Personal Models

Sam Harper

2024-09-23

Objective

The purpose of this document is to evaluate potential functional forms for the relationships between the personal exposure variables ($PM_{2.5}$ and BC) and the policy. We follow the work of Manning and Mulhally (2001) in applying a modified version of the 'Park test' (Park 1966). The basic idea of the Park test is to evaluate the relationship between the variance and the residuals. Here is the relevant part of Manning and Mulhally's paper:

> This moment structure (with a consistent initial estimate of β) is similar to one of the early tests for heteroscedasticity. In the original Park test (Park, 1966), the log of the estimated residual squared (on the scale of the analysis) is regressed on some factor z thought to cause heteroscedasticity in the error on the scale of the analysis. Here, we propose to use the residuals and predictions on the raw (untransformed) scale for y to estimate and test a very specific form of heteroscedasticity — one where the raw-scale variance is a power function of the raw-scale mean function is a power function of the raw-scale mean function. The OLS version of Eq. (17) is
>
> $$ln(y_i - \hat{y_i})^2 = \lambda_0 + \lambda_1 ln(\hat{y}) + v_i$$
>
> where $\hat{y_i} = exp(x_i \beta)$ is from one of the GLM specifications...The estimate of the coefficient λ_1 on the log of the raw-scale prediction will tell us which GLM model to employ if the GLM option is chosen.

Data

The data with personal exposures and all of the covariates are loaded below, and we fit the full ETWFE model to the data before obtaining the residuals that form the basis for the Park test.

	Unique	Missing Pct.	Mean	SD	Min	Median	Max	Histogram
pe	1276	59	99.9	122.9	0.0	60.7	1344.3	
bc	1164	63	3.7	6.5	0.0	2.1	83.3	
hh_num	11	2	2.3	1.1	0.0	2.0	9.0	
ets_former	3	0	0.1	0.3	0.0	0.0	1.0	
ets_lived	3	0	0.4	0.5	0.0	0.0	1.0	
ets_none	3	0	0.2	0.4	0.0	0.0	1.0	
out_temp	3085	0	-4.0	3.6	-14.0	-3.5	4.8	
out_dew	3085	0	-16.2	5.6	-34.6	-16.0	-2.6	

```
# necessary packages
library(here)
library(tidyverse)
library(fixest)
library(splines)
library(modelsummary)
library(modelr)
library(marginaleffects)
library(tinytable)

# load data
pd <- readRDS(here("data-clean",
  "ap-data-personal.rds"))

pd %>%
  select(pe, bc, hh_num, ets_former, ets_lived,
    ets_none, out_temp, out_dew) %>% datasummary_skim()
```

Models

Personal $PM_{2.5}$

Now fit the full model using a Gamma distribution with a log link (see Manning and Mullahy (2001) for details).

```r
pe_full_gamma <- feglm(
  pe ~ treat:cohort_year_2019:year_2019 +
    treat:cohort_year_2019:year_2021 +
    treat:cohort_year_2020:year_2021 +
    treat:cohort_year_2021:year_2021 +
    cohort_year_2019 + cohort_year_2020 +
    cohort_year_2021 + year_2019 + year_2021 +
    hh_num + ets_former + ets_lived +
    ets_none + ns(out_temp, df=2) +
    ns(out_dew, df=2),
    data = pd, cluster = ~v_id,
    family = Gamma(link = "log"))
```

```
NOTE: 1,844 observations removed because of NA values (LHS: 1,843, RHS: 74).
```

```r
modelsummary(list("PM<sub>2.5</sub> Gamma" = pe_full_gamma),
  statistic = c("SE" = "std.error",
  "95% CI" = "{conf.low}, {conf.high}"),
   shape = term ~ model + statistic,
   gof_omit = 'DF|Deviance|R2|AIC|BIC|RMSE')
```

Now we undertake the Park test. First, let's get the predictions from the above model on both the response (absolute) scale and on the link (log) scale, and calculate the squared residuals on the absolute scale.

```r
pe_pred <- pd %>%
  filter(row_number()
  %in% obs(pe_full_gamma)) %>%
  # add predicted E(y)
  add_predictions(pe_full_gamma,
    var = "pred_link", type = "link") %>%
  # add predicted E(y) on the response scale
  add_predictions(pe_full_gamma,
    var = "pred_response", type = "response") %>%
  mutate(
    resid_response = (pe - pred_response)^2)
```

Now we regress the squared residuals on the predicted values of the response variable (making sure to use clustered standard errors).

