a2

Xinyi Cui

10/17/2021

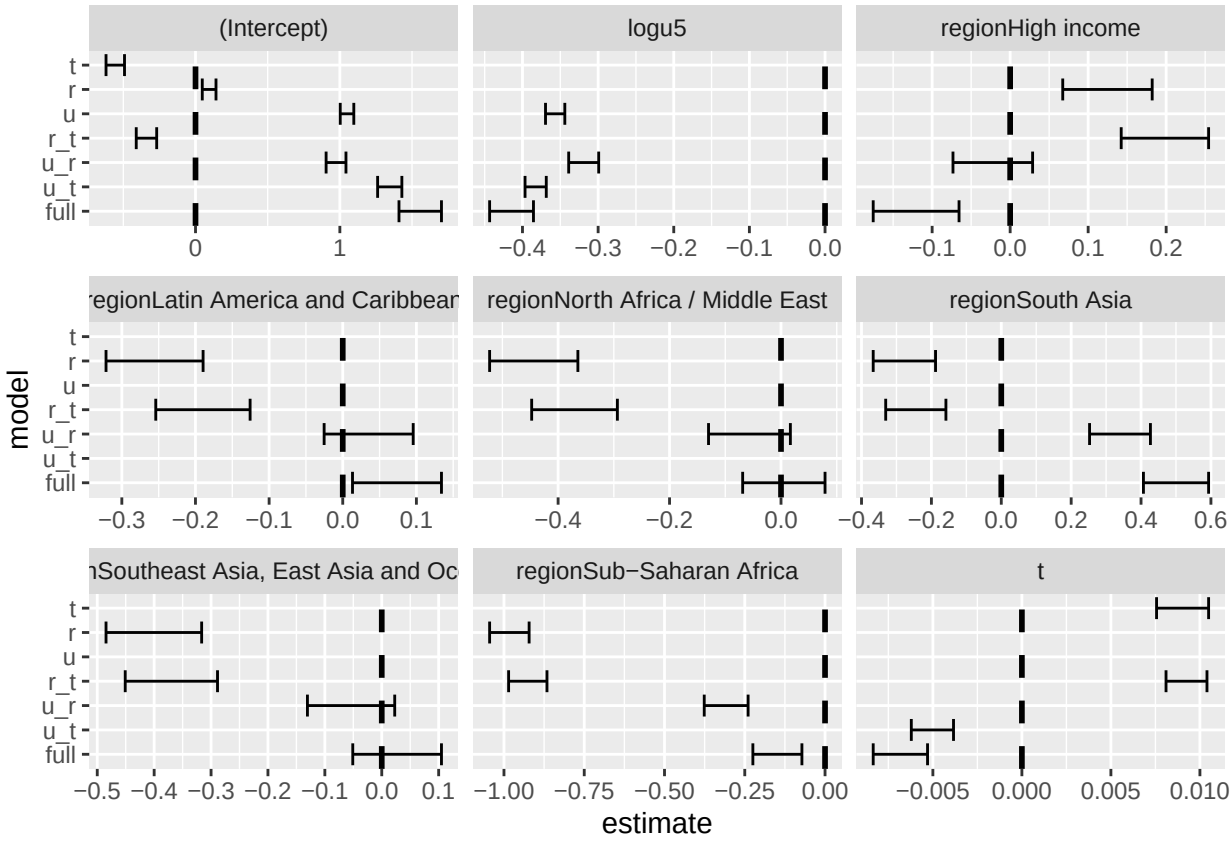
```
knitr::opts_chunk$set(echo = TRUE, message = FALSE, warning = FALSE)
```

```
library(printr)
library(tidyverse)
library(tidymodels)
library(broom)
library(splines)
library(dagitty)
library(ggdag)
```

