

I've interspersed R code throughout.

1. Let \mathbf{x}_i be a vector of variables on observations $i = 1, \dots, n$, z_i be a continuous variable, y_i be a binary dependent variable, and Φ be the normal CDF. In the model

$$\Pr(y_i = 1 | \mathbf{x}_i, z_i) = \Phi(\mathbf{x}_i \boldsymbol{\beta} + \gamma_1 z_i + \gamma_2 z_i^2)$$

what is the partial effect of z_i on the response probability? What is an estimate of that partial effect? How would you obtain the standard error of the estimated partial effect?

Answer:

- (a) The partial effect is

$$\begin{aligned} \tilde{p} = \frac{\partial P_i}{\partial z_i} &= \frac{\partial \Phi(\mathbf{x}_i \boldsymbol{\beta} + \gamma_1 z_i + \gamma_2 z_i^2)}{\partial (\mathbf{x}_i \boldsymbol{\beta} + \gamma_1 z_i + \gamma_2 z_i^2)} \times \frac{\partial (\mathbf{x}_i \boldsymbol{\beta} + \gamma_1 z_i + \gamma_2 z_i^2)}{\partial z_i} \\ &= \phi(\mathbf{x}_i \boldsymbol{\beta} + \gamma_1 z_i + \gamma_2 z_i^2) (\gamma_1 + 2\gamma_2 z_i), \end{aligned} \quad (1)$$

where ϕ is the probability density function for the standard normal.

- (b) An estimate of the partial effect can be obtained by substituting the maximum likelihood estimates of $\boldsymbol{\theta} = (\boldsymbol{\beta}, \gamma_1, \gamma_2)'$, $\hat{\boldsymbol{\theta}}$, into the expression in the previous answer.
- (c) A standard error on the estimated partial effect can be obtained via the following asymptotic approximation to the variance of the estimated partial effect (the “delta” method). If $h(\hat{\boldsymbol{\theta}})$ is a quantity of interest, where $\hat{\boldsymbol{\theta}} \in \mathbb{R}^k$ and $h : \mathbb{R}^k \mapsto \mathbb{R}$, then

$$V[h(\hat{\boldsymbol{\theta}})] \approx \nabla' V(\hat{\boldsymbol{\theta}}) \nabla$$

where $\nabla = \partial h(\hat{\boldsymbol{\theta}}) / \partial \hat{\boldsymbol{\theta}}$ is a k -by-1 vector of partial derivatives and $V(\hat{\boldsymbol{\theta}})$ is an (estimated) k -by- k variance-covariance matrix (i.e., the variance-covariance matrix of the MLEs of $\hat{\boldsymbol{\theta}}$). In the present context,

$$h(\hat{\boldsymbol{\theta}}) = \phi(\mathbf{x}_i \hat{\boldsymbol{\beta}} + \hat{\gamma}_1 z_i + \hat{\gamma}_2 z_i^2) (\hat{\gamma}_1 + 2\hat{\gamma}_2 z_i) \quad (2)$$

and deriving $\nabla = \partial h(\hat{\boldsymbol{\theta}}) / \partial \hat{\boldsymbol{\theta}}$ will involve some tedious mathematics. Begin by re-writing equation 2 as

$$h(\hat{\boldsymbol{\theta}}) = \phi(g(\hat{\boldsymbol{\theta}})) \times f(\hat{\boldsymbol{\theta}}) \quad (3)$$

noting that by the product rule of differentiation

$$\nabla = \frac{\partial h(\hat{\boldsymbol{\theta}})}{\partial \hat{\boldsymbol{\theta}}} = \frac{\partial \phi(g(\hat{\boldsymbol{\theta}}))}{\partial \hat{\boldsymbol{\theta}}} f(\hat{\boldsymbol{\theta}}) + \phi(g(\hat{\boldsymbol{\theta}})) \frac{\partial f(\hat{\boldsymbol{\theta}})}{\partial \hat{\boldsymbol{\theta}}},$$

