year 2009 with 0-10% of the population between ages 5-9 having a low BMI, is 71.41 years with 95% confidence interval bounds of (69.31, 73.27). Increasing the log of HIV prevalence by 1 results in a predicted life expectancy of 67.07 years with a 95% confidence interval of (64.26, 69.77). We note the range of the first percentile interval is about the same as the confidence interval from the previous model, but the interval after increasing the log of HIV prevalence is a bit more precise than that of the previous model.

Let $(x_1, x_3, x_4) = (3000, 0.1, 2009)$, create a plot that displays a point and interval estimate of:

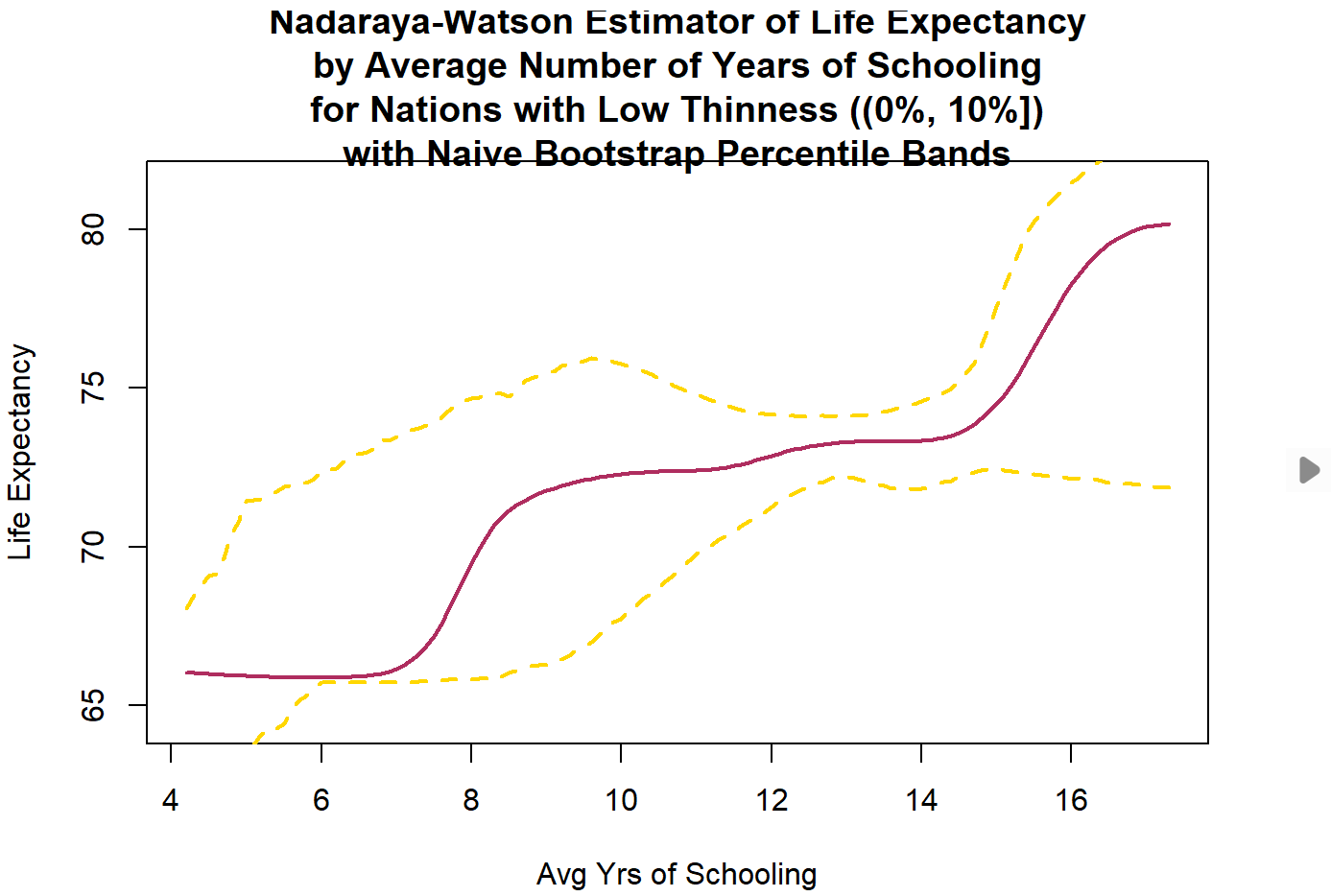
$$x_2 \mapsto E(Y \mid X_1 = x_1, X_2 = x_2, X_3, X_4 = x_4, X_5 \in [0, 10])$$

and

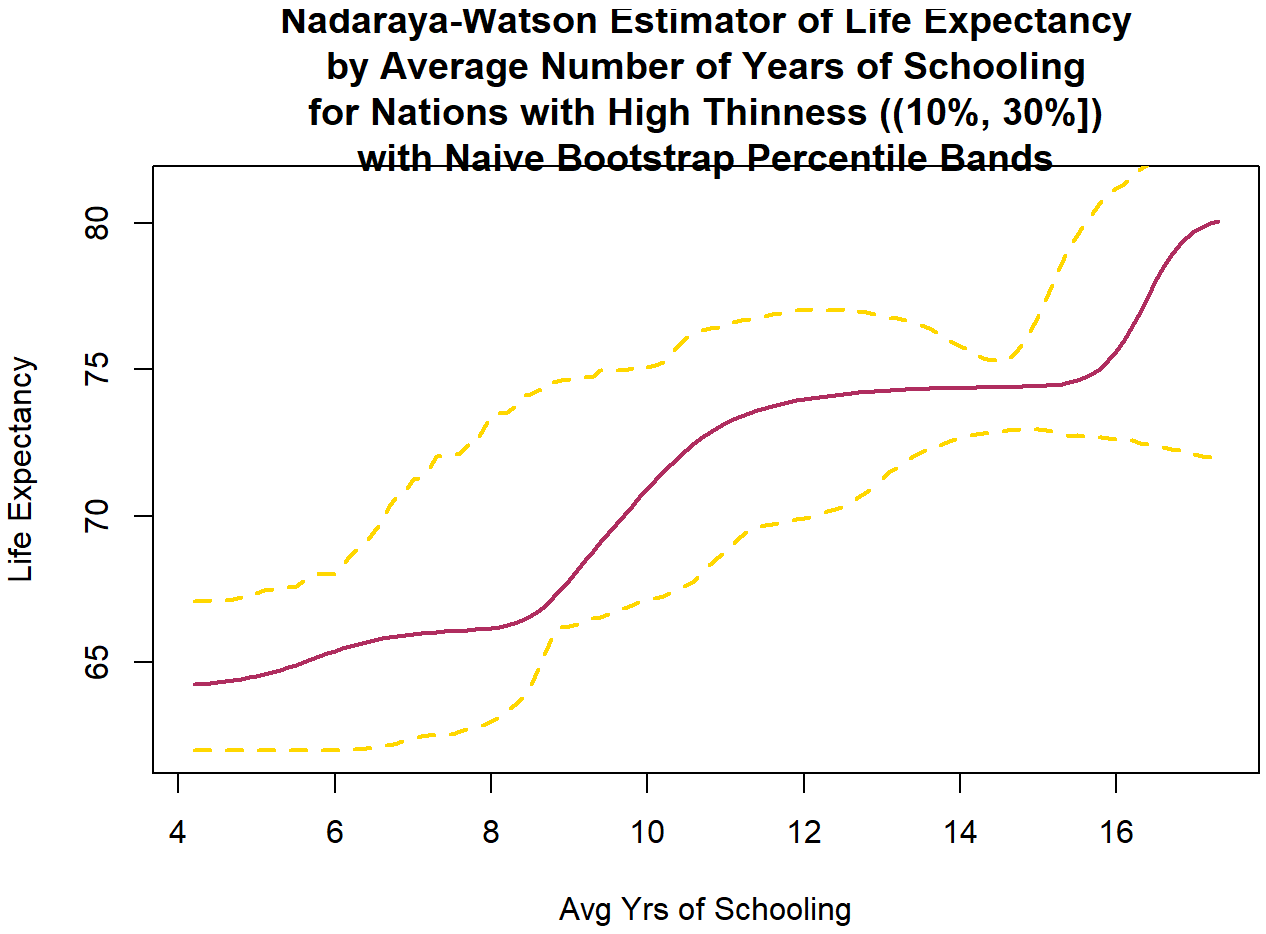
$$x_2 \mapsto E(Y \mid X_1 = x_1, X_2 = x_2, X_3, X_4 = x_4, X_5 \in (10, 30]).$$

```
#Set seed
set.seed(123)

#Plot mhat vs x2 for fixed x1, x3, and x4 and both levels of x5
x2 = seq(min(df_who$school), max(df_who$school), by = 0.1)
n = length(x2)

#Create data frame of points for which the conditional mean will be estimated
eval_pts <- data.frame(life_pred = NA, GDP = rep(log(as.numeric(x[1])), 2*n),
                       school = rep(x2, 2), hiv = rep(log(as.numeric(x[3])), 2*n),
                       year = rep(x[4], 2*n),
                       cat_thinness = rep(c("(0,10]", "(10,30]"),
                                          c(n, n)))
```

```r
eval_pts[,2:5] <- lapply(eval_pts[,2:5], as.numeric)
eval_pts[,5:6] <- lapply(eval_pts[,5:6], as.ordered)

#Perform naive bootstrap
mhat_n = LCR_boot(B, n, eval_pts)

#Compute estimate and 95% percentile bands for life expectancy
eval_pts$life_pred = npreg(bw, exdat = eval_pts[,-1])$mean

bounds = perc_int(mhat_n)
LB = bounds[,1]
UB = bounds[,2]

#Display plots
plot(x2, eval_pts$life_pred[1:n], type = "l", col = "maroon", lwd = 2, xlab = "Avg Yrs of Schooli
lines(x2, LB[1:n], col = "gold", lwd = 2, lty = 2)
lines(x2, UB[1:n], col = "gold", lwd = 2, lty = 2)
```




```r
plot(x2, eval_pts$life_pred[(n+1):(2*n)], type = "l", col = "maroon", lwd = 2, xlab = "Avg Yrs of
lines(x2, LB[(n+1):(2*n)], col = "gold", lwd = 2, lty = 2)
lines(x2, UB[(n+1):(2*n)], col = "gold", lwd = 2, lty = 2)
```



Interpretation: Interval boundaries are considerably improved from the previous estimates and are now much more meaningful. Bootstrapping the local constant regression estimator significantly outperforms the asymptotic distribution toward producing reasonable margins of error which now provide helpful information in context, as opposed to the previous confidence interval boundaries which relay little to no information in the plots produced. We note that these interval boundaries perform significantly less well at the tails.

Model 3: Local linear regression

Local constant regression is a specific case of local polynomial regression, which estimates the conditional mean of Y given a set of values $\vec{x}$ for the predictors $\vec{X}$ according to a weighted average of the observed responses Y_i where the weights are obtained via the kernels at each point of the observed predictors, yielding

$$W(\vec{X}_i) = \frac{K_H(\vec{x} - \vec{X}_i)}{\sum_{j=1}^{n} K_H(\vec{x} - \vec{X}_j)}.$$

Another special case of local polynomial regression is local linear regression, which, like the Nadaraya-Watson estimator, seeks to model $m(\vec{x}) := E(Y|\vec{X} = \vec{x})$ via $\hat{m}(\vec{x}; H) := \sum_{i=1}^{n} W(\vec{X}_i)Y_i$, but uses weights according to

$$W(\vec{X}_i) = \frac{\frac{1}{n}\sum_{i=1}^{n}\left\{\{\prod_{k=1}^{p}(x_k - X_{i,k})^2\}K_H(\vec{x} - \vec{X}_i) - \{\prod_{k=1}^{p}(x_k - X_{i,k})\}\hat{s}_1(\vec{x}; H)\right\}}{\hat{s}_2(\vec{x}; H)\hat{s}_0(\vec{x}; H) - \hat{s}_1(\vec{x}; H)}$$

where $\hat{s}_r(\vec{x}; H) = \sum_{i=1}^{n}(\vec{X}_i - \vec{x})^r K_H(\vec{x} - \vec{X}_i)/n$. Unlike the Nadaraya-Watson estimator of m, the asymptotic distribution of the local linear regression estimator is not as straightforward to derive. Therefore, we estimate its distribution via bootstrapping for the purpose of obtaining confidence bands for our estimate.

Rather than continue to use a naive bootstrap method, we instead use the wild bootstrap. Consider the residuals $\hat{\epsilon}_i = Y_i - \hat{m}(\vec{x}; H)$ (where $\hat{m}$ is now the local linear regression estimator of m). The wild bootstrap residuals are defined as $\epsilon_i^* := \hat{\epsilon}_i \tau_i$ where $\tau_1, \ldots, \tau_n$ are i.i.d Rademacher random variables. For each bootstrap b, we obtain $(\vec{X}_1^*, Y_1^*), \ldots, (\vec{X}_n^*, Y_n^*)$ where $\vec{X}_1^*, \ldots, \vec{X}_n^*$ are sampled with replacement from the original observations of the predictors and $Y_i^* = \epsilon_i^* + \sum_{i=1}^{n} W(\vec{X}_i^*)Y_i$, and compute $\hat{m}_b(\vec{x}; H) := \sum_{i=1}^{n} W(\vec{X}_i^*)Y_i^*$. We then consider the simulated distribution of $m(\vec{x})$ using $\hat{m}_1(\vec{x}; H), \ldots, \hat{m}_B(\vec{x}; H)$ for the purposes of obtaining interval bounds.