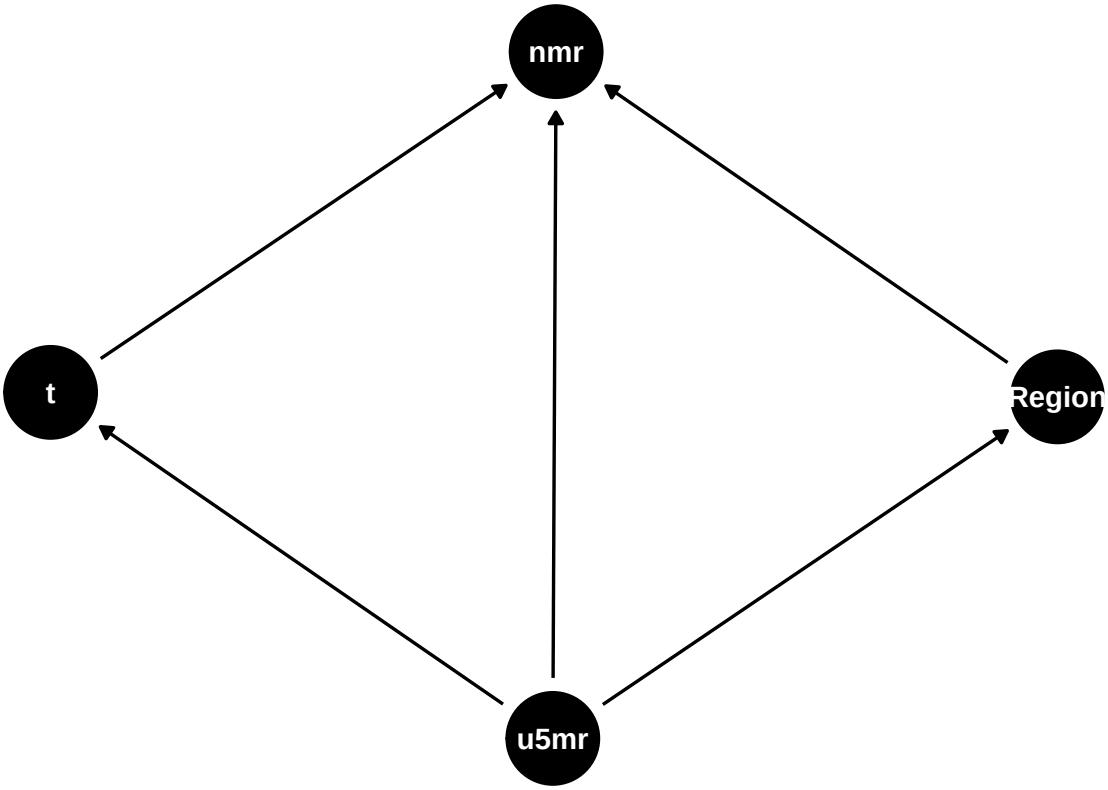
```
A2 <- read_csv("neonatal_mortality.csv") %>%
  mutate(r = nmr/(u5mr - nmr),
         logr = log(r),
         t = year - min(year),
         logu5 = log(u5mr)) %>%
  select(-c(year,r, u5mr))
```

```
# Q1 part1
A2_split <- initial_split(A2, strate = region)
A2_training <- training(A2_split)
A2_test <- testing(A2_split)

# choice of variables
fit_1 <- lm(logr ~ logu5 + region + t, A2_training) %>%
  tidy() %>%
  mutate(model = "full")

fit_2 <- lm(logr ~ logu5 + t, A2_training) %>%
  tidy() %>%
  mutate(model = "u_t")

fit_3 <- lm(logr ~ logu5 + region, A2_training) %>%
  tidy() %>%
  mutate(model = "u_r")

fit_4 <- lm(logr ~ region + t, A2_training) %>%
  tidy() %>%
  mutate(model = "r_t")

fit_5 <- lm(logr ~ logu5, A2_training) %>%
  tidy() %>%
```

```
  mutate(model = "u")

fit_6 <- lm(logr ~ region, A2_training) %>%
  tidy() %>%
  mutate(model = "r")

fit_7 <- lm(logr ~ t, A2_training) %>%
  tidy() %>%
  mutate(model = "t")

# full model
all_model <- fit_1 %>%
  full_join(fit_2) %>%
  full_join(fit_3) %>%
  full_join(fit_4) %>%
  full_join(fit_5) %>%
  full_join(fit_6) %>%
  full_join(fit_7) %>%
  mutate(lower = estimate - 1.96*std.error,
         upper = estimate + 1.96*std.error)

all_model %>% ggplot(aes(estimate, y = model)) +
  geom_errorbar(aes(xmin = lower, xmax = upper)) +
  geom_vline(aes(xintercept = 0),
             linetype = "dashed",
             size = 1) +
  facet_wrap(.~term, scale = "free_x") +
  scale_y_discrete(limits = c("full", "u_t", "u_r", "r_t", "u", "r", "t"))
```




```
# relationship between each variable
dagify(nmr ~ t,
       nmr ~ Region,
       nmr ~ u5mr,
       Region ~ u5mr,
       t ~ u5mr) %>%
  ggdag()+
  theme_dag()
```