where

$$\begin{aligned} g(\hat{\boldsymbol{\theta}}) &= \mathbf{x}_i \hat{\boldsymbol{\beta}} + \hat{\gamma}_1 z_i + \hat{\gamma}_2 z_i^2, \\ f(\hat{\boldsymbol{\theta}}) &= \hat{\gamma}_1 + 2\hat{\gamma}_2 z_i, \\ \frac{\partial g(\hat{\boldsymbol{\theta}})}{\partial \hat{\boldsymbol{\theta}}} &= \begin{bmatrix} \mathbf{x}_i \\ z_i \\ z_i^2 \end{bmatrix}, \\ \frac{\partial f(\hat{\boldsymbol{\theta}})}{\partial \hat{\boldsymbol{\theta}}} &= \begin{bmatrix} \mathbf{0} \\ 1 \\ 2z_i \end{bmatrix}. \end{aligned}$$

i.e., $g(\hat{\boldsymbol{\theta}})$ and $f(\hat{\boldsymbol{\theta}})$ are scalars, and the derivatives are k -by-1 gradient vectors. Note also that written as a function of terms that only involve $\hat{\boldsymbol{\theta}}$,

$$\phi(g(\hat{\boldsymbol{\theta}})) \propto \exp \left[\frac{-g(\hat{\boldsymbol{\theta}})^2}{2} \right]$$

and so

$$\frac{\partial \phi(g(\hat{\boldsymbol{\theta}}))}{\partial \hat{\boldsymbol{\theta}}} = -\phi(g(\hat{\boldsymbol{\theta}})) \frac{\partial g(\hat{\boldsymbol{\theta}})}{\partial \hat{\boldsymbol{\theta}}}.$$

Thus we obtain

$$\begin{aligned} \nabla &= -\phi(g(\hat{\boldsymbol{\theta}})) f(\hat{\boldsymbol{\theta}}) \begin{bmatrix} \mathbf{x}_i \\ z_i \\ z_i^2 \end{bmatrix} + \phi(g(\hat{\boldsymbol{\theta}})) \begin{bmatrix} \mathbf{0} \\ 1 \\ 2z_i \end{bmatrix} \\ &= \phi(g(\hat{\boldsymbol{\theta}})) \left(\begin{bmatrix} \mathbf{0} \\ 1 \\ 2z_i \end{bmatrix} - f(\hat{\boldsymbol{\theta}}) \begin{bmatrix} \mathbf{x}_i \\ z_i \\ z_i^2 \end{bmatrix} \right) \end{aligned}$$

which is actually reasonably straightforward to compute.

Monte Carlo methods are also a simple way to proceed in this instance: letting t index Monte Carlo iterates, we could

- i. sample $\hat{\boldsymbol{\theta}}^{(t)}$ from its multivariate normal distribution, $N(\hat{\boldsymbol{\theta}}, V(\hat{\boldsymbol{\theta}}))$; i.e., asymptotically this corresponds to the Bayesian posterior density for $\boldsymbol{\theta}$ under a uniform prior density.
- ii. compute $\tilde{P}^{(t)}$ as a function of $\hat{\boldsymbol{\theta}}^{(t)}$ as in equation 1.

This procedure generates (arbitrarily many) samples from the posterior density for \tilde{P} , which can then summarize as we wish, computing a variance, a standard deviation, critical quantiles, etc, and has the virtue of involving a lot less mathematics than the delta method.