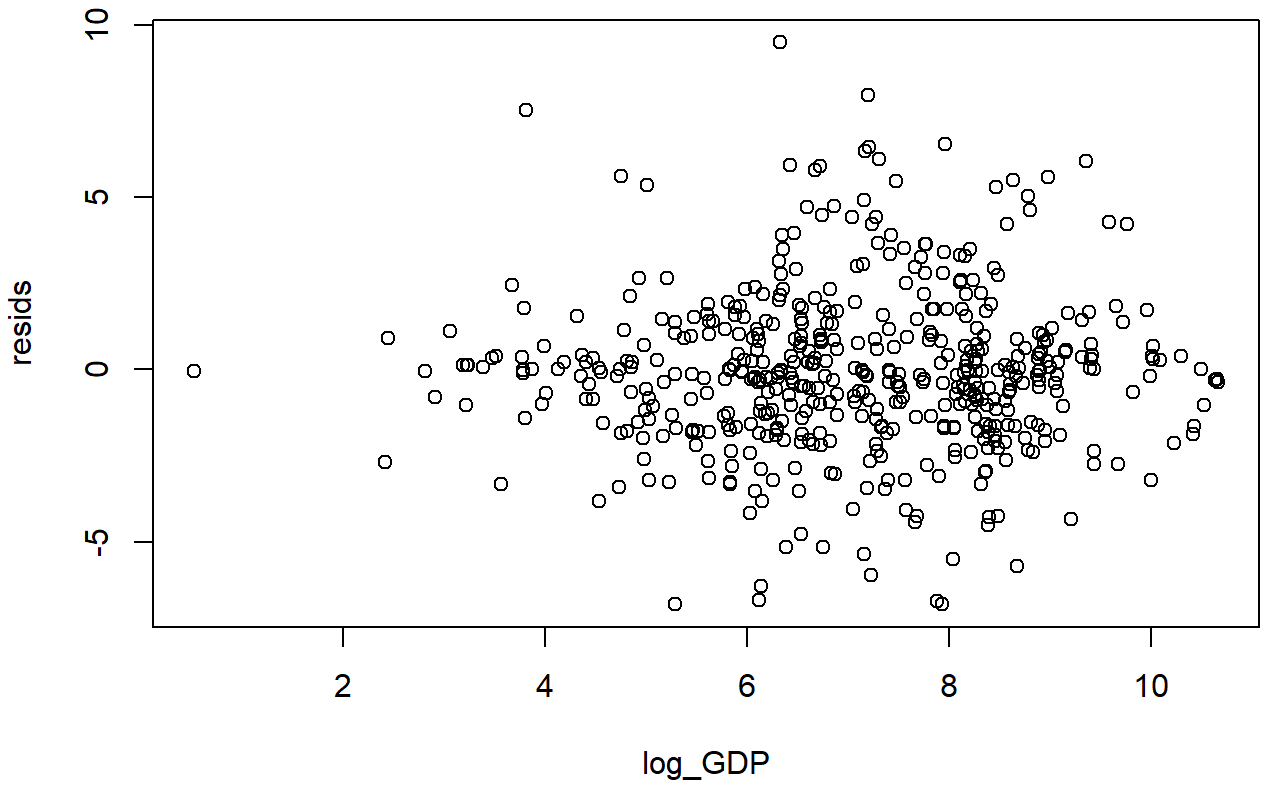
Assumptions

The assumptions for a nonparametric model using the local linear regression estimator are:

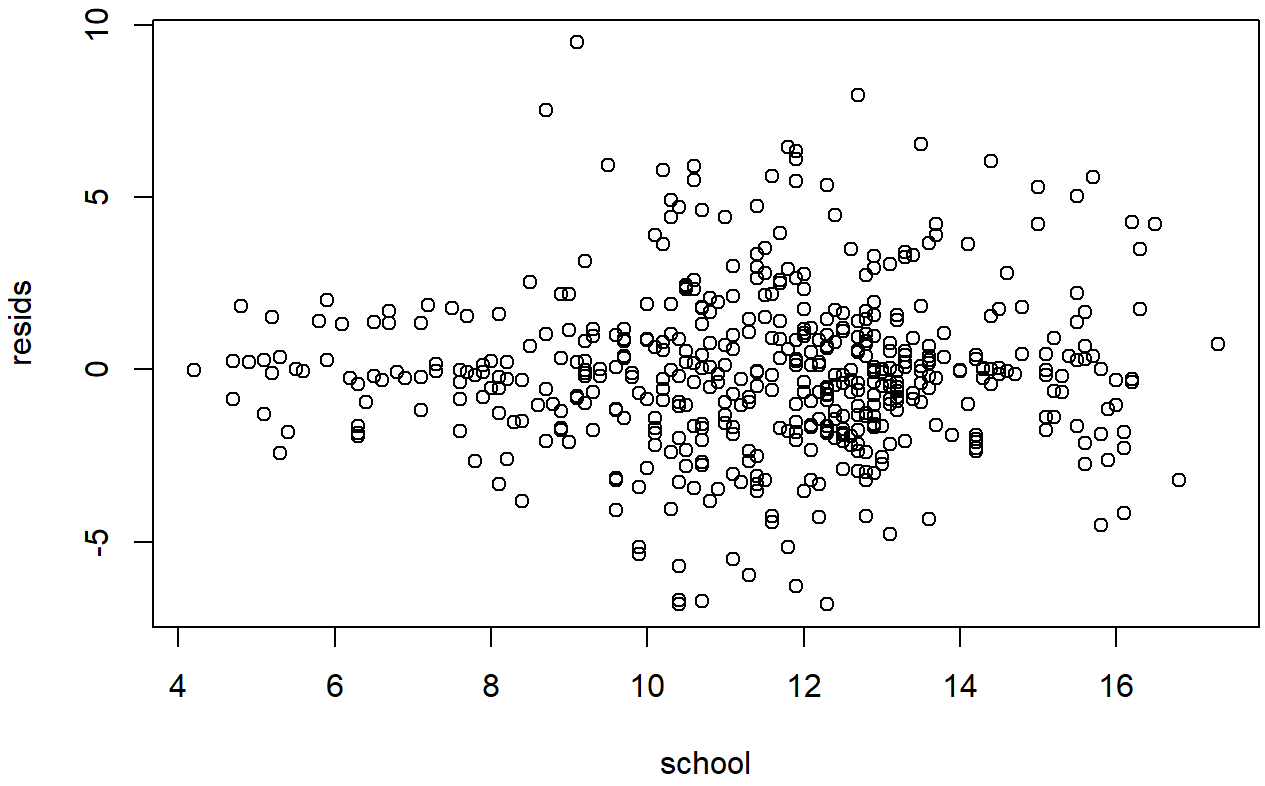
1. Around point $\vec{X}$ the mean of Y can be locally approximated by a linear function
2. Errors in estimating Y are independent and identically Normally distributed with mean 0 and constant variance
3. $|H|$ is small while $n|H|$ is large

```
#Compute least-squares cross-validation bandwidth for local linear estimator
bw <- npregbw(formula = life ~ log_GDP + school + log_hiv + year + cat_thinness, data = df_who, r
```

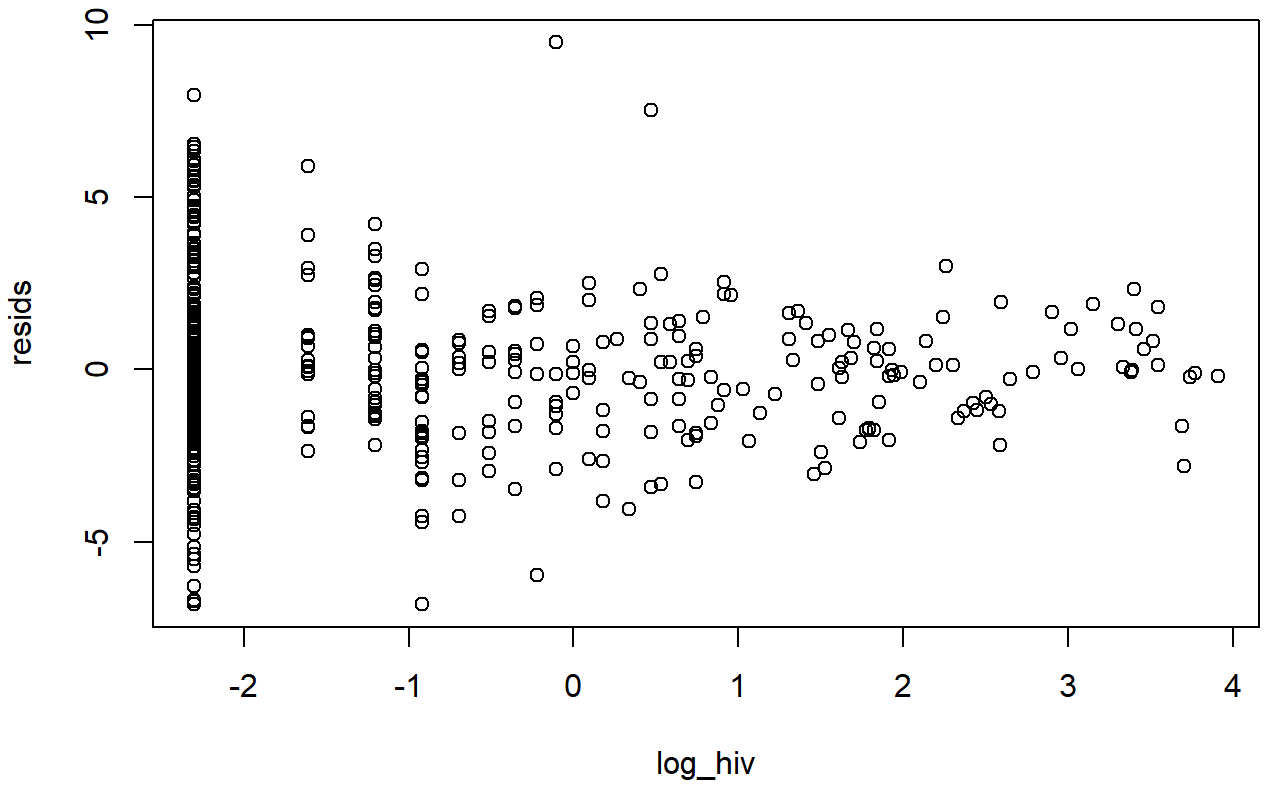
```
#Check residuals are normal with constant variance
attach(df_who)

mhat = npreg(bw)$mean
resids = mhat - life

#Check residual plots
plot(log_GDP, resids)
```