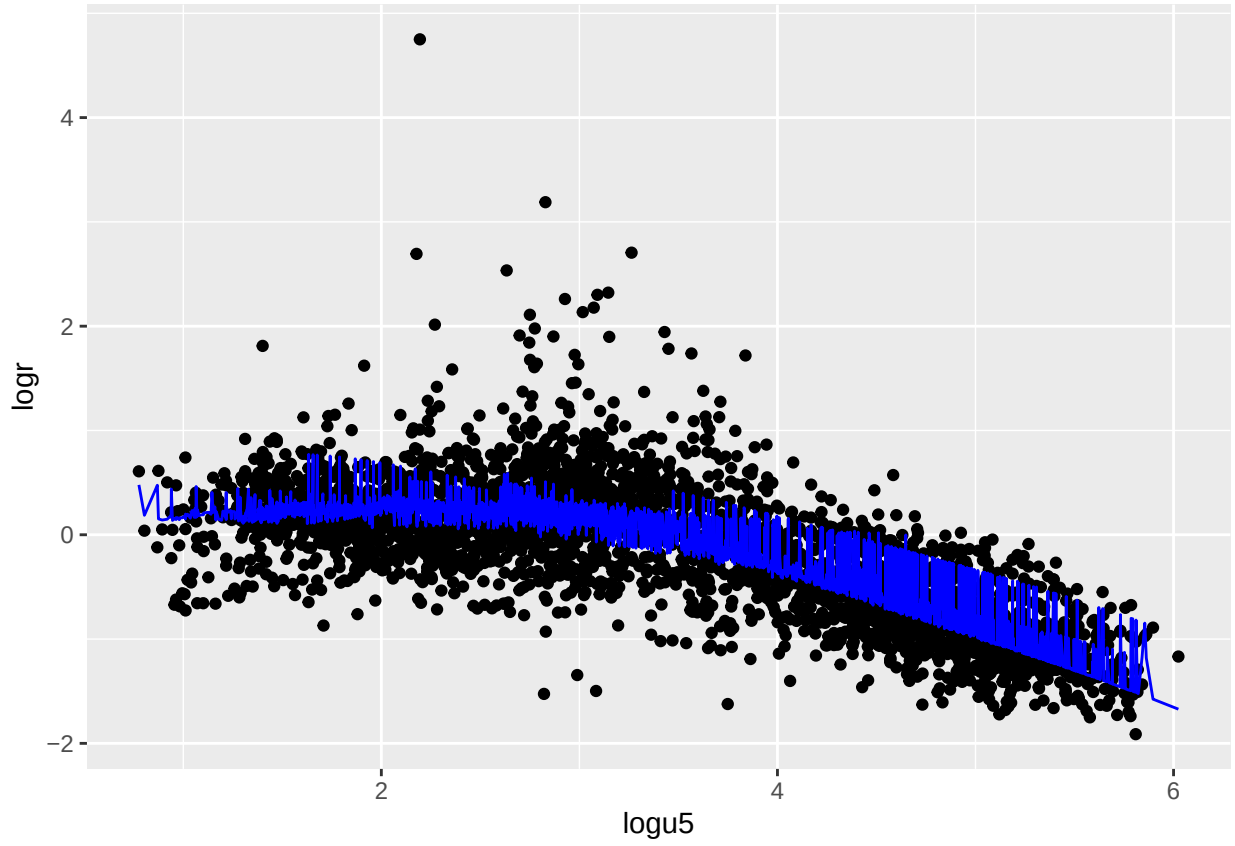

```
# Final model
fit0 <- lm(logr ~ logu5 + logu5*region + logu5*t, A2_training)
summary(fit0)
```

```
##
## Call:
## lm(formula = logr ~ logu5 + logu5 * region + logu5 * t, data = A2_training)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7281 -0.2322 -0.0192  0.2011  4.5396
##
## Coefficients:
##                                                 Estimate Std. Error t value
## (Intercept)                                    1.4304269  0.1337832  10.692
## logu5                                         -0.3796656  0.0366860 -10.349
## regionHigh income                             -0.4543528  0.0788388  -5.763
## regionLatin America and Caribbean              0.9379178  0.1100972   8.519
## regionNorth Africa / Middle East               0.4714853  0.1251952   3.766
## regionSouth Asia                               1.5736581  0.3039689   5.177
## regionSoutheast Asia, East Asia and Oceania    0.5643105  0.1435449   3.931
## regionSub-Saharan Africa                       1.4252242  0.1458920   9.769
## t                                             -0.0113545  0.0019478  -5.829
## logu5:regionHigh income                        0.1742724  0.0298345   5.841
## logu5:regionLatin America and Caribbean       -0.2610720  0.0331542  -7.874
## logu5:regionNorth Africa / Middle East        -0.1472862  0.0352319  -4.180
```

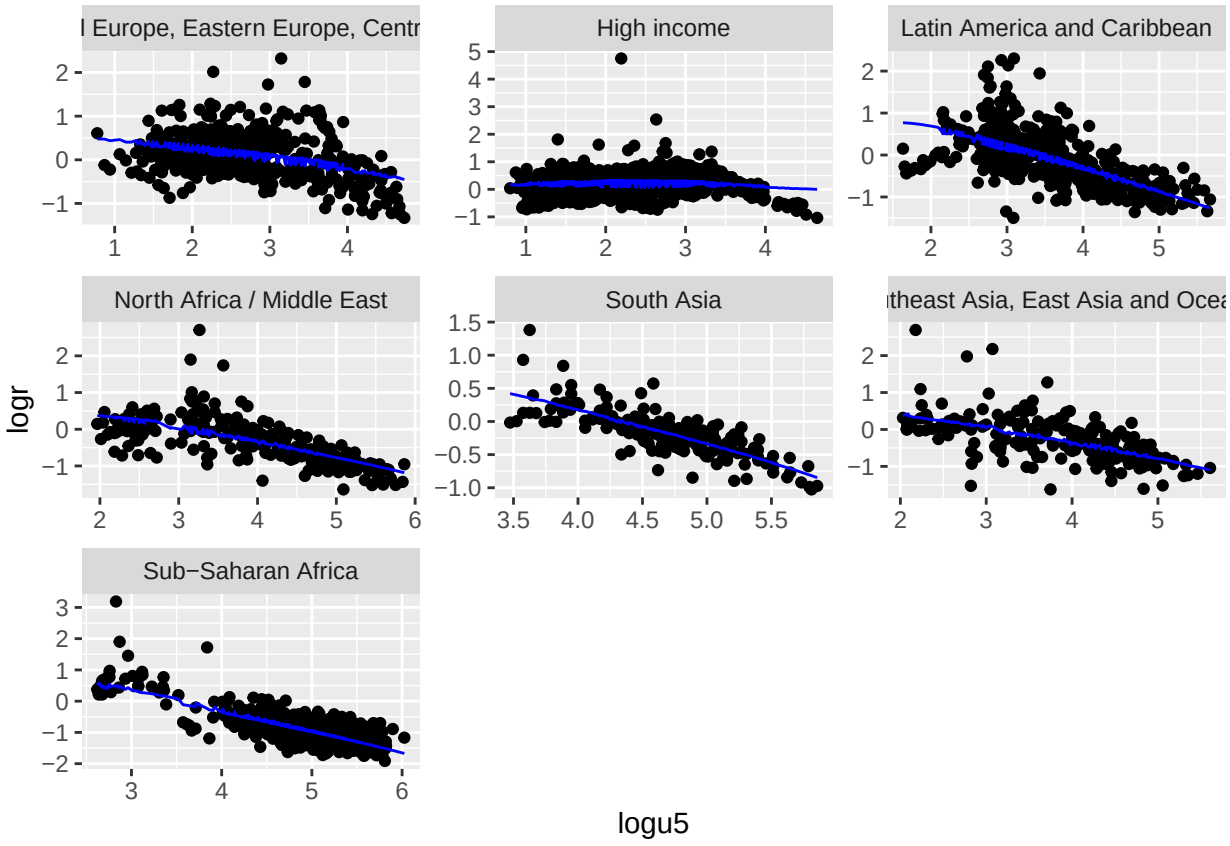
```
## logu5:regionSouth Asia                                -0.2781502  0.0671585  -4.142
## logu5:regionSoutheast Asia, East Asia and Oceania -0.1713357  0.0397931  -4.306
## logu5:regionSub-Saharan Africa                        -0.3732616  0.0357546 -10.440
## logu5:t                                                0.0019881  0.0004786   4.153
##                                                       Pr(>|t|)
## (Intercept)                                            < 2e-16 ***
## logu5                                                  < 2e-16 ***
## regionHigh income                                     9.03e-09 ***
## regionLatin America and Caribbean                      < 2e-16 ***
## regionNorth Africa / Middle East                      0.000169 ***
## regionSouth Asia                                      2.39e-07 ***
## regionSoutheast Asia, East Asia and Oceania           8.63e-05 ***
## regionSub-Saharan Africa                               < 2e-16 ***
## t                                                     6.10e-09 ***
## logu5:regionHigh income                               5.69e-09 ***
## logu5:regionLatin America and Caribbean               4.62e-15 ***
## logu5:regionNorth Africa / Middle East                2.99e-05 ***
## logu5:regionSouth Asia                                3.53e-05 ***
## logu5:regionSoutheast Asia, East Asia and Oceania     1.71e-05 ***
## logu5:regionSub-Saharan Africa                         < 2e-16 ***
## logu5:t                                               3.36e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4129 on 3262 degrees of freedom
## Multiple R-squared:  0.6078, Adjusted R-squared:  0.606
## F-statistic: 337.1 on 15 and 3262 DF,  p-value: < 2.2e-16
```