	PM<sub>2.5</sub> Gamma		
	Est.	SE	95% CI
(Intercept)	4.911	0.254	4.400, 5.422
cohort_year_2019	−0.045	0.112	−0.270, 0.179
cohort_year_2020	−0.059	0.108	−0.277, 0.159
cohort_year_2021	0.018	0.085	−0.153, 0.189
year_2019	−0.389	0.115	−0.621, −0.157
year_2021	−0.426	0.110	−0.646, −0.205
hh_num	−0.087	0.029	−0.146, −0.029
ets_former	−0.782	0.122	−1.028, −0.536
ets_lived	−0.436	0.084	−0.605, −0.268
ets_none	−0.917	0.108	−1.135, −0.699
ns(out_temp, df = 2)1	−1.191	0.395	−1.985, −0.398
ns(out_temp, df = 2)2	−0.981	0.263	−1.510, −0.452
ns(out_dew, df = 2)1	2.194	0.596	0.995, 3.393
ns(out_dew, df = 2)2	0.887	0.262	0.360, 1.414
treat × cohort_year_2019 × year_2019	0.058	0.173	−0.289, 0.405
treat × cohort_year_2019 × year_2021	−0.194	0.167	−0.529, 0.141
treat × year_2021 × cohort_year_2020	0.154	0.264	−0.375, 0.684
treat × year_2021 × cohort_year_2021	−0.301	0.142	−0.587, −0.015
Num.Obs.	1280		
Std.Errors	by: v_id		

	PM<sub>2.5</sub>		
	Est.	SE	95% CI
(Intercept)	3.499	1.104	1.280, 5.718
pred_link	1.295	0.223	0.847, 1.742
Num.Obs.	1280		
Std.Errors	by: v_id		

```r
pe_park <- feglm(
  resid_response ~ pred_link,
  family=Gamma(link = "log"),
  data = pe_pred, cluster = ~v_id)

modelsummary(list("PM<sub>2.5</sub>" = pe_park),
  statistic = c("SE" = "std.error",
  "95% CI" = "{conf.low}, {conf.high}"),
  shape = term ~ model + statistic,
  gof_omit = 'DF|Deviance|R2|AIC|BIC|RMSE')
```

Finally, we test the value of $x\beta$ against several alternatives that help to guide our choice of model. As Manning and Mulhally note, if the raw-scale variance (`resid_response`) does not depend on the raw-scale prediction `pred-link` ($\beta = 0$), then consider the lognormal distribution; if the raw-scale variance is proportional to the raw-scale prediction ($\beta = 1$), consider the Poisson-like model; if the raw-scale variance is quadratic in the raw-scale prediction ($\beta = 2$), then consider the gamma model, and if the raw-scale variance is cubic in the raw-scale prediction ($\beta = 3$), then consider the inverse Gaussian (Wald) model.

```r
pe_tests <- avg_comparisons(pe_park,
  var = "pred_link", type = "link",
  hypothesis = c("b1 - 0 = 0",
                 "b1 - 1 = 0",
                 "b1 - 2 = 0",
                 "b1 - 3 = 0"))

pe_results <- pe_tests %>%
  select(term, estimate, std.error,
    statistic, p.value) %>%
  mutate(family = c("Lognormal", "Poisson-like",
    "Gamma", "Inverse Gaussian")) %>%
  relocate(family)
```

family	term	estimate	std.error	statistic	p.value
Lognormal	b1-0=0	1.295	0.223	5.81	0.0000
Poisson-like	b1-1=0	0.295	0.223	1.32	0.1855
Gamma	b1-2=0	-0.705	0.223	-3.17	0.0015
Inverse Gaussian	b1-3=0	-1.705	0.223	-7.66	0.0000

```
tt(pe_results, digits = 3) %>%
  format_tt(j=6, sprintf = "%.4f")
```

On the basis of these tests, it seems like the Poisson-like model generates the smallest test statistic (and the largest p-value), but the Gamma model also seems to fit the data relatively well.

Personal BC

For personal BC we follow the same procedure as above, using the full model with covariates.

```
bc_full_gamma <- feglm(
  pe ~ treat:cohort_year_2019:year_2019 +
    treat:cohort_year_2019:year_2021 +
    treat:cohort_year_2020:year_2021 +
    treat:cohort_year_2021:year_2021 +
    cohort_year_2019 + cohort_year_2020 +
    cohort_year_2021 + year_2019 + year_2021 +
    hh_num + ets_former + ets_lived +
    ets_none + ns(out_temp, df=2) +
    ns(out_dew, df=2),
    data = pd, cluster = ~v_id,
    family = Gamma(link = "log"))
```

```
NOTE: 1,844 observations removed because of NA values (LHS: 1,843, RHS: 74).
```

```
bc_pred <- pd %>%
  filter(row_number()
  %in% obs(bc_full_gamma)) %>%
  # add predicted E(y)
  add_predictions(bc_full_gamma,
```