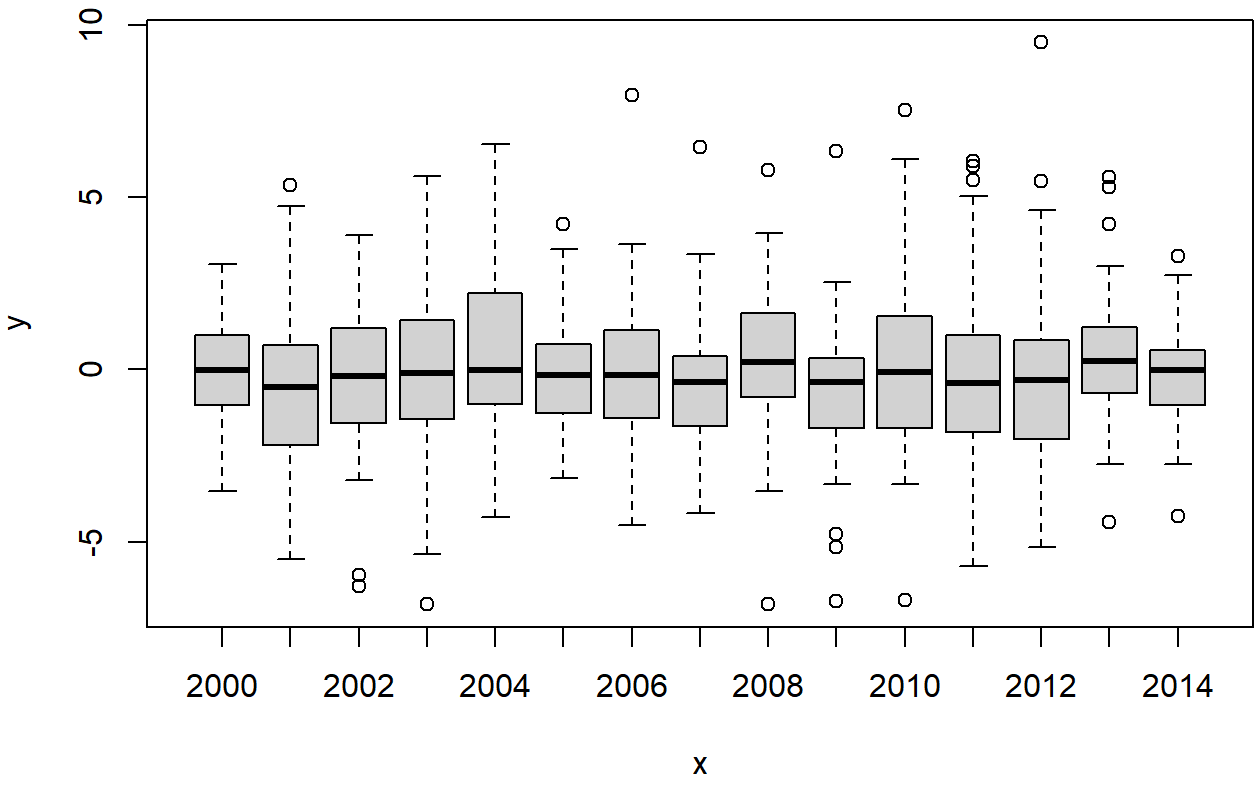

```
plot(school, resids)
```



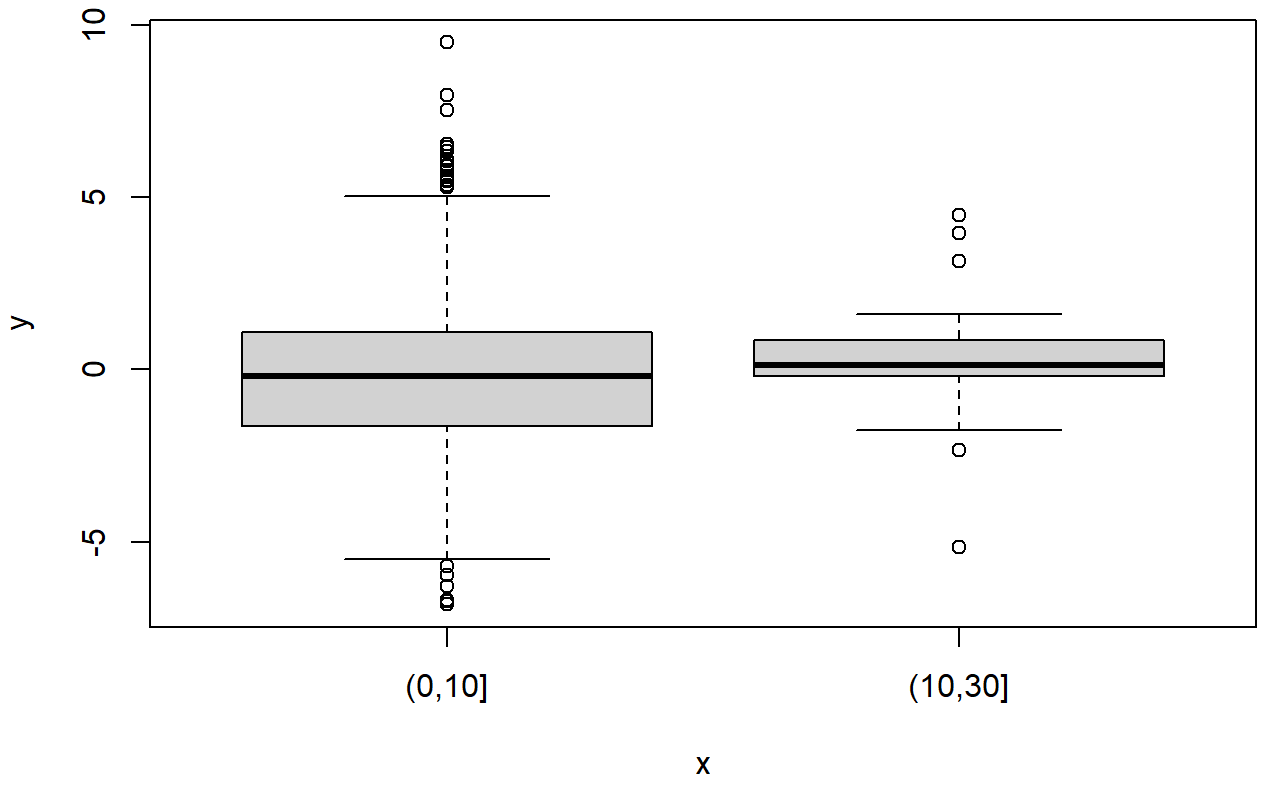

```
plot(log_hiv, resids)
```




```
plot(year, resids)
```




```
plot(cat_thinness, resids)
```




```
#Check if residuals are multivariate normal given values of quantitative predictors
MVN::mvn(data = cbind(resids, df_who[,2:4]))$multivariateNormality
```

```
           Test       HZ p value MVN
1 Henze-Zirkler 6.519326       0  NO
```

```
#Check if residuals are multivariate normal given levels of categorical predictors
for(i in levels(year)){
  LF = lillie.test(resids[which(year==i)])
  print(paste("Lilliefors test of residuals for year", i, "p-value =", LF$p.value))
}
```

```
[1] "Lilliefors test of residuals for year 2000 p-value = 0.459785688447315"
[1] "Lilliefors test of residuals for year 2001 p-value = 0.316604092547404"
[1] "Lilliefors test of residuals for year 2002 p-value = 0.403407166203053"
[1] "Lilliefors test of residuals for year 2003 p-value = 0.349662948239595"
[1] "Lilliefors test of residuals for year 2004 p-value = 0.0969645465510271"
[1] "Lilliefors test of residuals for year 2005 p-value = 0.137396805467512"
[1] "Lilliefors test of residuals for year 2006 p-value = 0.346721677403117"
[1] "Lilliefors test of residuals for year 2007 p-value = 0.187259214049721"
[1] "Lilliefors test of residuals for year 2008 p-value = 0.252773676403931"
[1] "Lilliefors test of residuals for year 2009 p-value = 0.194292385258702"
[1] "Lilliefors test of residuals for year 2010 p-value = 0.302500097874976"
```

```
[1] "Lilliefors test of residuals for year 2011 p-value = 0.135954456614526"
[1] "Lilliefors test of residuals for year 2012 p-value = 0.0404744418302209"
[1] "Lilliefors test of residuals for year 2013 p-value = 0.378430875863033"
[1] "Lilliefors test of residuals for year 2014 p-value = 0.0179817955368328"
```

```
for(i in levels(cat_thinness)){
  LF = lillie.test(resids[which(cat_thinness==i)])
  print(paste("Lilliefors test of residuals for cat_thinness", i, "p-value =",
              LF$p.value))
}
```

```
[1] "Lilliefors test of residuals for cat_thinness (0,10] p-value = 2.05571842567807e-06"
[1] "Lilliefors test of residuals for cat_thinness (10,30] p-value = 7.99723168375219e-05"
```

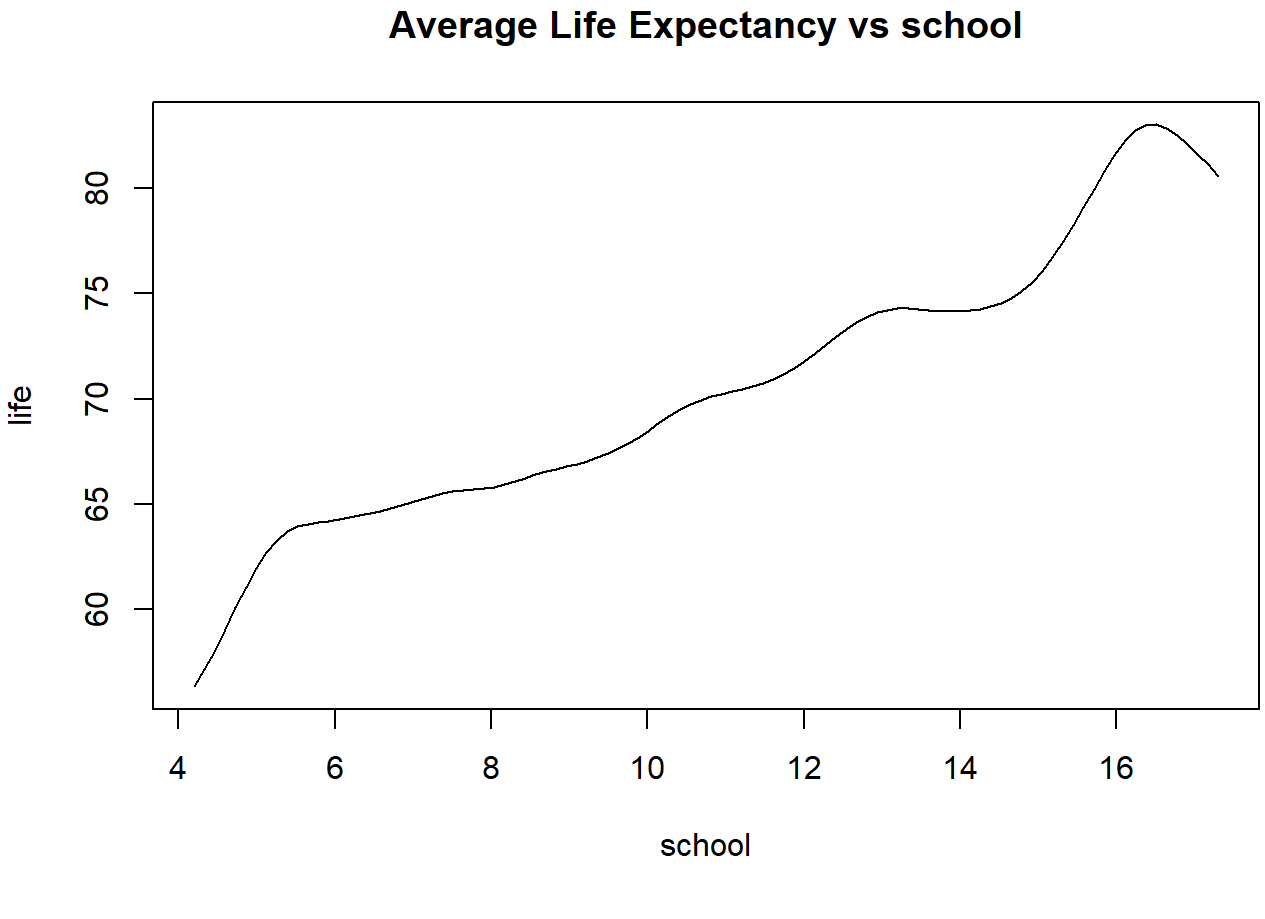
```
detach(df_who)
```

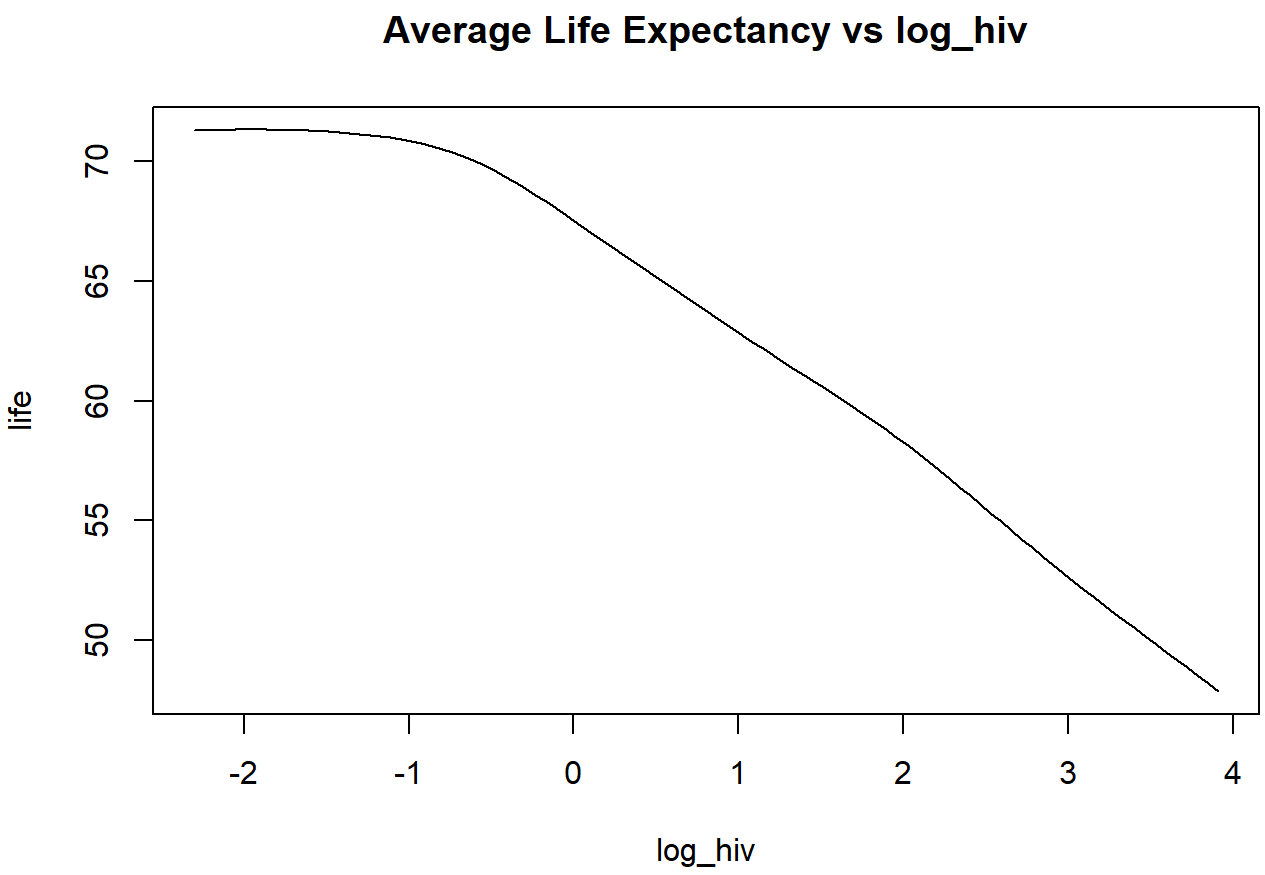
Diagnostic plots show some improvement from prior models toward assumption verification. Residual plots show some clustering, but the assumption of nonconstant variance seems less suspect. Normality tests are also improved, with few significant results from the Lilliefors tests and generally higher p values across the board.