logu5*region significant

```
#Q1 part2
# all data
fit0 %>%
  augment(data = A2_training) %>%
  ggplot(aes(x = logu5, y= logr)) +
  geom_point()+
  geom_line(aes(x = logu5, y = .fitted), color = "blue")
```



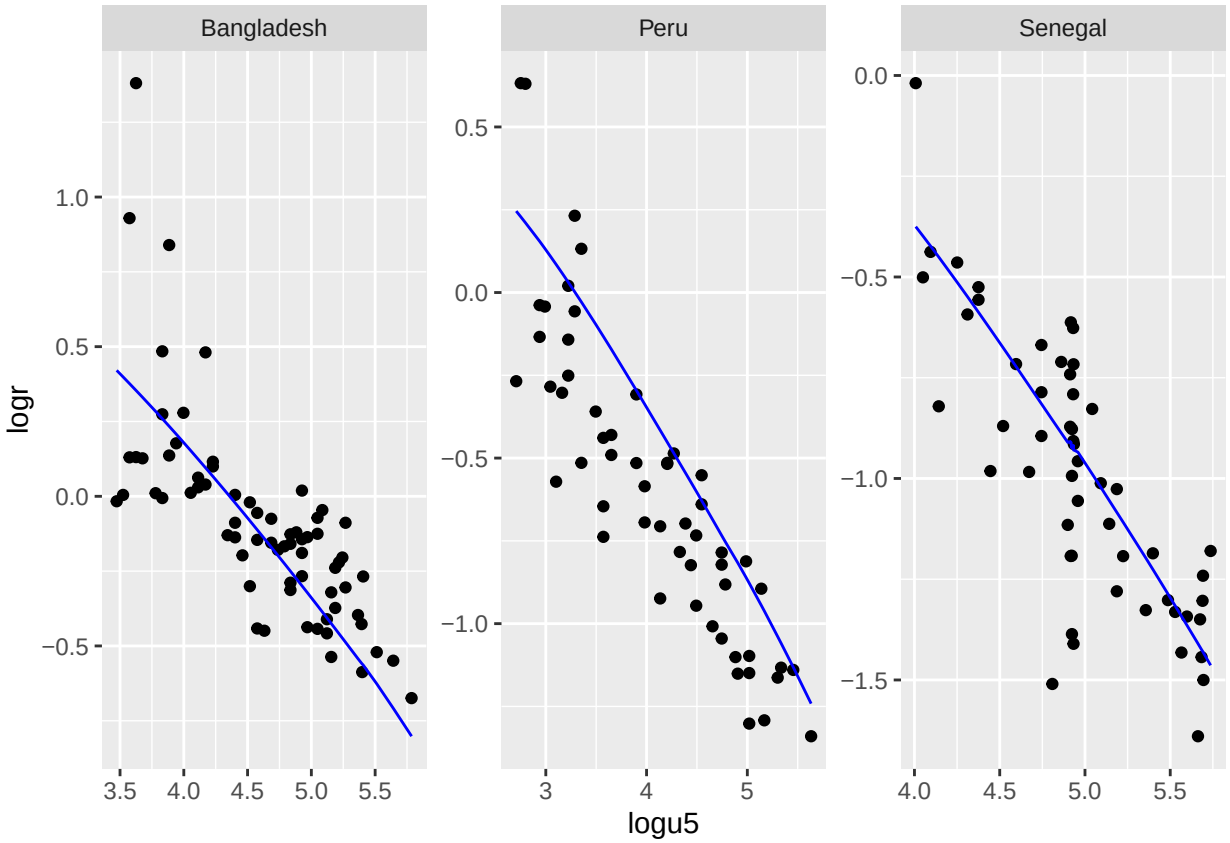

```
# different region
fit0 %>%
  augment(data = A2_training) %>%
  ggplot(aes(x = logu5, y= logr)) +
  geom_point()+
  geom_line(aes(x = logu5, y = .fitted), color = "blue")+
  facet_wrap(~region, scale = "free")
```




```
# 3 countries
A2_training %>%
  count(country_name) %>%
  top_n(3)
```