2. Download the Marinov [data](#) (in R dpt format, read it with the `dget` command) on interstate trade sanctions and answer the following questions. Key variables in the data set are

- `statea`: name of sender
- `stateb`: name of target
- `year`: year of observation
- `startyear`: length of spell, start of current year
- `endyear`: length of spell, end of current year
- `ended`: did sanctions end in current year
- `usdummy`: is the United States the sender (time invariant)
- `demtarg`: is the target a democracy (time invariant)
- `align`: is there an alliance between the two countries (time invariant)
- `multilat`: is the sanction episode part of a multilateral sanction (time invariant)
- `gdppctarg`: GDP per capita of the target country (time varying)

(a) Show the Kaplan-Meier estimates of the survivor functions for sanctions for when the United States is the sender, versus when the United States is not the sender. Comment briefly on what these graphs reveal.

```

R Code
1 > data <- dget(file="marinovhw4.dpt")
2 > library(survival)
3 > attach(data)
4 > km2a <- survfit(Surv(startyear, endyear, ended) ~ usdummy,
5 +                 data=data)
6 > summary(km2a)

```

Call: `survfit(formula = Surv(startyear, endyear, ended) ~ usdummy, data = data)`

		usdummy=0						
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
1	22	5	0.7727	0.0893		0.6160		0.969
2	17	4	0.5909	0.1048		0.4174		0.837
3	13	2	0.5000	0.1066		0.3292		0.759
4	11	1	0.4545	0.1062		0.2876		0.718
7	10	1	0.4091	0.1048		0.2476		0.676
11	9	1	0.3636	0.1026		0.2092		0.632
12	8	2	0.2727	0.0950		0.1378		0.540
19	6	1	0.2273	0.0893		0.1052		0.491

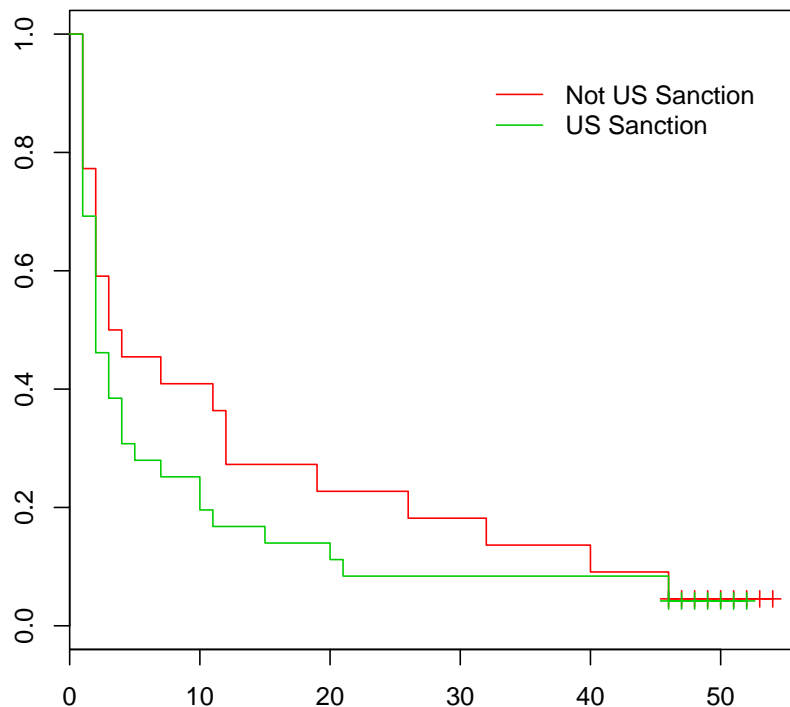
26	5	1	0.1818	0.0822	0.0749	0.441
32	4	1	0.1364	0.0732	0.0476	0.390
40	3	1	0.0909	0.0613	0.0243	0.341
46	2	1	0.0455	0.0444	0.0067	0.308

```
usdummy=1
```

time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
1	39	12	0.692	0.0739	0.56161	0.853
2	27	9	0.462	0.0798	0.32884	0.648
3	18	3	0.385	0.0779	0.25859	0.572
4	15	3	0.308	0.0739	0.19216	0.493
5	11	1	0.280	0.0723	0.16856	0.464
7	10	1	0.252	0.0703	0.14568	0.435
10	9	2	0.196	0.0648	0.10232	0.375
11	7	1	0.168	0.0613	0.08202	0.343
15	6	1	0.140	0.0571	0.06282	0.311
20	5	1	0.112	0.0521	0.04492	0.279
21	4	1	0.084	0.0460	0.02868	0.246
46	2	1	0.042	0.0375	0.00727	0.242