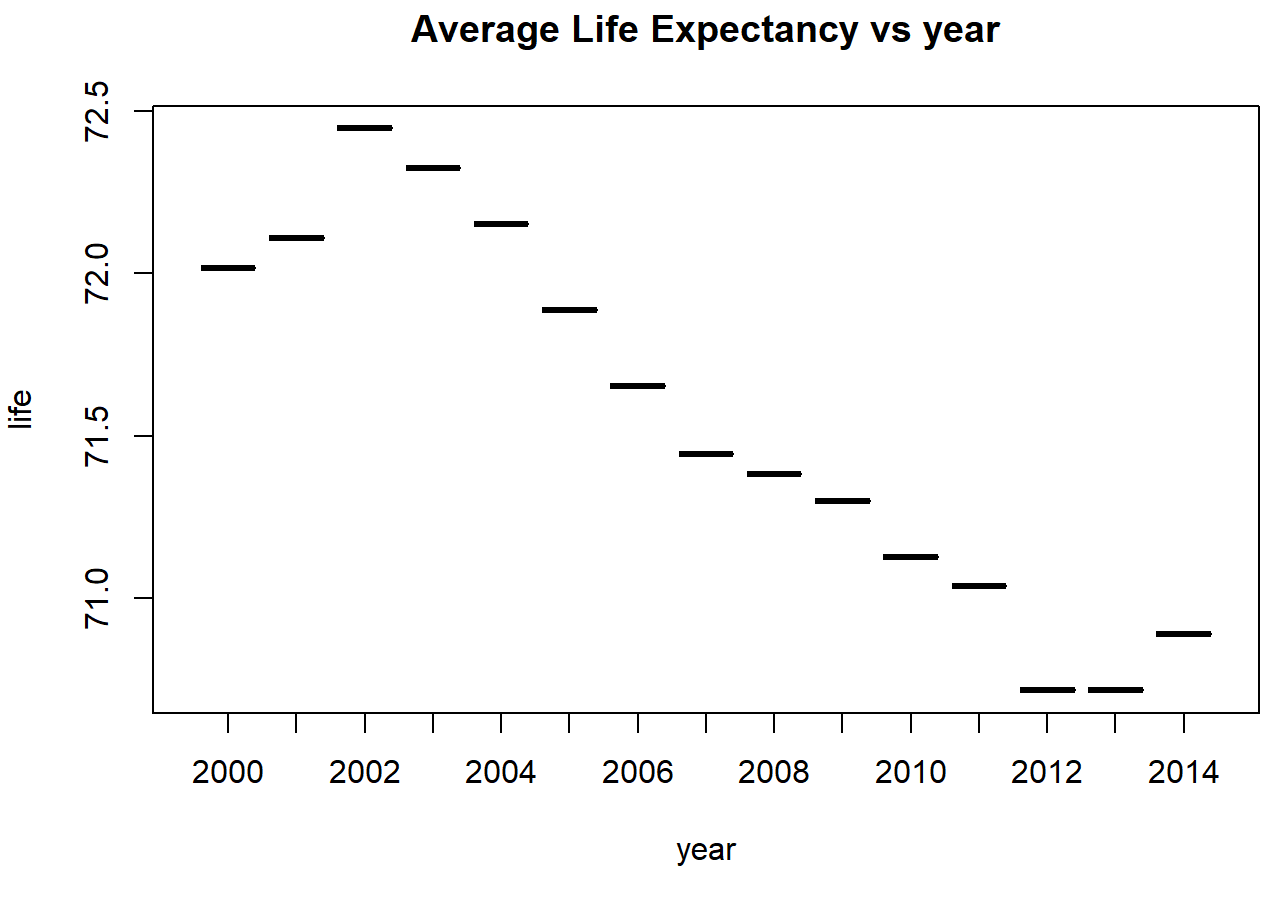
Model output and interpretation

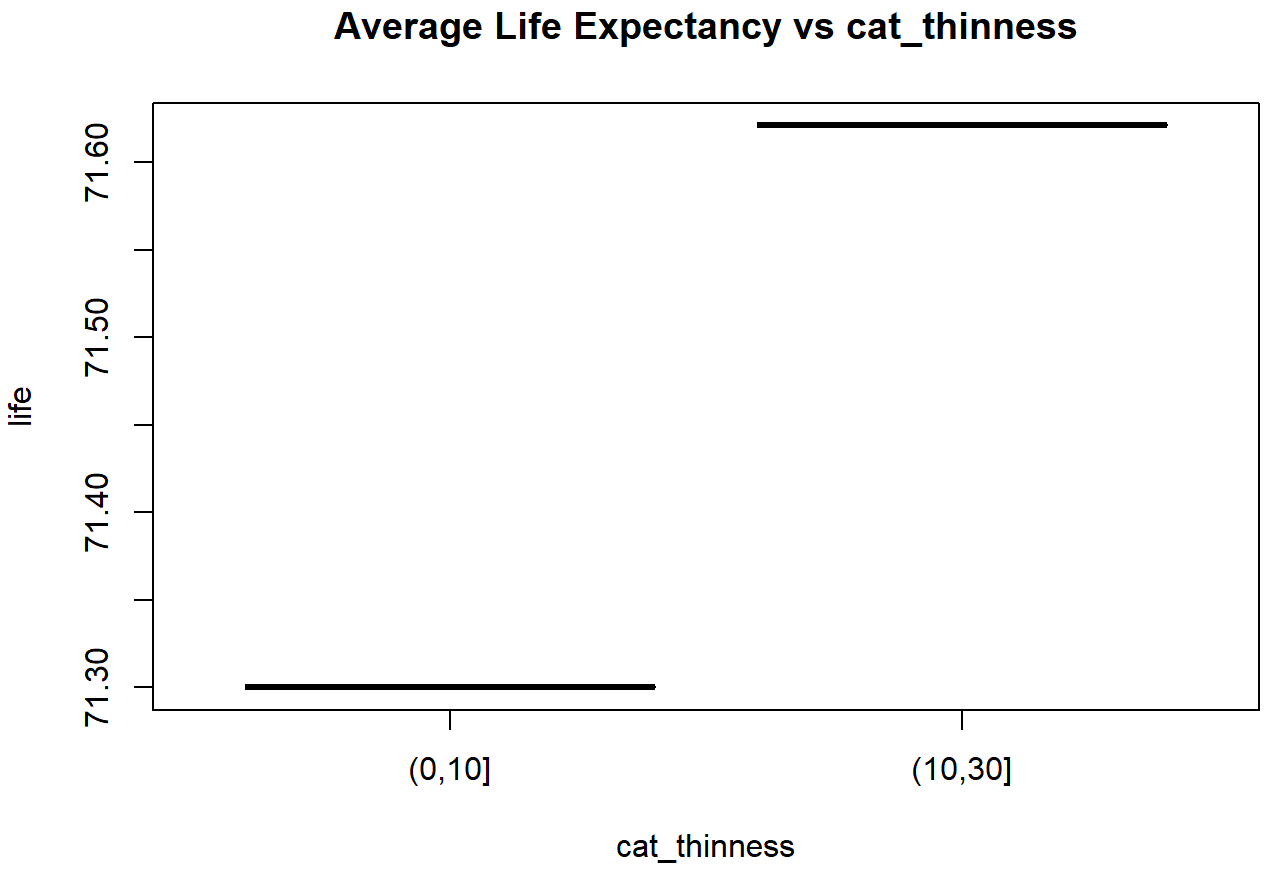
```
model_output(bw)
```


Average Life Expectancy vs log_GDP
life
69
70
71
72
73
74
2
4
6
8
10
log_GDP


Average Life Expectancy vs school
life
60
65
70
75
80
4
6
8
10
12
14
16
school


Average Life Expectancy vs log_hiv
life
70
65
60
55
50
-2
-1
0
1
2
3
4
log_hiv


Average Life Expectancy vs year
life
72.5
72.0
71.5
71.0
2000
2002
2004
2006
2008
2010
2012
2014
year



Interpretation: Model output results suggest that the local linear regression estimator does a much better job of recognizing trends in the data while avoiding overfitting the model. The life expectancy of developing countries no longer fluctuates dramatically, but still captures the change in slope around the midregion of the plot. Although there are slight differences to average life expectancy by school and the log of HIV from prior plots, the trends remain mostly the same. However, we note that while the overall shape of average life expectancy by year is likewise similar to previous models, the dip in life expectancy now occurs around the year 2012 as opposed to 2008.

Point and interval estimates

Point and interval estimate of

$$E(Y|X_1 = x_1, X_2 = x_2, X_3 = x_3, X_4 = x_4, X_5 = x_5)$$

and

$$E(Y|X_1 = x_1, X_2 = x_2, X_3 = x_3 + \Delta_3, X_4 = x_4, X_5 = x_5)$$

when $(x_1, x_2, x_3, x_4, x_5) = (3000, 10.7, 0.1, 2009, (0, 10])$ and $\Delta_3 = 1$:

```
#Set seed
set.seed(20241118)
```

```r
#Local linear regression estimator wild bootstrap function
LLR_wild_boot <- function(B, n, eval_df){
  #Obtain observed residuals
  e = df_who$life - npreg(bw)$mean

  T_n = matrix(NA, nrow = nrow(eval_df), ncol = B)
  for(b in 1:B){
    #Sample observations
    samp = sample(1:nrow(df_who), n, replace = T)
    df_samp = df_who[samp,]

    #Obtain wild boostrap residuals
    tau = sample(c(-1,1), n, replace = T)
    et = e[samp]*tau

    #Obtain simulated life expectancies
    life_sim = npreg(bw, txdat = df_samp[,-1], tydat = df_samp[,1])$mean + et
    df_samp[,1] = life_sim

    #Obtain estimates based on sampled data and wild bootstrap residuals
    life_pred = npreg(bw, txdat = df_samp[,-1], tydat = df_samp[,1],
                      exdat = eval_df[,-1])$mean
    T_n[,b] = life_pred
  }
  return(T_n)
}

#Create data frame of points for which the conditional mean will be estimated
eval_pts <- data.frame(life_pred = NA, log_GDP =rep(log(as.numeric(x[1])), 2),
                       school = rep(x[2], 2),
                       hiv = log(as.numeric(x[3]) + c(0, d)),
                       year = rep(x[4], 2),
                       cat_thinness = rep(x[5], 2))
eval_pts[,2:5] <- lapply(eval_pts[,2:5], as.numeric)
eval_pts[,5:6] <- lapply(eval_pts[,5:6], as.ordered)

#Initialize variables
B = 1000
n = nrow(df_who)

#Perform naive bootstrap
mhat_n = LLR_wild_boot(B, n, eval_pts)
eval_pts[,1] = rowMeans(mhat_n)
eval_pts
```

```
  life_pred  log_GDP school         hiv year cat_thinness
1  70.82389 8.006368   10.7 -2.30258509 2009       (0,10]
2  65.75794 8.006368   10.7  0.09531018 2009       (0,10]
```

```
#Compute and display 95% percentile interval bounds for mhat
bounds = perc_int(mhat_n)
LB = bounds[,1]
UB = bounds[,2]

PI1 = paste("95% CI for GDP = ", x[1], ", school = ", x[2],
            ", hiv = ", x[3], ", year = ", x[4], ", cat_thinness = ",
            x[5], ": (", floor(LB[1]*100)/100, ", ",
            ceiling(UB[1]*100)/100, ")", sep="")
PI2 = paste("95% CI for GDP = ", x[1], ", school = ", x[2], ", hiv = ",
            as.numeric(x[3]) + d, ", year = ", x[4],
            ", cat_thinness = ", x[5], ": (",
            floor(LB[2]*100)/100, ", ",
            ceiling(UB[2]*100)/100, ")", sep="")
print(PI1)
```

```
[1] "95% CI for GDP = 3000, school = 10.7, hiv = 0.1, year = 2009, cat_thinness = (0,10]: (68.17,
73.56)"
```

```
print(PI2)
```

```
[1] "95% CI for GDP = 3000, school = 10.7, hiv = 1.1, year = 2009, cat_thinness = (0,10]: (63.14,
69.54)"
```

Interpretation: This model predicts that the life expectancy of a country with a per-capita GDP of $3000, an average of 10.7 years of schooling, and an HIV prevalence of 0.1 per 1000 people in the year 2009 with 0-10% of the population between ages 5-9 having a low BMI, is 70.82 years with 95% percentile interval bounds of (68.17, 73.56). Increasing the log of HIV prevalence by 1 results in a predicted life expectancy of 65.76 years with a 95% percentile interval of (63.14, 69.54).

Let $(x_1, x_3, x_4) = (3000, 0.1, 2009)$, create a plot that displays a point and interval estimate of:

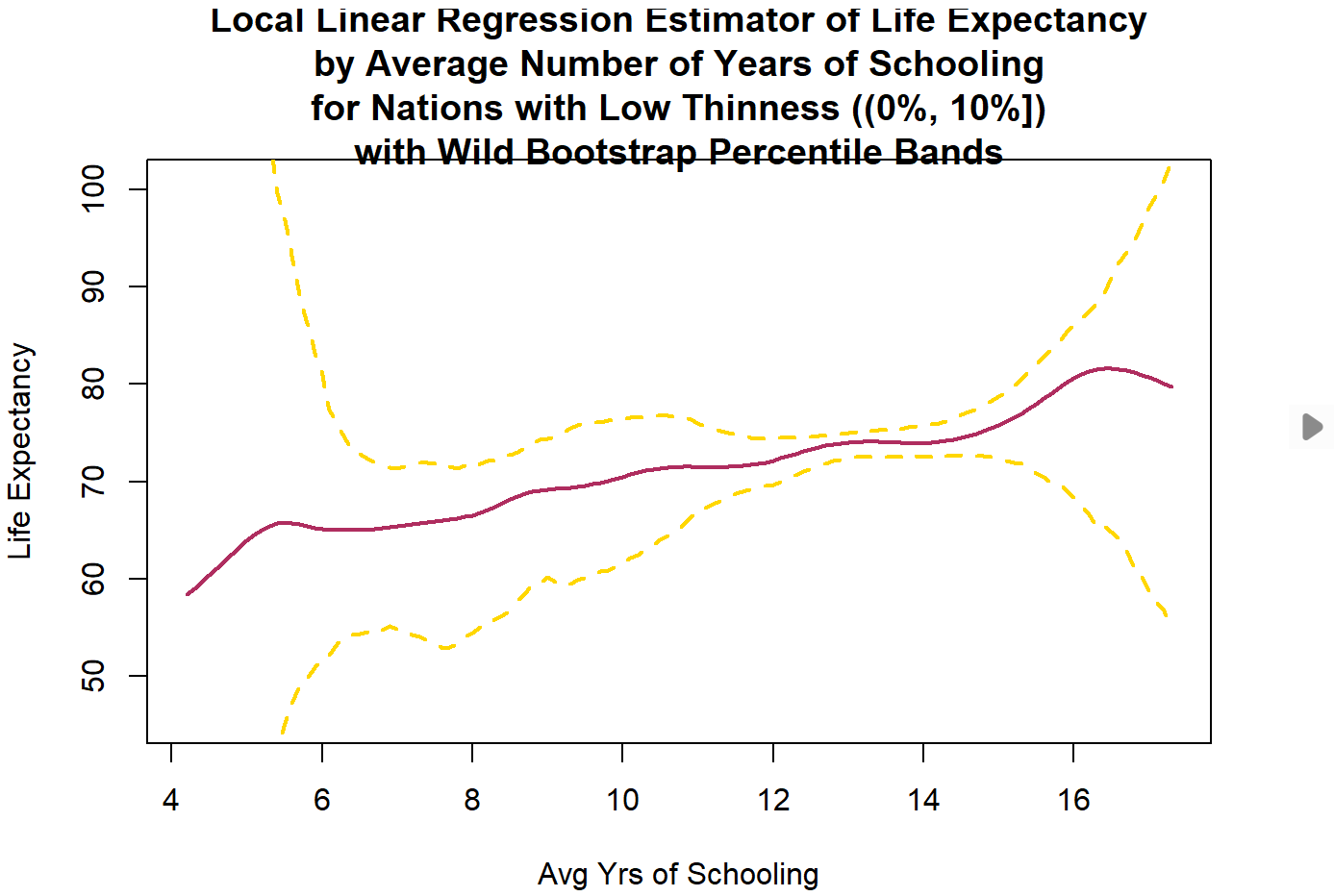
$$x_2 \mapsto E(Y \mid X_1 = x_1, X_2 = x_2, X_3, X_4 = x_4, X_5 \in [0, 10])$$

and

$$x_2 \mapsto E(Y \mid X_1 = x_1, X_2 = x_2, X_3, X_4 = x_4, X_5 \in (10, 30]).$$