country_name	n
Bangladesh	69
Peru	55
Senegal	53

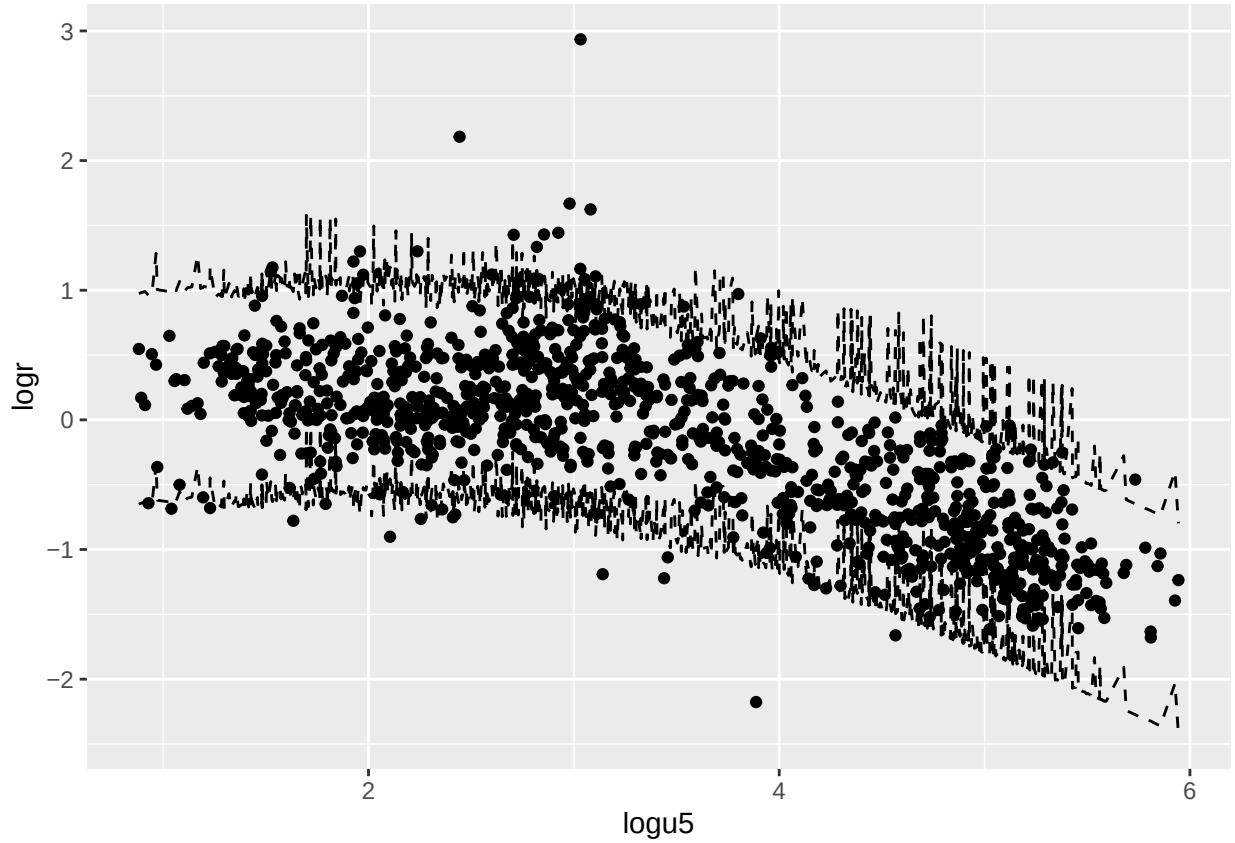
```
# 3 countries
fit0 %>% augment(data = A2_training) %>%
  filter(country_name %in% c("Bangladesh", "Peru", "Senegal")) %>%
  ggplot(aes(x = logu5, y = logr)) +
  geom_point() +
  geom_line(aes(x = logu5, y = .fitted), color = "blue") +
  facet_wrap(~country_name, scales = "free")
```




```
#Q1 part3
# estimate the root mean square error and absolute error
lm_pred <- tibble(pred = predict(fit0, A2_test))
lm_pred <- bind_cols(A2_test, lm_pred)
lm_pred %>%
  metrics(truth = logr,
          estimate = pred)
```