R Code

```
1 > plot(km2a,
2 + legend.pos=1,
3 + legend.text=c("Not US Sanction", "US Sanction"),
4 + col=2:3)
```



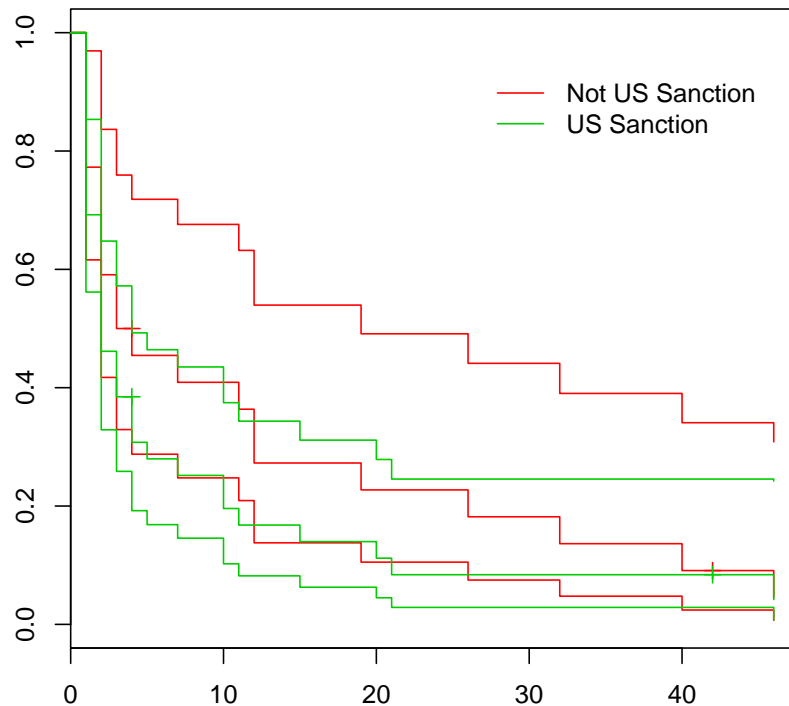
The graph has more right-censoring than is really in the data, but this

seems to be a bug in the plot method. Here is a fix, where we explicitly tell the plotting method where it find the right-censored observations; I also overlay the confidence intervals.

```

R Code
1 > ## find where we have breaks in the endyear series
2 > n <- dim(data)[1]
3 > spellbreak <- rep(FALSE,n)
4 > endyear.lag <- c(NA,endyear[-n])
5 > spellbreak[endyear != (endyear.lag + 1)] <- TRUE
6 > lastObs <- c(spellbreak[-1],TRUE)
7 > cox2a <- coxph(Surv(startyear,endyear,ended) ~ strata(usdummy),
8 + data=data)
9 > kmCox2a <- survfit(cox2a,type="kaplan-m")
10 > plot(kmCox2a,
11 + conf.int=TRUE,
12 + legend.pos=1,
13 + legend.text=c("Not US Sanction","US Sanction"),
14 + mark.time=data$endyear[lastObs & data$ended==0],
15 + col=2:3)

```



- (b) Show the Kaplan-Meier estimates of the survivor functions for sanctions for when the target country is a democracy, versus when the target country is not a democracy. Comment briefly on what these graphs reveal.