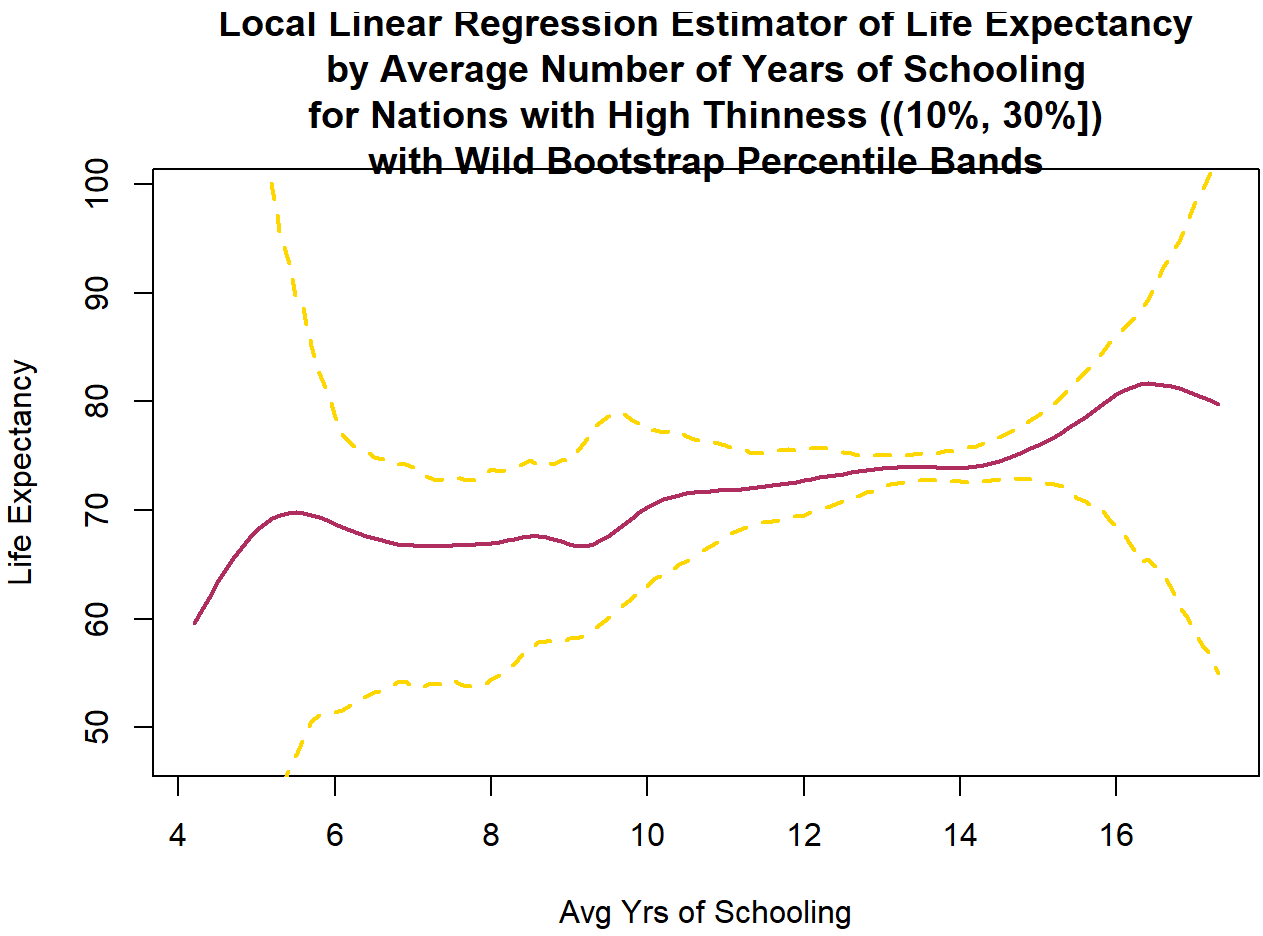
```
#Set seed
set.seed(123)

#Plot mhat vs x2 for fixed x1, x3, and x4 and both levels of x5
x2 = seq(min(df_who$school), max(df_who$school), by = 0.1)
n = length(x2)

#Create data frame of points for which the conditional mean will be estimated
eval_pts <- data.frame(life_pred = NA, GDP = rep(log(as.numeric(x[1])), 2*n),
                       school = rep(x2, 2), hiv = rep(log(as.numeric(x[3])), 2*n),
                       year = rep(x[4], 2*n),
                       cat_thinness = rep(c("(0,10]", "(10,30]"),
```

```
                                        c(n, n)))
eval_pts[,2:5] <- lapply(eval_pts[,2:5], as.numeric)
eval_pts[,5:6] <- lapply(eval_pts[,5:6], as.ordered)

#Perform naive bootstrap
mhat_n = LLR_wild_boot(B, n, eval_pts)

#Compute estimate and 95% confidence bands for life expectancy
eval_pts$life_pred = npreg(bw, exdat = eval_pts[,-1])$mean

bounds = perc_int(mhat_n)
LB = bounds[,1]
UB = bounds[,2]

#Display plots
plot(x2, eval_pts$life_pred[1:n], type = "l", col = "maroon", lwd = 2, xlab = "Avg Yrs of Schooli
lines(x2, LB[1:n], col = "gold", lwd = 2, lty = 2)
lines(x2, UB[1:n], col = "gold", lwd = 2, lty = 2)
```




```
plot(x2, eval_pts$life_pred[(n+1):(2*n)], type = "l", col = "maroon", lwd = 2, xlab = "Avg Yrs of
lines(x2, LB[(n+1):(2*n)], col = "gold", lwd = 2, lty = 2)
```

```
lines(x2, UB[(n+1):(2*n)], col = "gold", lwd = 2, lty = 2)
```



Interpretation: Although the wild bootstrap interval boundaries perform well throughout the majority of the regression curve, the estimated margins of error are quite wide toward the tails. This likely results from the high concentration of data between 10 to 15 years of school, while data is less dense toward the tails leading to more uncertain interval boundaries.

Model 4: Generalized additive model

In a Generalized Additive Model (referred to as a *GAM*), we calculate a separate f_j for each X_j, and then add together their contributions without requiring a specific parametric form. This provides a natural extension to the typical multiple linear regression equation:

$$Y_i = \beta_0 + \beta_1 X_{i,1} + \cdots + \beta_p X_{i,p} + \epsilon_i$$

in order to allow for non-linear relationships (any smooth f_j) between each feature and the response. We express this extension, the GAM, as:

$$Y_i = \beta_0 + f_1(X_{i,1}) + \cdots + f_p(X_{i,p}) + \epsilon_i.$$

Employing GAM's allows us to work with situations where $Y|X$ is not normal as well as consider the effect of *each* X_j individually, holding other variables fixed. In this application we will not be including interaction

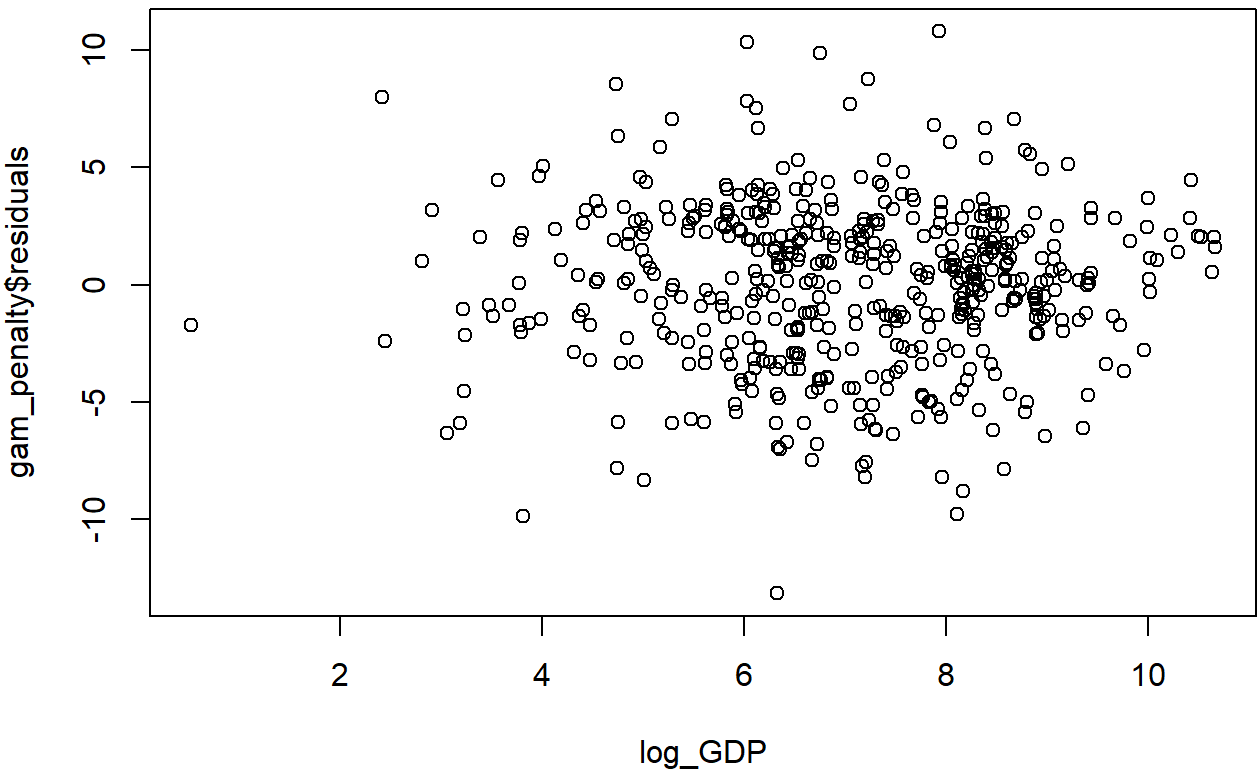
terms.

Assumptions:

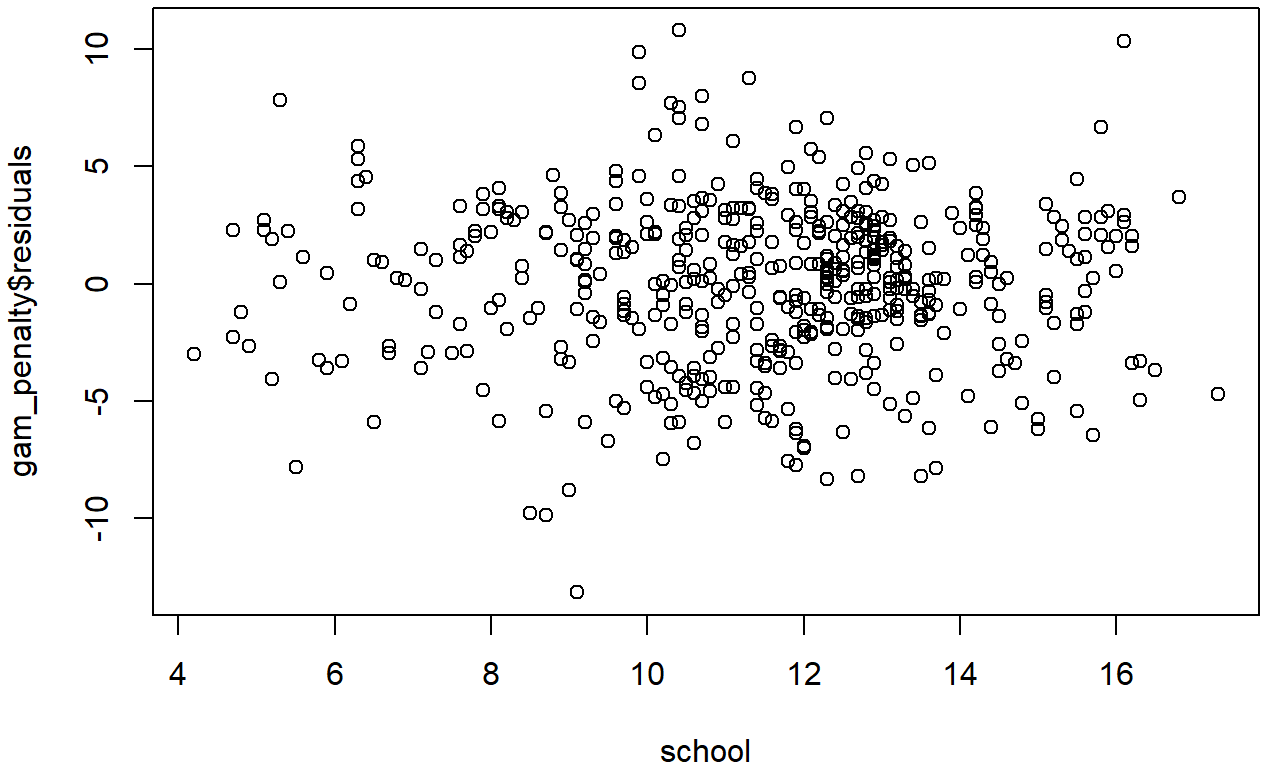
The assumptions of the generalized additive model are:

1. At a point $\vec{X}$, the mean of Y is a combination of functions of $X_1, \ldots, X_p$
2. Functions f_j fitted by the model should be sufficiently smooth
3. Errors in estimating Y are independent and identically Normally distributed with mean 0 and constant variance