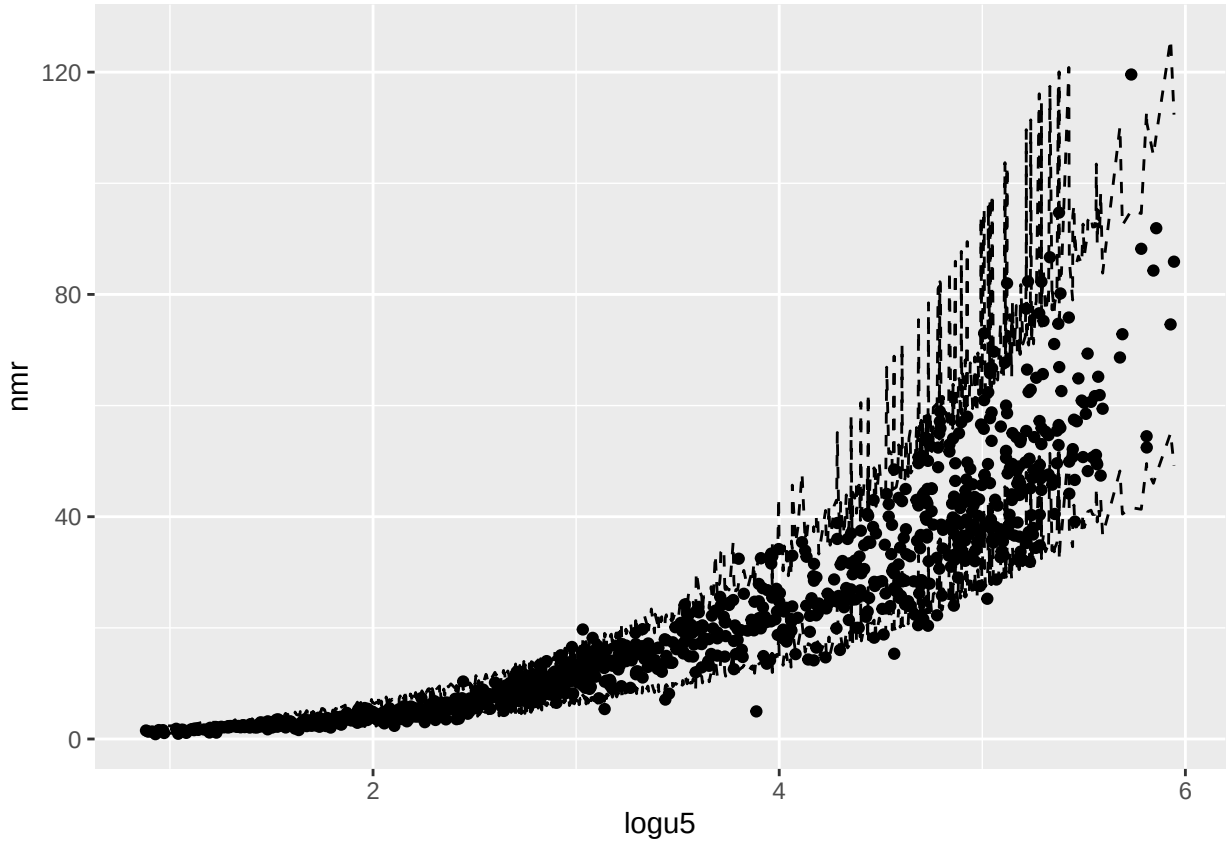
.metric	.estimator	.estimate
rmse	standard	0.4054866
rsq	standard	0.6135978
mae	standard	0.3002785

```
#Q1 part 4
# prediction
A2_PI <- as_tibble(predict(fit0, A2_test, interval = "prediction")) %>%
  bind_cols(lm_pred)

A2_PI %>%
  ggplot(aes(x = logu5, y = logr))+
  geom_point()+
  geom_line(aes(y = lwr), linetype = "dashed")+
  geom_line(aes(y = upr), linetype = "dashed")
```

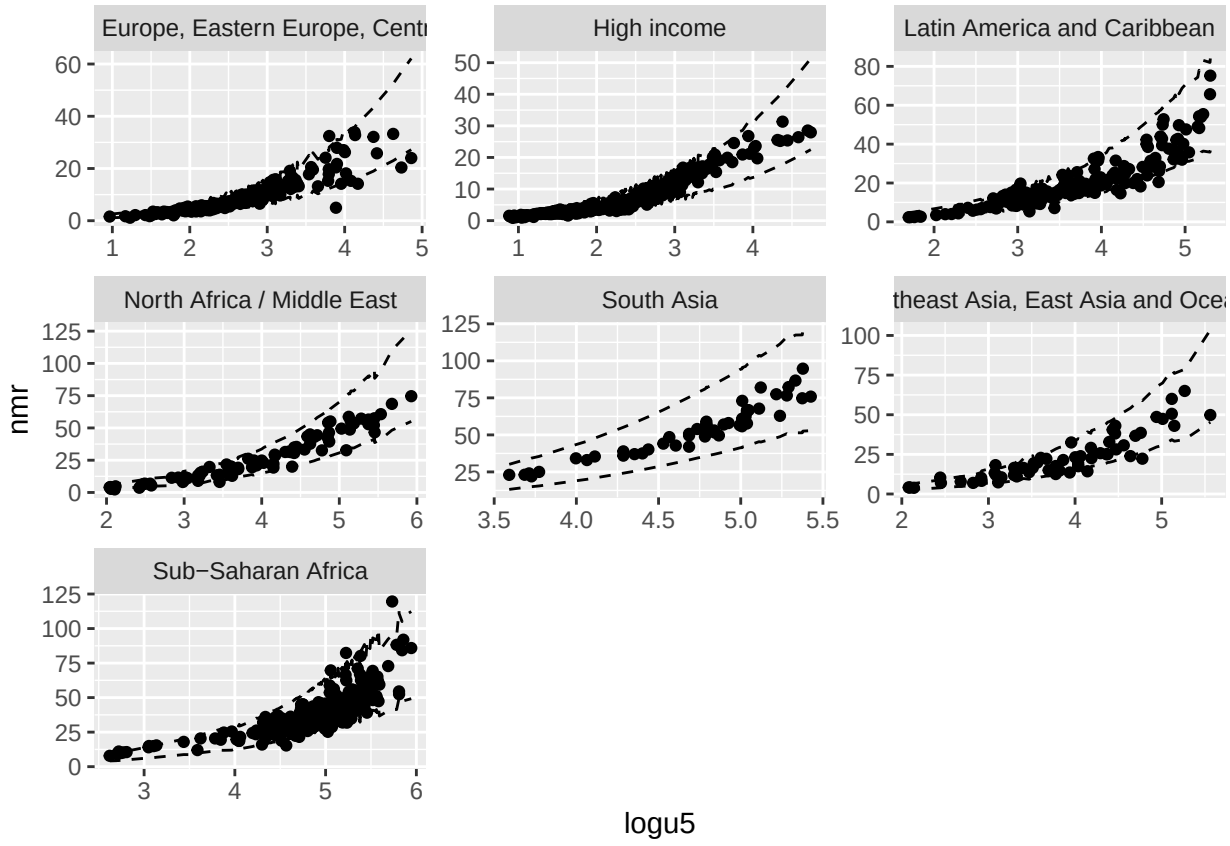



```
#non-linear

fit_test <- lm(log(nmr) ~ t +logu5 +region +logu5*t, data = A2_training)

lm_pred_test <- tibble(pred = predict(fit_test, A2_test))

lm_pred_test <- bind_cols(A2_test, lm_pred_test)
```