Answer: There is some missing data here that we might want to deal with. I covered this in the R session on Tuesday; e.g., we created a stripped down version of the data set, with just one observation per spell, treating democracy as a non-time-varying covariate.

```

R Code
1 > ## collapse data so we have only one obs per spell
2 > ## for KM plots
3 > newdata <- data[lastObs,]
4 > newdata$startyear <- 0
5 > ## patch missing data
6 > newdata$demtarg[9] <- 1
7 > km2b <- survfit(Surv(startyear, endyear, ended) ~ demtarg,
8 + data=newdata)
9 > summary(km2b)

```

```

Call: survfit(formula = Surv(startyear, endyear, ended) ~ demtarg,
data = newdata)

```

5 observations deleted due to missingness

```

demtarg=0
time n.risk n.event survival std.err lower 95% CI upper 95% CI
1 34 8 0.7647 0.0727 0.6346 0.921
2 26 8 0.5294 0.0856 0.3856 0.727
3 18 3 0.4412 0.0852 0.3022 0.644
4 15 3 0.3529 0.0820 0.2239 0.556
5 12 1 0.3235 0.0802 0.1990 0.526
7 11 1 0.2941 0.0781 0.1747 0.495
10 10 2 0.2353 0.0727 0.1284 0.431
11 8 1 0.2059 0.0693 0.1064 0.398
12 7 2 0.1471 0.0607 0.0655 0.330
21 5 1 0.1176 0.0553 0.0469 0.295
26 4 1 0.0882 0.0486 0.0299 0.260
40 3 1 0.0588 0.0404 0.0153 0.226
46 2 2 0.0000 NA NA NA

```

```

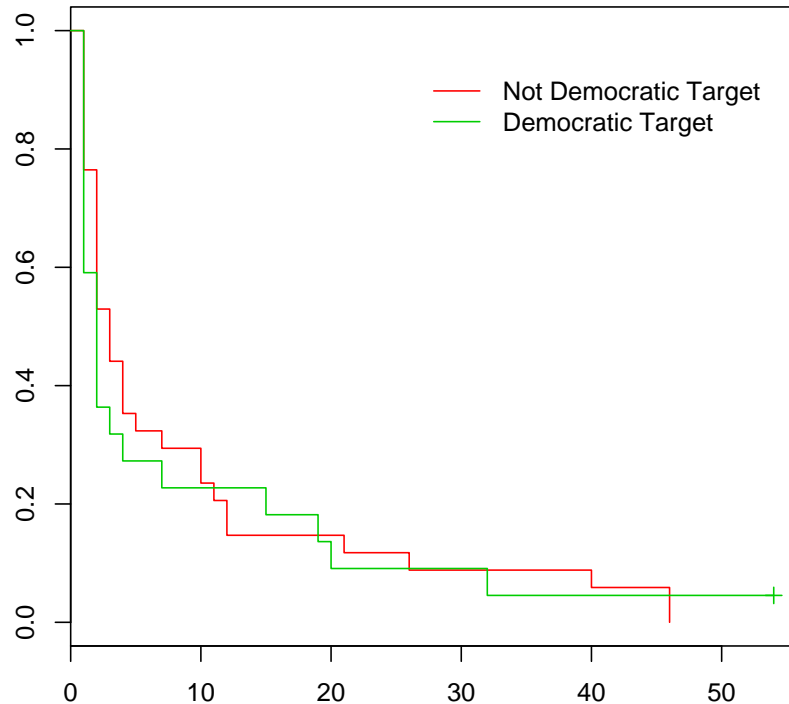
demtarg=1
time n.risk n.event survival std.err lower 95% CI upper 95% CI
1 22 9 0.5909 0.1048 0.4174 0.837
2 13 5 0.3636 0.1026 0.2092 0.632
3 8 1 0.3182 0.0993 0.1726 0.587
4 7 1 0.2727 0.0950 0.1378 0.540
7 6 1 0.2273 0.0893 0.1052 0.491
15 5 1 0.1818 0.0822 0.0749 0.441
19 4 1 0.1364 0.0732 0.0476 0.390
20 3 1 0.0909 0.0613 0.0243 0.341
32 2 1 0.0455 0.0444 0.0067 0.308

```

```

R Code
1 > plot(km2b,
2 + legend.text=c("Not Democratic Target",
3 + "Democratic Target"),
4 + legend.pos=1,
5 + col=2:3)