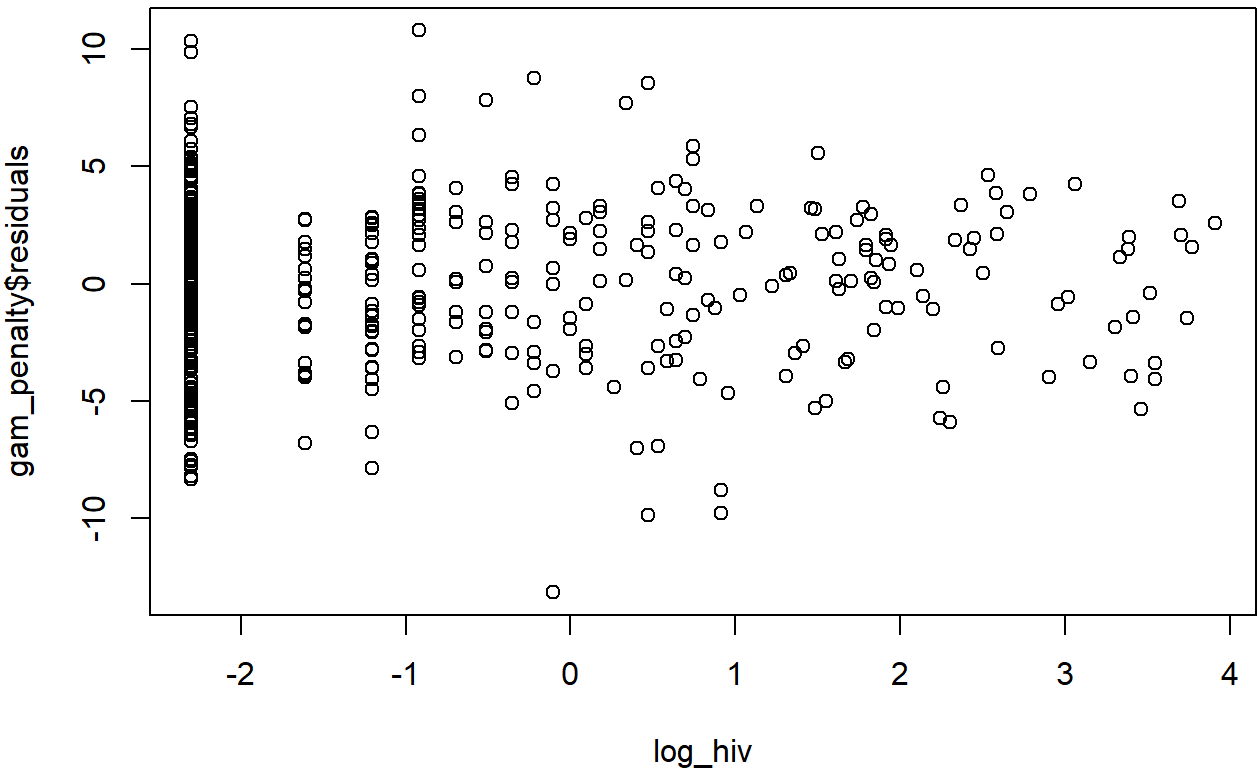
```
attach(df_who)

#GAM model
gam_penalty <- gam(life ~ s(log_GDP) + s(school) + s(log_hiv)
                   + (year) + (cat_thinness), data = df_who,
                   method = "REML", select = TRUE)

#Check residual plots
plot(log_GDP, gam_penalty$residuals)
```



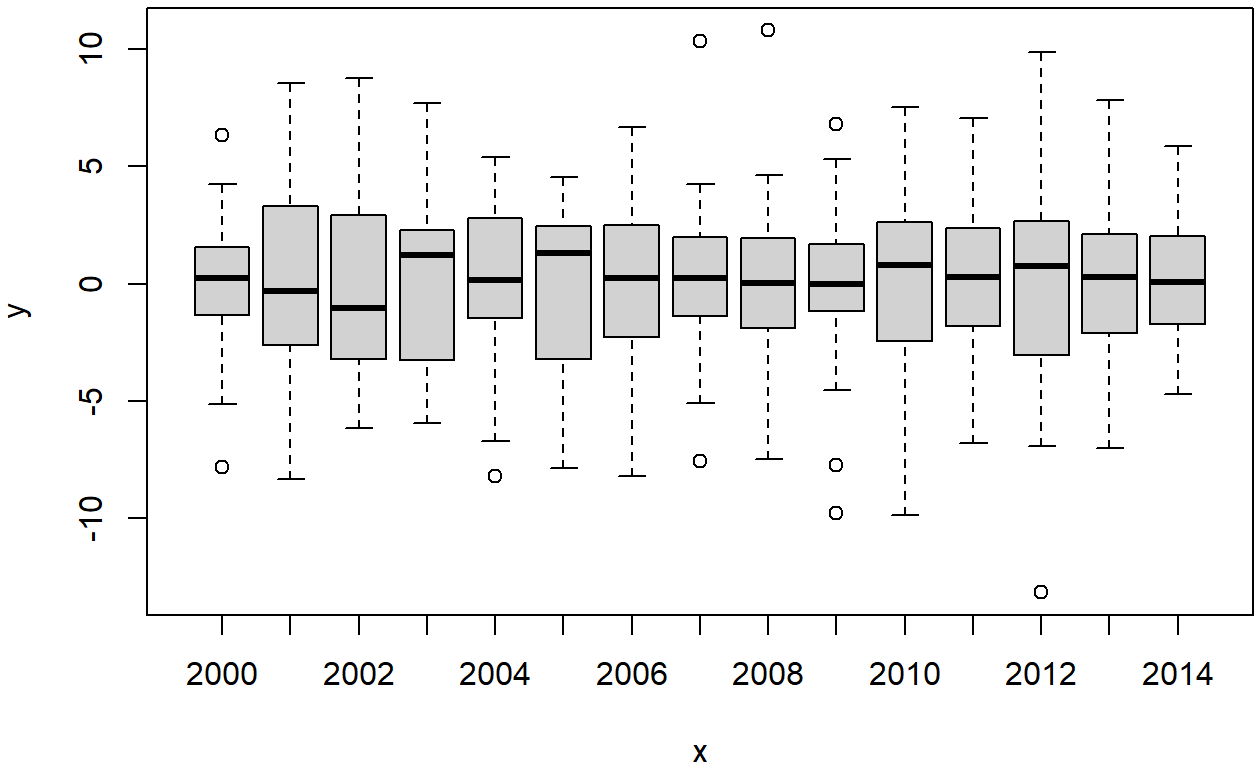
```
plot(school, gam_penalty$residuals)
```



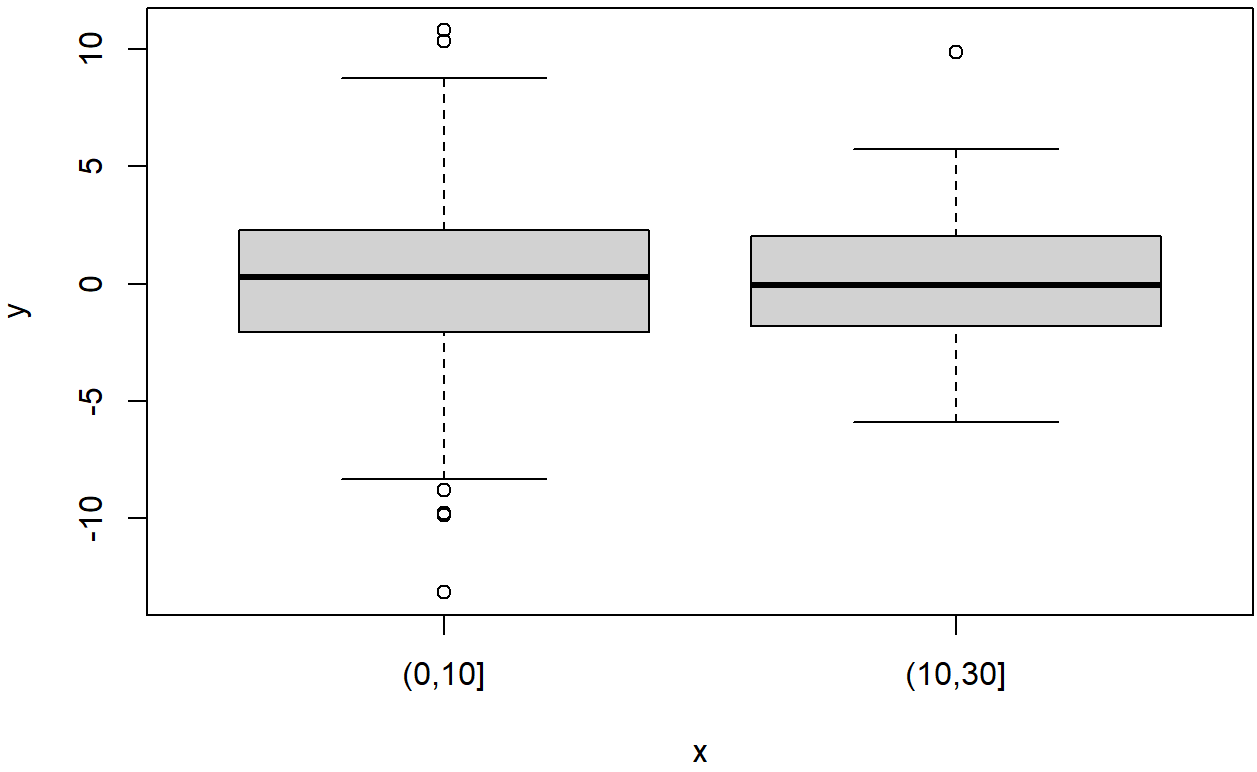

```
plot(log_hiv, gam_penalty$residuals)
```




```
plot(year, gam_penalty$residuals)
```




```
plot(cat_thinness, gam_penalty$residuals)
```




```
#Check if residuals are multivariate normal given values of quantitative predictors
MVN::mvn(data = cbind(gam_penalty$residuals, df_who[,2:4]))$multivariateNormality
```

```
           Test       HZ p value MVN
1 Henze-Zirkler 7.280789       0  NO
```

```
#Check if residuals are multivariate normal given levels of categorical predictors
for(i in levels(year)){
  LF = lillie.test(gam_penalty$residuals[which(year==i)])
  print(paste("Lilliefors test of residuals for year", i, "p-value =", LF$p.value))
}
```

```
[1] "Lilliefors test of residuals for year 2000 p-value = 0.512352412632531"
[1] "Lilliefors test of residuals for year 2001 p-value = 0.884382908768735"
[1] "Lilliefors test of residuals for year 2002 p-value = 0.481763300650393"
[1] "Lilliefors test of residuals for year 2003 p-value = 0.0318786062047904"
[1] "Lilliefors test of residuals for year 2004 p-value = 0.0715116064594453"
[1] "Lilliefors test of residuals for year 2005 p-value = 0.0280302394270183"
[1] "Lilliefors test of residuals for year 2006 p-value = 0.695983654249774"
[1] "Lilliefors test of residuals for year 2007 p-value = 0.339918057312396"
[1] "Lilliefors test of residuals for year 2008 p-value = 0.600374591799311"
[1] "Lilliefors test of residuals for year 2009 p-value = 0.0240401552463672"
[1] "Lilliefors test of residuals for year 2010 p-value = 0.0785548908623231"
```

```
[1] "Lilliefors test of residuals for year 2011 p-value = 0.712970490164359"
[1] "Lilliefors test of residuals for year 2012 p-value = 0.0944877284883876"
[1] "Lilliefors test of residuals for year 2013 p-value = 0.733077132156978"
[1] "Lilliefors test of residuals for year 2014 p-value = 0.761436793595373"
```

```
for(i in levels(cat_thinness)){
  LF = lillie.test(gam_penalty$residuals[which(cat_thinness==i)])
  print(paste("Lilliefors test of residuals for cat_thinness", i, "p-value =",
              LF$p.value))
}
```

```
[1] "Lilliefors test of residuals for cat_thinness (0,10] p-value = 0.00148014207087106"
[1] "Lilliefors test of residuals for cat_thinness (10,30] p-value = 0.430889725505328"
```

```
detach(df_who)
```

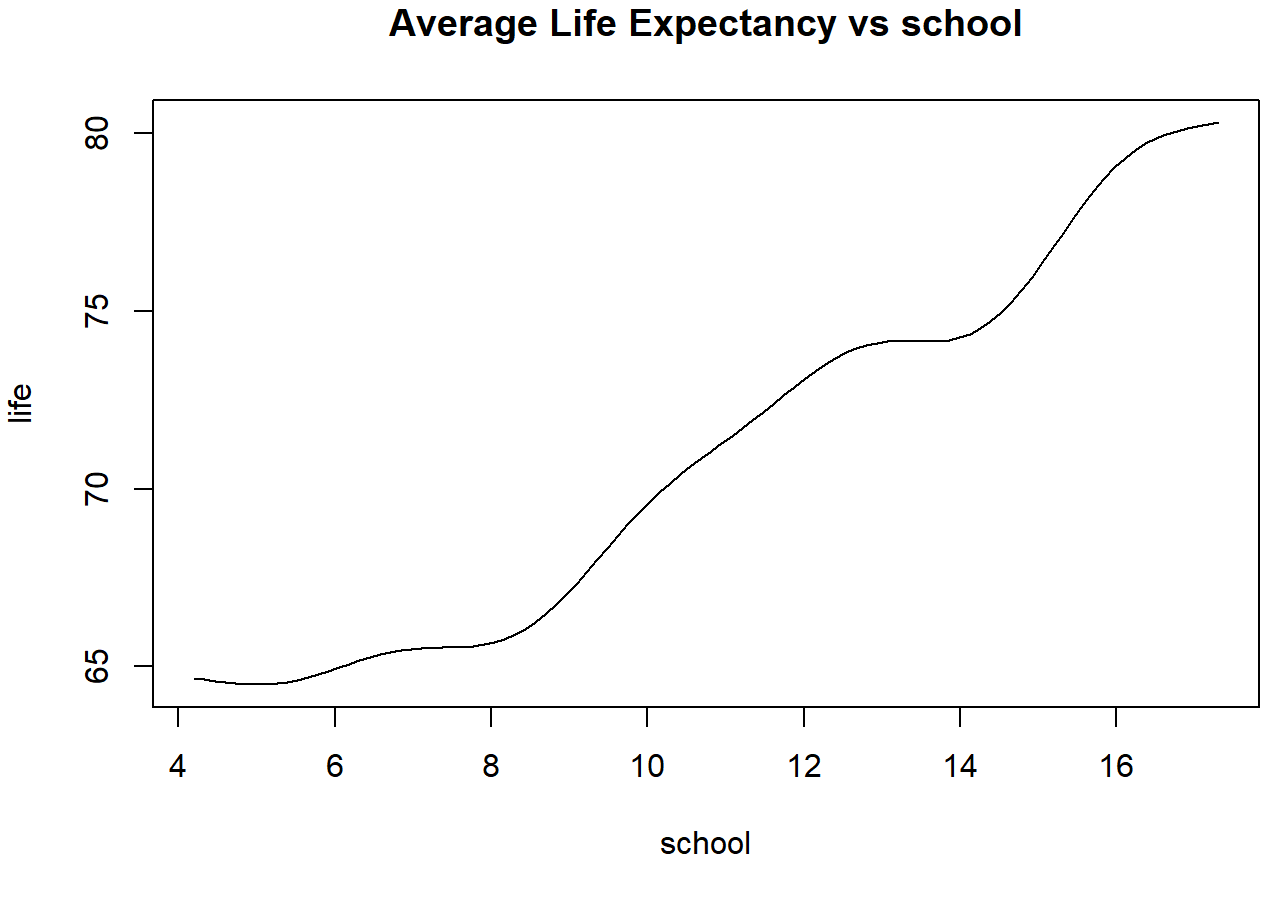
Model assumptions appear to be reasonably satisfied, as residual plots suggest relatively constant variance toward error terms and results of the Lilliefors normality tests are mostly insignificant indicating that predictors, for the most part, can be assumed to be normal. Smoothness of the functions f_j are guaranteed by the packages utilized for analysis.