	Black carbon		
	Est.	SE	95% CI
(Intercept)	−0.193	0.030	−0.252, −0.133
pred_link	2.026	0.007	2.013, 2.039
Num.Obs.	1156		
Std.Errors	by: v_id		

```
    var = "pred_link", type = "link") %>%
  # add predicted E(y) on the response scale
  add_predictions(bc_full_gamma,
    var = "pred_response", type = "response") %>%
  mutate(
    resid_response = (bc - pred_response)^2)

bc_park <- feglm(
  resid_response ~ pred_link,
  family=Gamma(link = "log"),
  data = bc_pred, cluster = ~v_id)
```

```
NOTE: 124 observations removed because of NA values (LHS: 124).
```

```
modelsummary(list("Black carbon" = bc_park),
  statistic = c("SE" = "std.error",
  "95% CI" = "{conf.low}, {conf.high}"),
   shape = term ~ model + statistic,
   gof_omit = 'DF|Deviance|R2|AIC|BIC|RMSE')
```

```
bc_tests <- avg_comparisons(bc_park,
  var = "pred_link", type = "link",
  hypothesis = c("b1 - 0 = 0",
                 "b1 - 1 = 0",
                 "b1 - 2 = 0",
                 "b1 - 3 = 0"))

bc_results <- bc_tests %>%
  select(term, estimate, std.error,
    statistic, p.value) %>%
  mutate(family = c("Lognormal", "Poisson-like",
```

family	term	estimate	std.error	statistic	p.value
Lognormal	b1-0=0	2.0258	0.0066	306.94	0.0000
Poisson-like	b1-1=0	1.0258	0.0066	155.42	0.0000
Gamma	b1-2=0	0.0258	0.0066	3.91	0.0001
Inverse Gaussian	b1-3=0	-0.9742	0.0066	-147.6	0.0000

```
    "Gamma", "Inverse Gaussian")) %>%
  relocate(family)
tt(bc_results, digits = 3) %>%
  format_tt(j=6, sprintf = "%.4f")
```

For BC it appears that the Gamma family is the best fit. Given the simliarity between the Gamma and Poisson-like models for $PM_{2.5}$, using a Gamma model for both seems like a reasonable choice.

Indoor 24h $PM_{2.5}$

Now for 24h indoor $PM_{2.5}$ we follow the same procedure as above. However, since there is no Wave 1 data for indoor, we need to exclude the cohort of villages that were treated in 2019, since their values of indoor $PM_{2.5}$ may already have been affected by the policy. Thus we can only estimate the ATTs for the 2020 and 2021 cohorts in 2021.

```
idd <- readRDS(here("data-clean",
 "ap-data-i24h.rds")) %>%
  filter(cohort_year_2019 != 1)

i24_full_gamma <- feglm(
  i24  ~
    treat:cohort_year_2020:year_2021 +
    treat:cohort_year_2021:year_2021 +
    cohort_year_2020 + cohort_year_2021 +
    year_2021 + hh_num + ets_former + ets_lived +
    ets_none + ns(out_temp, df=2) +
    ns(out_dew, df=2),
    data = idd, cluster = ~v_id,
    family = Gamma(link = "log"))
```

```
NOTE: 14 observations removed because of NA values (RHS: 14).
```

	Indoor 24h PM<sub>2.5</sub>		
	Est.	SE	95% CI
(Intercept)	0.195	0.929	−1.685, 2.075
pred_link	1.937	0.214	1.505, 2.370
Num.Obs.	411		
Std.Errors	by: v_id		

```r
i24_pred <- idd %>%
  filter(row_number()
  %in% obs(i24_full_gamma)) %>%
  # add predicted E(y)
  add_predictions(i24_full_gamma,
    var = "pred_link", type = "link") %>%
  # add predicted E(y) on the response scale
  add_predictions(i24_full_gamma,
    var = "pred_response", type = "response") %>%
  mutate(
    resid_response = (i24 - pred_response)^2)

i24_park <- feglm(
  resid_response ~ pred_link,
  family=Gamma(link = "log"),
  data = i24_pred, cluster = ~v_id)

modelsummary(list(
  "Indoor 24h PM<sub>2.5</sub>" = i24_park),
  statistic = c("SE" = "std.error",
  "95% CI" = "{conf.low}, {conf.high}"),
   shape = term ~ model + statistic,
   gof_omit = 'DF|Deviance|R2|AIC|BIC|RMSE')
```

```r
i24_tests <- avg_comparisons(i24_park,
  var = "pred_link", type = "link",
  hypothesis = c("b1 - 0 = 0",
                 "b1 - 1 = 0",
                 "b1 - 2 = 0",
                 "b1 - 3 = 0"))

i24_results <- i24_tests %>%
```