```



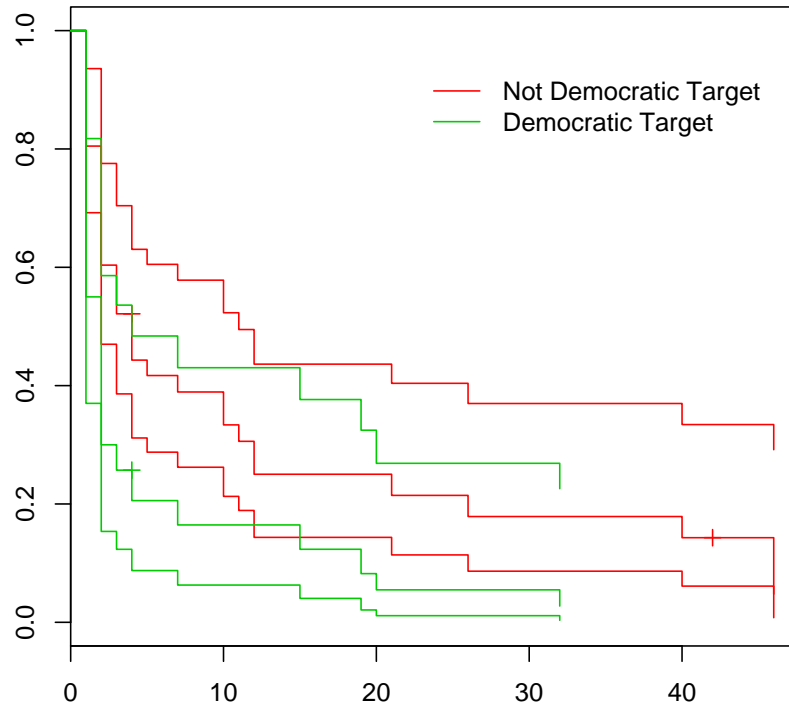
On the other hand, to the extent that democracy is time-varying, then the Cox model handles it easily, stratifying on `demtarg`: n.b., no handling of the missing data at all.

— R Code —

```

1 > cox2b <- coxph(Surv(startyear, endyear, ended) ~ strata(demtarg),
2 +               data=data)
3 > kmCox2b <- survfit(cox2b, type="kaplan-m")
4 > plot(kmCox2b,
5 +     conf.int=TRUE,
6 +     legend.pos = 1,
7 +     legend.text=c("Not Democratic Target", "Democratic Target"),
8 +     mark.time=data$endyear[lastObs & data$ended==0],
9 +     col=2:3)

```



This set of KM functions strongly suggests that sanctions against non-democratic targets are longer lived than those against democratic targets, but there is considerable uncertainty around these estimates.

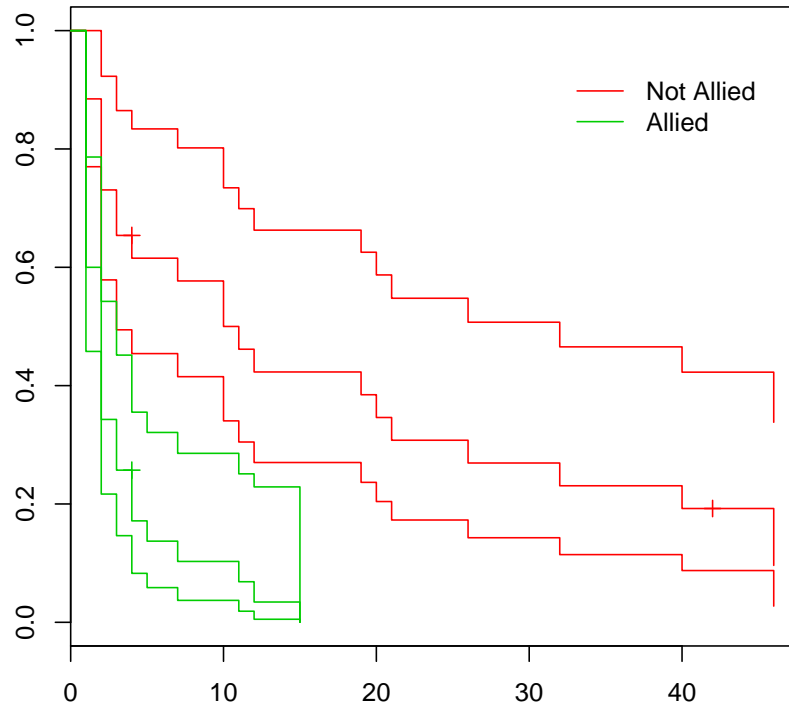
- (c) Show the Kaplan-Meier estimates of the survivor functions for sanctions for when the target country is allied with the sender country, versus when there is no such alliance. Comment briefly on what these graphs reveal.