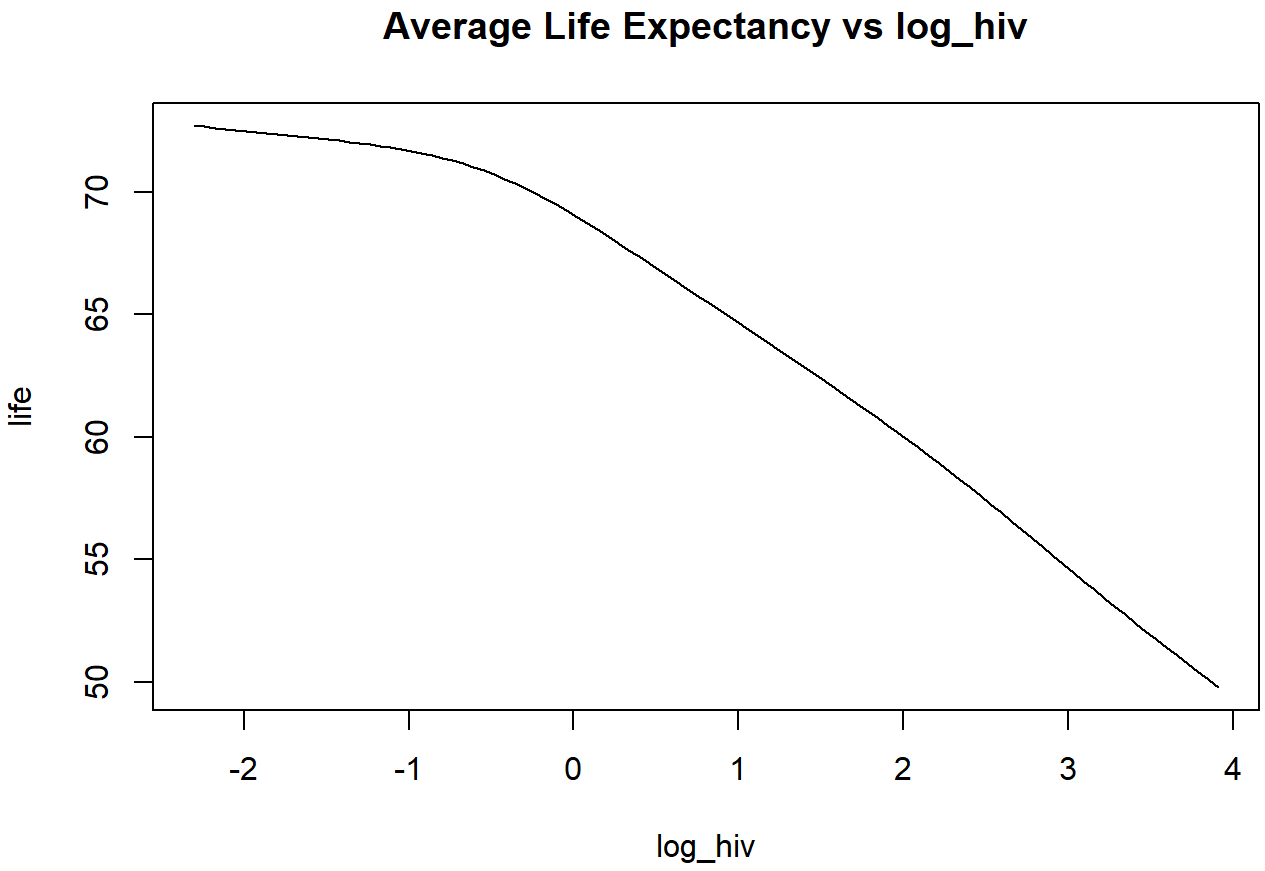
Models output and interpretation

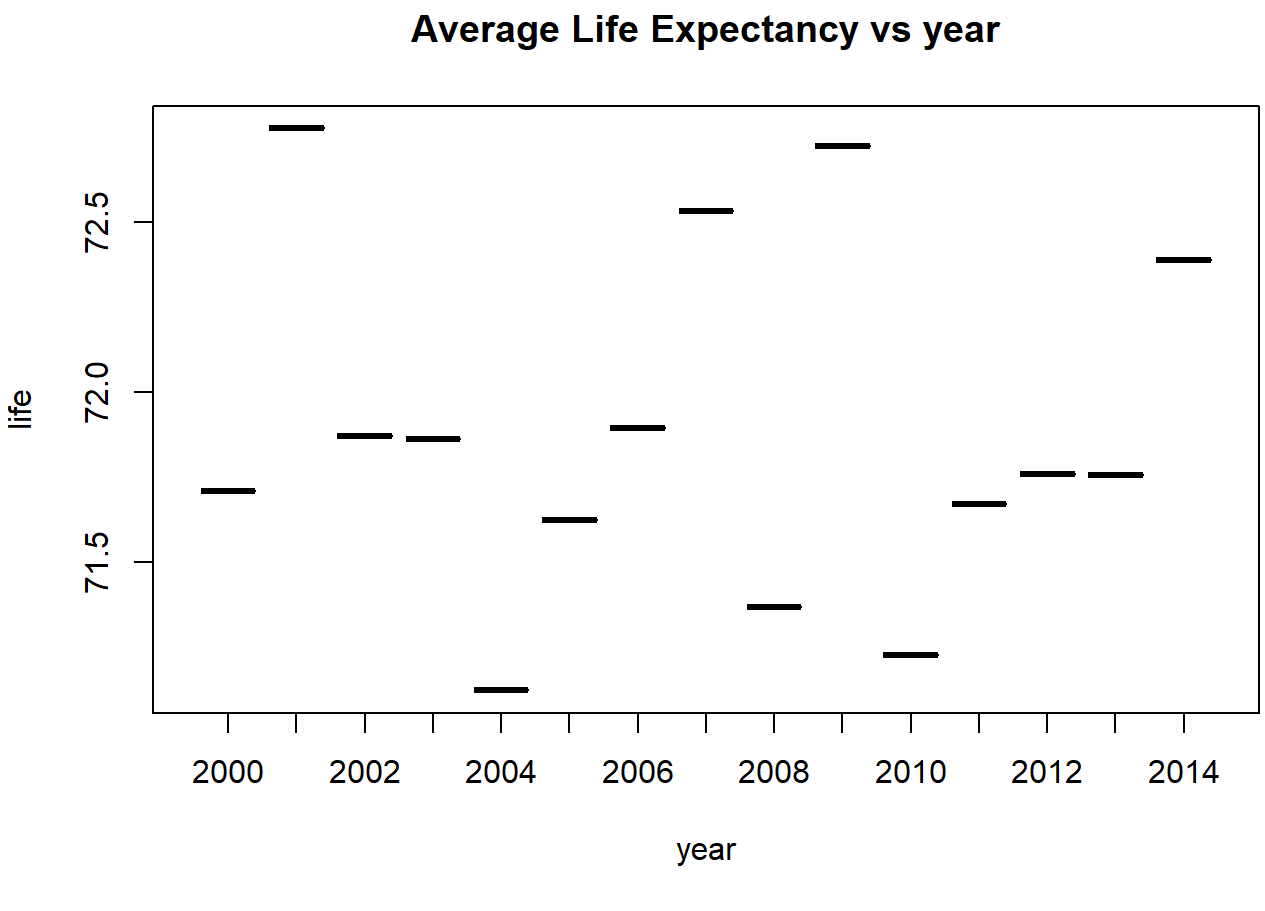
```
#Fix quantitative predictors to their medians, and categorical predictors to the most frequent ca
x = c(median(df_who$log_GDP), median(df_who$school), median(df_who$log_hiv),
      names(sort(table(df_who$year), decreasing = T))[1],
      names(sort(table(df_who$cat_thinness), decreasing = T))[1])

#Plot mhat vs x[i], i = 1, ..., 5, with other x's fixed
for(i in 2:6){
  #For quantitative x[i], plot a curve
  if(class(df_who[,i])[1] == "numeric"){
    n = 100
    x_seq = seq(min(df_who[,i]), max(df_who[,i]),
                length.out = n)
    eval_pts = matrix(c(rep(NA, n), rep(x[1], n), rep(x[2], n),
                        rep(x[3], n), rep(x[4], n), rep(x[5], n)),
                      nrow = n)
    colnames(eval_pts) <- names(df_who)
    eval_pts[,i] = x_seq
    eval_pts <- as.data.frame(eval_pts)
    eval_pts[,2:4] <- lapply(eval_pts[,2:4], as.numeric)
    eval_pts[,5:6] <- lapply(eval_pts[,5:6], as.ordered)

    gampred = predict(gam_penalty, newdata = eval_pts[,-1], se.fit = TRUE)
    eval_pts[,1] = gampred$fit

    xlab = names(df_who)[i]
```

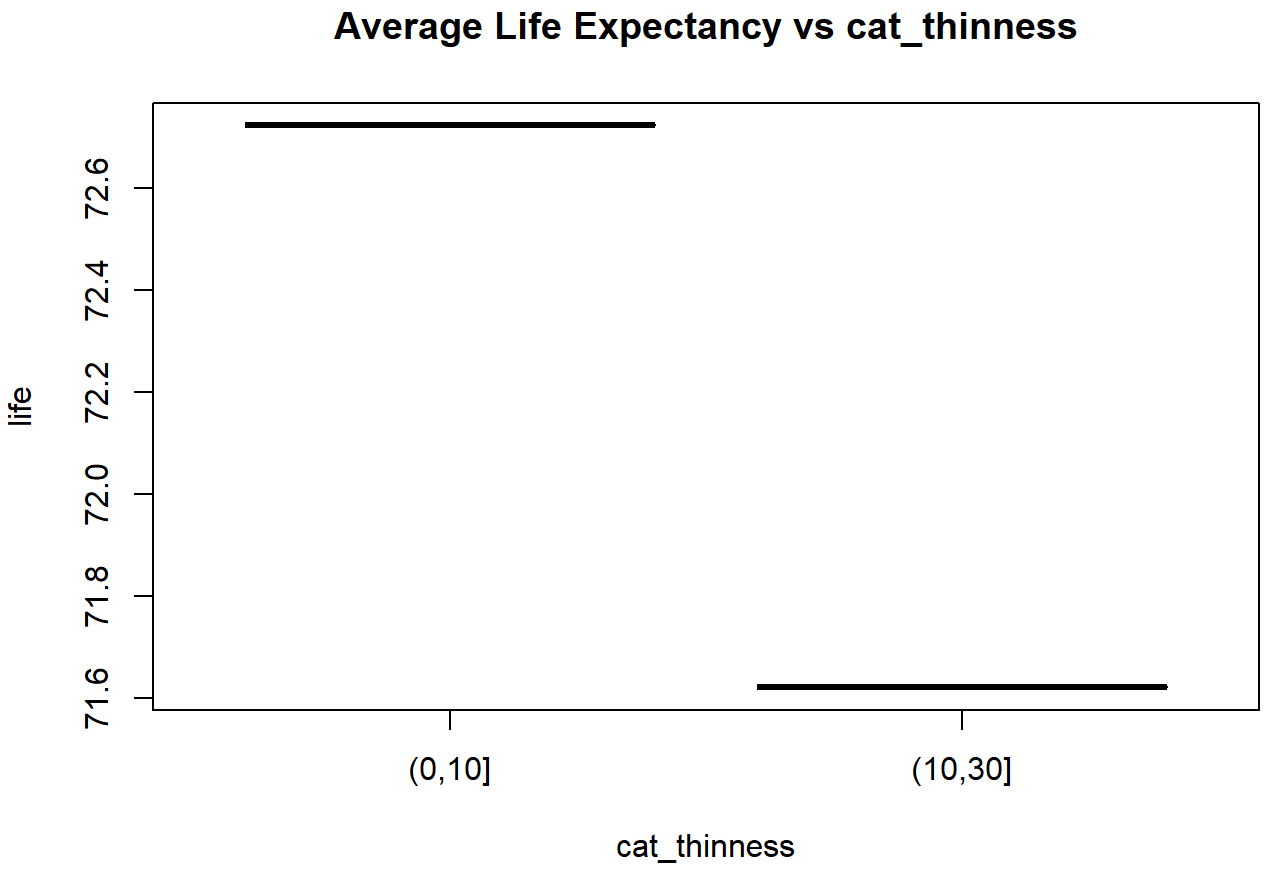
```
    title = paste("Average Life Expectancy vs", xlab)
    plot(eval_pts[,i], eval_pts$life, xlab = xlab, ylab = "life", main = title,
         type = "l")

  #For categorical x[i], plot points
  }else{
    lvls = levels(df_who[,i])
    n = length(lvls)
    eval_pts = matrix(c(rep(NA, n), rep(x[1], n), rep(x[2], n),
                        rep(x[3], n), rep(x[4], n), rep(x[5], n)),
    nrow = n)
    colnames(eval_pts) <- names(df_who)
    eval_pts[,i] = lvls
    eval_pts <- as.data.frame(eval_pts)
    eval_pts[,2:4] <- lapply(eval_pts[,2:4], as.numeric)
    eval_pts[,5:6] <- lapply(eval_pts[,5:6], as.ordered)

    gampred = predict(gam_penalty, newdata = eval_pts[,-1], se.fit = TRUE)
    eval_pts[,1] = gampred$fit

    xlab = names(df_who)[i]
    title = paste("Average Life Expectancy vs", xlab)
    plot(eval_pts[,i], eval_pts$life, xlab = xlab, ylab = "life", main = title,
         type = "p")
  }
}
```


Average Life Expectancy vs log_GDP
life
70
71
72
73
74
2
4
6
8
10
log_GDP


Average Life Expectancy vs school
life
65
70
75
80
4
6
8
10
12
14
16
school


Average Life Expectancy vs log_hiv
life
50
55
60
65
70
-2
-1
0
1
2
3
4
log_hiv


Average Life Expectancy vs year
life
71.5
72.0
72.5
2000
2002
2004
2006
2008
2010
2012
2014
year



Interpretation: For the generalized additive model estimates, we can see that although life expectancy of developing countries still has a positive relationship with the log of GDP, this relationship is now strictly linear, suggesting that prior models could have been slightly overfitting this variable. Similarly, the average life expectancy by school still has a curvilinear relationship but changes in slope are less pronounced. Average life expectancy by year is the most different from other models, suggesting that the relationship between these variables is sporadic and not necessarily captured well by a regression curve. This conclusion is in line with prior model assessments; despite previous models suggesting remotely similar shapes for this pair of variables, the qualities of the curves produced throughout prior analysis had remarkably different qualities; for example, where the minimum fell and if the slope changed sign from positive to negative toward the end region of the plot around 2012-2014.