family	term	estimate	std.error	statistic	p.value
Lognormal	b1-0=0	1.9374	0.214	9.06	0.0000
Poisson-like	b1-1=0	0.9374	0.214	4.384	0.0000
Gamma	b1-2=0	-0.0626	0.214	-0.293	0.7699
Inverse Gaussian	b1-3=0	-1.0626	0.214	-4.969	0.0000

```
  select(term, estimate, std.error,
    statistic, p.value) %>%
  mutate(family = c("Lognormal", "Poisson-like",
    "Gamma", "Inverse Gaussian")) %>%
  relocate(family)
tt(i24_results, digits = 3) %>%
  format_tt(j=6, sprintf = "%.4f")
```

The best fit is provided by the Gamma family.

Indoor seasonal $PM_{2.5}$

Now for seasonal indoor $PM_{2.5}$ we follow the same procedure as above. Again here there is no Wave 1 data for indoor, so we exclude the cohort of villages that were treated in 2019, since their values of indoor $PM_{2.5}$ may already have been affected by the policy. Thus we can only estimate the ATTs for the 2020 and 2021 cohorts in 2021.

```
# filter for useable values of indoor seasonal
isd <- readRDS(here("data-clean",
 "ap-data-iseason.rds")) %>%
  filter(cohort_year_2019 != 1)

is_full_gamma <- feglm(
  is  ~
    treat:cohort_year_2020:year_2021 +
    treat:cohort_year_2021:year_2021 +
    cohort_year_2020 + cohort_year_2021 +
    year_2021 + hh_num + ets_former + ets_lived +
    ets_none + ns(out_temp, df=2) +
    ns(out_dew, df=2),
    data = isd, cluster = ~v_id,
    family = Gamma(link = "log"))
```

	Indoor Seasonal PM<sub>2.5</sub>		
	Est.	SE	95% CI
(Intercept)	−0.947	0.963	−2.895, 1.001
pred_link	2.089	0.227	1.630, 2.548
Num.Obs.	374		
Std.Errors	by: v_id		

```
NOTE: 14 observations removed because of NA values (RHS: 14).
```

```r
is_pred <- isd %>%
  filter(row_number()
  %in% obs(is_full_gamma)) %>%
  # add predicted E(y)
  add_predictions(is_full_gamma,
    var = "pred_link", type = "link") %>%
  # add predicted E(y) on the response scale
  add_predictions(is_full_gamma,
    var = "pred_response", type = "response") %>%
  mutate(
    resid_response = (is - pred_response)^2)

is_park <- feglm(
  resid_response ~ pred_link,
  family=Gamma(link = "log"),
  data = is_pred, cluster = ~v_id)

modelsummary(list(
  "Indoor Seasonal PM<sub>2.5</sub>" = is_park),
  statistic = c("SE" = "std.error",
  "95% CI" = "{conf.low}, {conf.high}"),
   shape = term ~ model + statistic,
   gof_omit = 'DF|Deviance|R2|AIC|BIC|RMSE')
```

```r
is_tests <- avg_comparisons(is_park,
  var = "pred_link", type = "link",
  hypothesis = c("b1 - 0 = 0",
                 "b1 - 1 = 0",
                 "b1 - 2 = 0",
                 "b1 - 3 = 0"))
```

family	term	estimate	std.error	statistic	p.value
Lognormal	b1-0=0	2.0894	0.227	9.205	0.0000
Poisson-like	b1-1=0	1.0894	0.227	4.799	0.0000
Gamma	b1-2=0	0.0894	0.227	0.394	0.6938
Inverse Gaussian	b1-3=0	-0.9106	0.227	-4.012	0.0001

```r
is_results <- is_tests %>%
  select(term, estimate, std.error,
    statistic, p.value) %>%
  mutate(family = c("Lognormal", "Poisson-like",
    "Gamma", "Inverse Gaussian")) %>%
  relocate(family)
tt(is_results, digits = 3) %>%
  format_tt(j=6, sprintf = "%.4f")
```

Again, seems like Gamma provides the best overall fit. Overall, although technically for personal exposure the Poisson-like family provides the best fit, it seems reasonable for consistency to use the Gamma family across all 4 outcomes.

Manning WG, Mullahy J. 2001. Estimating log models: To transform or not to transform? Journal of Health Economics 20:461–494; doi:10.1016/S0167-6296(01)00086-8.

Park RE. 1966. Estimation with Heteroscedastic Error Terms. Econometrica 34:888–888; doi:10.2307/1910108.