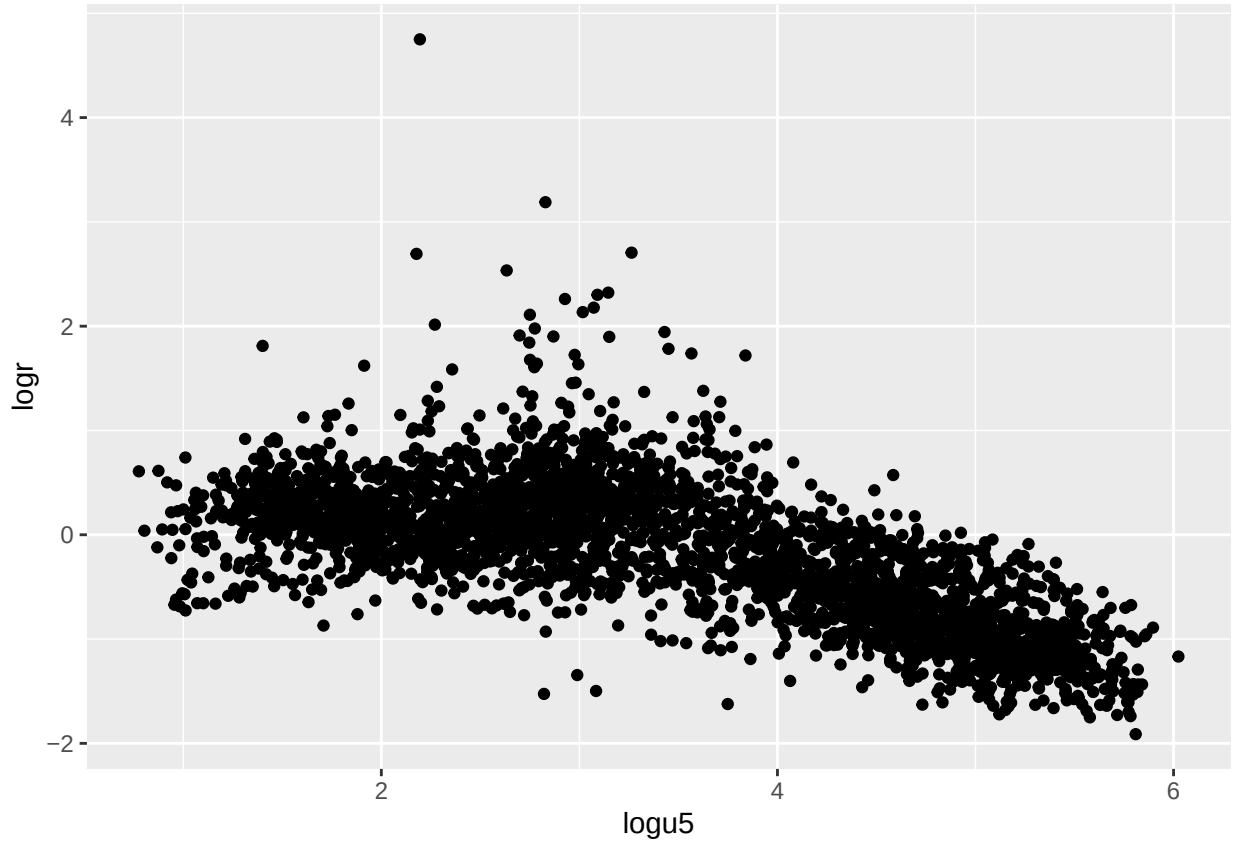
```
# all data
bb <- A2_PI <- as_tibble(predict(fit_test, A2_test, interval = "prediction")) %>%
  bind_cols(lm_pred) %>%
  mutate(n_lwr = exp(lwr),
         n_upr = exp(upr))

bb %>%
  ggplot(aes(x = logu5, y = nmr))+
  geom_point()+
  geom_line(aes(y = n_lwr), linetype = "dashed") +
  geom_line(aes(y = n_upr), linetype = "dashed")
```




```
# by region
bb %>%
  ggplot(aes(x = logu5, y = nmr))+
  geom_point()+
  geom_line(aes(y = n_lwr), linetype = "dashed") +
  geom_line(aes(y = n_upr), linetype = "dashed")+
  facet_wrap(~ region, scale = "free")
```




```
#non-linear variable

fit0 <- lm(logr ~ logu5 + logu5*region + logu5*t, A2_training)


ggplot(A2_training)+
  geom_point(aes(x = logu5, y = logr))
```