Point and interval estimates:

Point and interval estimate of

$$E(Y|X_1 = x_1, X_2 = x_2, X_3 = x_3, X_4 = x_4, X_5 = x_5)$$

and

$$E(Y|X_1 = x_1, X_2 = x_2, X_3 = x_3 + \Delta_3, X_4 = x_4, X_5 = x_5)$$

when $(x_1, x_2, x_3, x_4, x_5) = (3000, 10.7, 0.1, 2009, (0, 10])$ and $\Delta_3 = 1$:

```r
# combination of (x1:x5)
x = c(3000, 10.7, 0.1, 2009, "(0,10]")

eval_pts = data.frame(life_pred = NA, log_GDP = rep(log(as.numeric(x[1])), 2),
                      school = rep(x[2], 2),
                      log_hiv = log(as.numeric(x[3]) + c(0, d)),
                      year = rep(x[4], 2), cat_thinness = rep(x[5], 2))
eval_pts[,2:5] <- lapply(eval_pts[,2:5], as.numeric)
eval_pts[,5:6] <- lapply(eval_pts[,5:6], as.ordered)

# predict using penalized GAM
gampred <- predict(gam_penalty, newdata = eval_pts[,-1], se.fit = TRUE)
eval_pts[,1] = gampred$fit
eval_pts
```

```
  life_pred  log_GDP school     log_hiv year cat_thinness
1  71.23645 8.006368   10.7 -2.30258509 2009       (0,10]
2  67.19971 8.006368   10.7  0.09531018 2009       (0,10]
```

```r
LB = gampred$fit - z * gampred$se.fit
UB = gampred$fit + z * gampred$se.fit

# point estimate and confidence intervals
#cat("Point Estimate:", gampred$fit, "\n")
#cat("95% Interval Estimate: (", gampred$fit - z * gampred$se.fit, ",",  gampred$fit + z * gampre
CI1 = paste("95% CI for GDP = ", x[1], ", school = ", x[2],
            ", hiv = ", x[3], ", year = ", x[4], ", cat_thinness = ",
            x[5], ": (", floor(LB[1]*100)/100, ", ",
            ceiling(UB[1]*100)/100, ")", sep = "")
CI2 = paste("95% CI for GDP = ", x[1], ", school = ", x[2], ", hiv = ",
            as.numeric(x[3]) + d, ", year = ", x[4],
            ", cat_thinness = ", x[5], ": (",
            floor(LB[2]*100)/100, ", ",
            ceiling(UB[2]*100)/100, ")", sep = "")
print(CI1)
```

```
[1] "95% CI for GDP = 3000, school = 10.7, hiv = 0.1, year = 2009, cat_thinness = (0,10]: (69.93, 72.54)"
```

```r
print(CI2)
```

```
[1] "95% CI for GDP = 3000, school = 10.7, hiv = 1.1, year = 2009, cat_thinness = (0,10]: (65.75, 68.65)"
```

Interpretation:

This model suggests that the life expectancy of a country with a per-capita GDP of $3000, an average of 10.7 years of schooling, and an HIV prevalence of 0.1 per 1000 people in the year 2009 with 0-10% of the population between ages 5-9 having a low BMI, is 71.24 years with 95% confidence interval bounds of (69.93, 72.54). Increasing the log of HIV prevalence by 1 results in a predicted life expectancy of 67.2 years with a 95% confidence interval of (65.75, 68.65). The interval estimate implies that if we sampled 100 developing countries in 2005 with the above observed variables, 95 would have life expediencies at birth between 65.75 and 68.65 years.

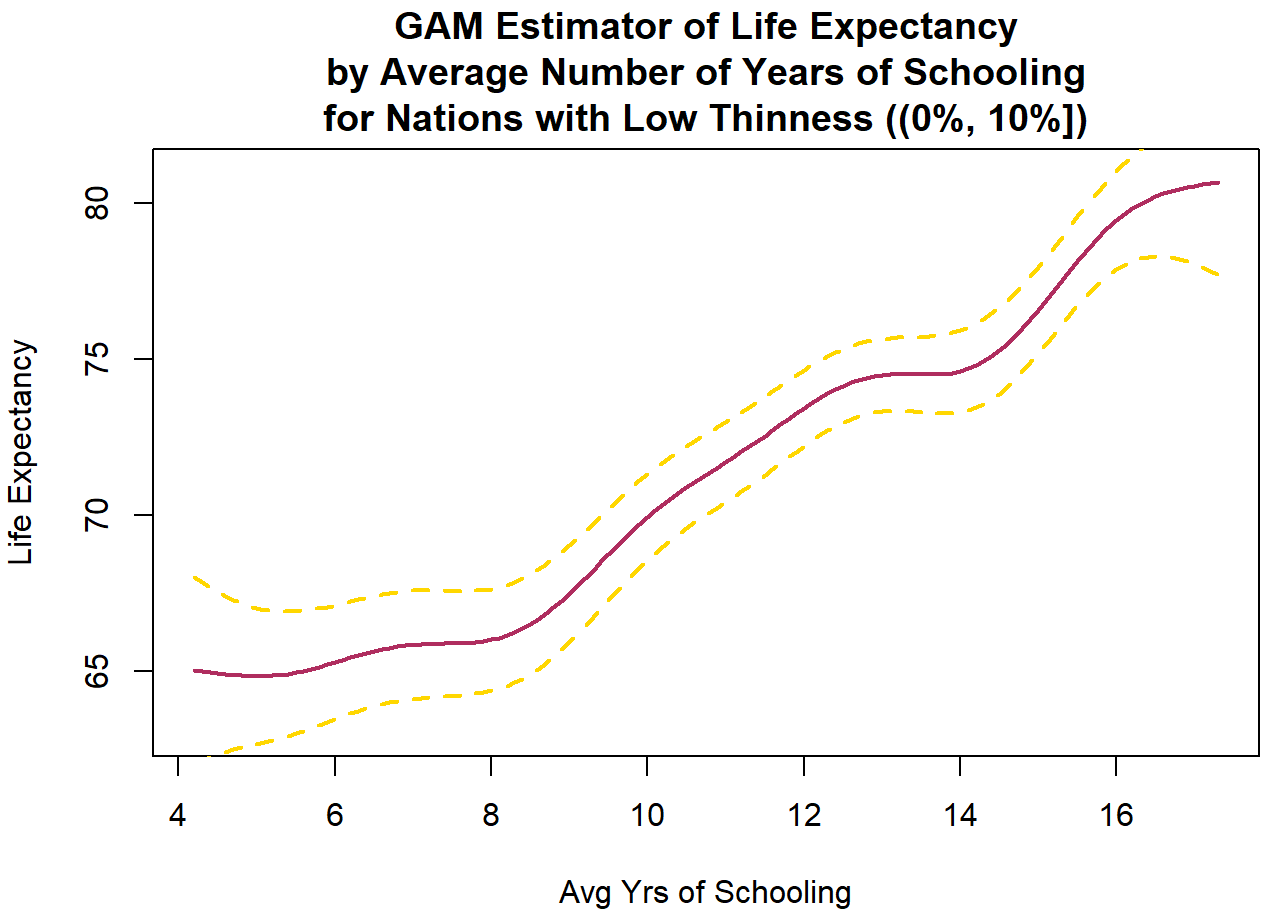
Let $(x_1, x_3, x_4) = (3000, 2, 2005)$, create a plot that displays a point and interval estimate of:

$$x_2 \mapsto E(Y \mid X_1 = x_1, X_2 = x_2, X_3, X_4 = x_4, X_5 \in [0, 10])$$

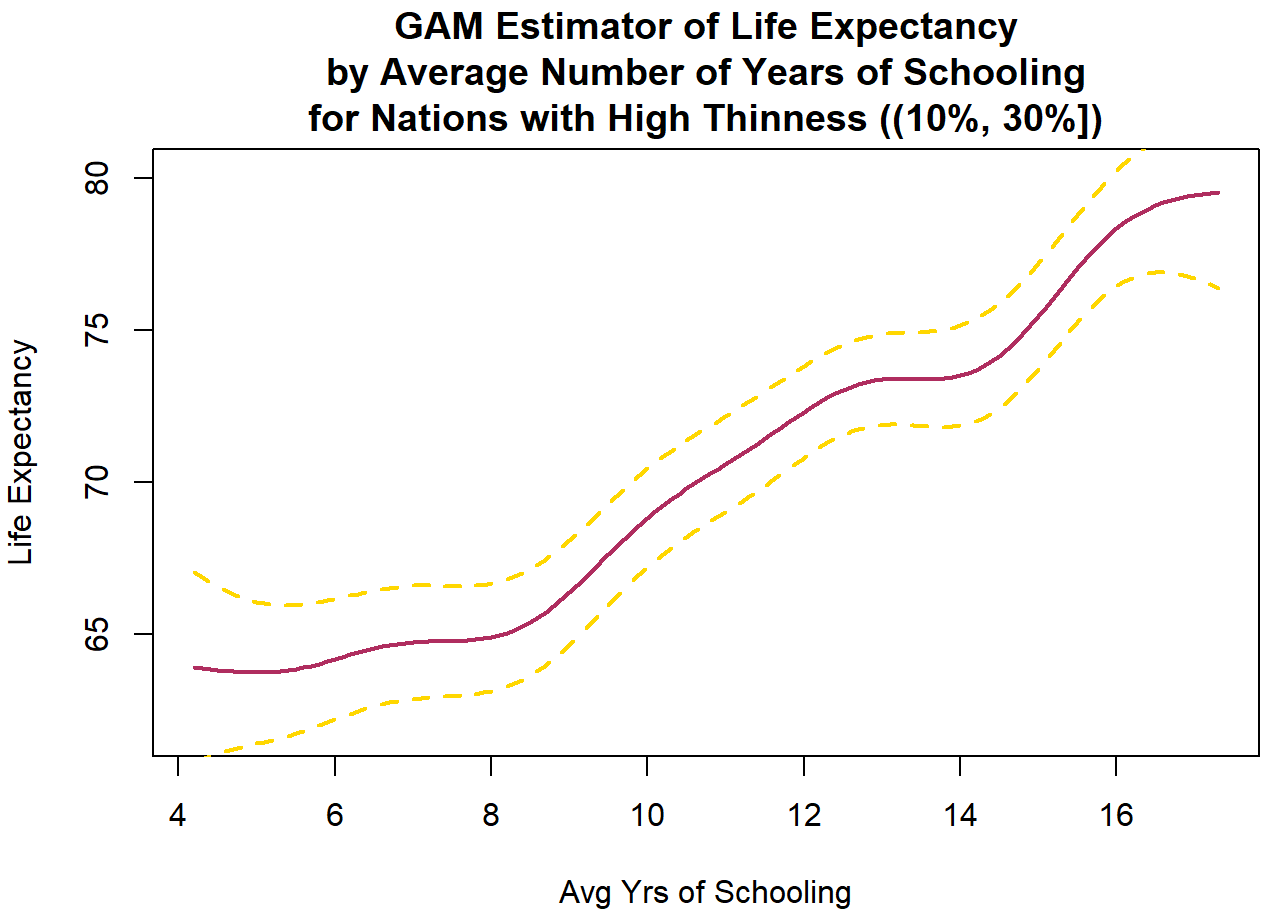
and

$$x_2 \mapsto E(Y \mid X_1 = x_1, X_2 = x_2, X_3, X_4 = x_4, X_5 \in (10, 30]).$$

```
#Plot mhat vs x2 for fixed x1, x3, and x4 and both levels of x5
x2 = seq(min(df_who$school), max(df_who$school), by = 0.1)
n = length(x2)

#Create data frame of points for which the conditional mean will be estimated
eval_pts <- data.frame(life_pred = NA, log_GDP = rep(log(as.numeric(x[1])), 2*n),
                       school = rep(x2, 2), log_hiv = rep(log(as.numeric(x[3])), 2*n),
                       year = rep(x[4], 2*n),
                       cat_thinness = rep(c("(0,10]", "(10,30]"),
                                          c(n, n)))
eval_pts[,2:5] <- lapply(eval_pts[,2:5], as.numeric)
eval_pts[,5:6] <- lapply(eval_pts[,5:6], as.ordered)

#Compute estimate and 95% confidence bands for life expectancy
gampred <- predict(gam_penalty, newdata = eval_pts[,-1], se.fit = TRUE)
eval_pts$life_pred = gampred$fit

LB = gampred$fit - z * gampred$se.fit
UB = gampred$fit + z * gampred$se.fit

#Display plots
plot(x2, eval_pts$life_pred[1:n], type = "l", col = "maroon", lwd = 2, xlab = "Avg Yrs of Schooli
lines(x2, LB[1:n], col = "gold", lwd = 2, lty = 2)
lines(x2, UB[1:n], col = "gold", lwd = 2, lty = 2)
```




```
plot(x2, eval_pts$life_pred[(n+1):(2*n)], type = "l", col = "maroon", lwd = 2, xlab = "Avg Yrs of
lines(x2, LB[(n+1):(2*n)], col = "gold", lwd = 2, lty = 2)
lines(x2, UB[(n+1):(2*n)], col = "gold", lwd = 2, lty = 2)
```



Interpretation:

The first plot illustrates that for developing countries in 2009 with:

- Gross Domestic Product per capita is $3000
- 0.1 HIV cases per 1,000 people
- ***a thinness indicator (low BMI in children 5-9 years of age) of (0,10]***

We can expect the predicted life expectancy at birth to increase from approximately 65 to approximately 80 years as average years in schooling increase within its observed range. The rate of increase can be seen in the plot. The plot also displays a 95% confidence band about the plot.

The second plot illustrates that for developing countries in 2009 with:

- Gross Domestic Product per capita is $3000
- 0.1 HIV cases per 1,000 people
- ***a thinness indicator (low BMI in children 5-9 years of age) of (10,30]***

We can expect the predicted life expectancy at birth to increase from approximately 62 to approximately 80 years as average years in schooling increase within its observed range. This plot shows a 95% confidence band as well.