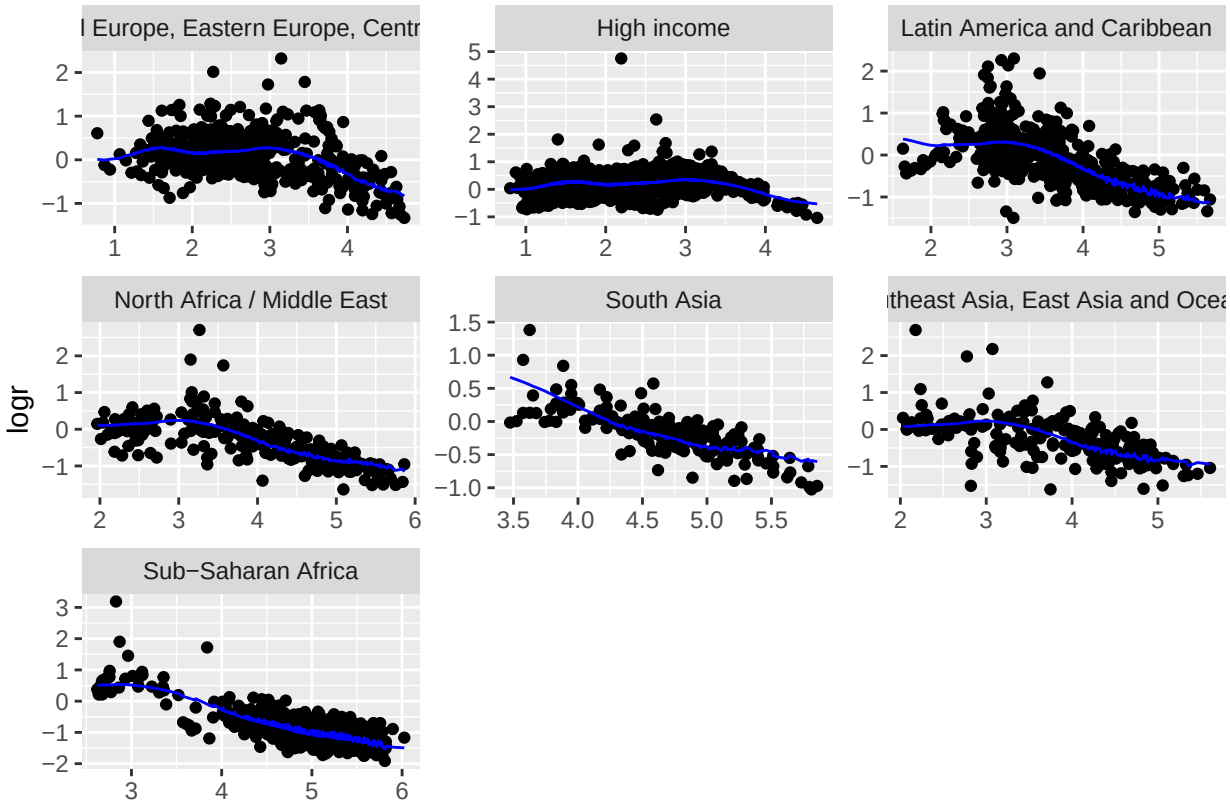



```
fit_bs <- lm(logr ~ bs(logu5, df= 15) + logu5*region + logu5*t, data = A2_training)
```

```
# root mean sqaure error and mean absolute error

lm_pred_bs <- tibble(pred = predict(fit_bs, A2_test))

lm_pred_bs <- bind_cols(A2_test, lm_pred_bs)

lm_pred_bs %>%
  metrics(truth = logr,
          estimate = pred)
```

.metric	.estimator	.estimate
rmse	standard	0.3797526
rsq	standard	0.6613863
mae	standard	0.2839409

```
fit_bs %>%
  augment(data = A2_training) %>%
  ggplot(aes(x = logu5, y = logr)) +
  geom_point()+
  geom_line(aes(x = logu5, y = .fitted), color = "blue")+
  facet_wrap(~ region, scale = "free")
```




```
#cross validation

folds <- vfold_cv(A2, v= 20)

spec <- linear_reg() %>%
  set_engine("lm")

fits_cv <- fit_resamples(spec,
                         logr ~ bs(logu5, df =15)+ bs(logu5, df =15)*region + bs(logu5, df =15)*t,
                         resamples = folds
)

# find mean
collect_metrics(fits_cv)
```

.metric	.estimator	mean	n	std_err	.config
rmse	standard	0.7674173	20	0.3032530	Preprocessor1_Model1
rsq	standard	0.5781415	20	0.0464494	Preprocessor1_Model1