R Code

```

1 > cox2c <- coxph(Surv(startyear,endyear,ended) ~ strata(align),
2 +               data=data)
3 > kmCox2c <- survfit(cox2c,type="kaplan-m")
4 > plot(kmCox2c,
5 +     conf.int=TRUE,
6 +     legend.text=c("Not Allied","Allied"),
7 +     legend.pos = 1,
8 +     mark.time=data$endyear[lastObs & data$ended==0],
9 +     col=2:3)

```



Here it is quite obvious that sanctions among non-allied countries last much longer than those between allied countries.

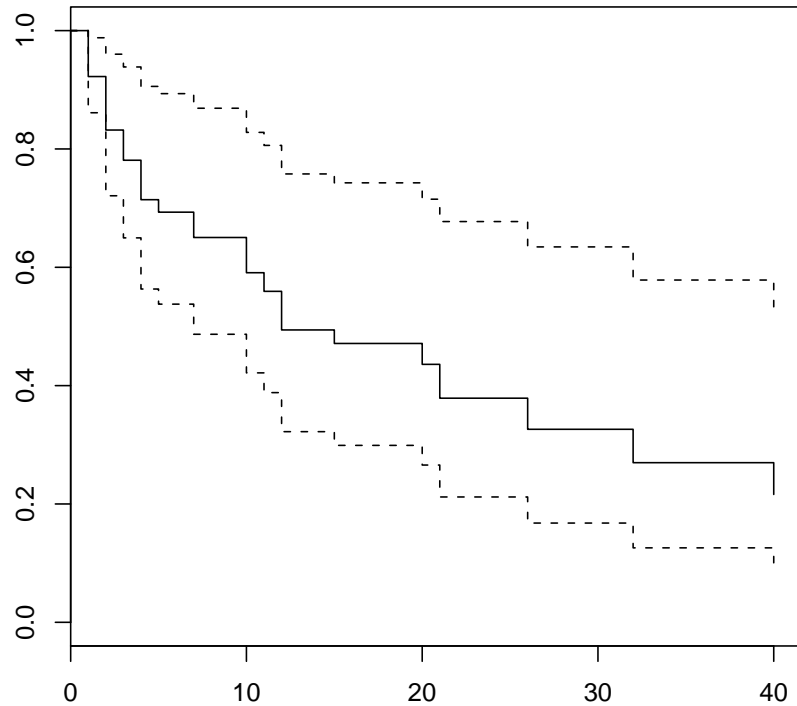
- (d) In a brief research note (no more than three pages of text), summarize the findings of the analysis. Estimate a Cox proportional hazards model for these data, using all the available predictors. What does the baseline hazard look like, at least qualitatively? What are the effects of various predictors (or combinations of predictors) on risk or expected survival time? Use whatever graphs and tables you need to convey your findings.

Answer: I fit a Cox model, and plot the fitted survivor function.

```

R Code
1 > cox2d <- coxph(Surv(startyear, endyear, ended) ~ usdummy + demtarg + align
2 + multilat + gdpctarg,
3 + data=data)
4 > ## look at survivor function for typical subject
5 > plot(survfit(cox2d),
6 + conf.int=TRUE)

```



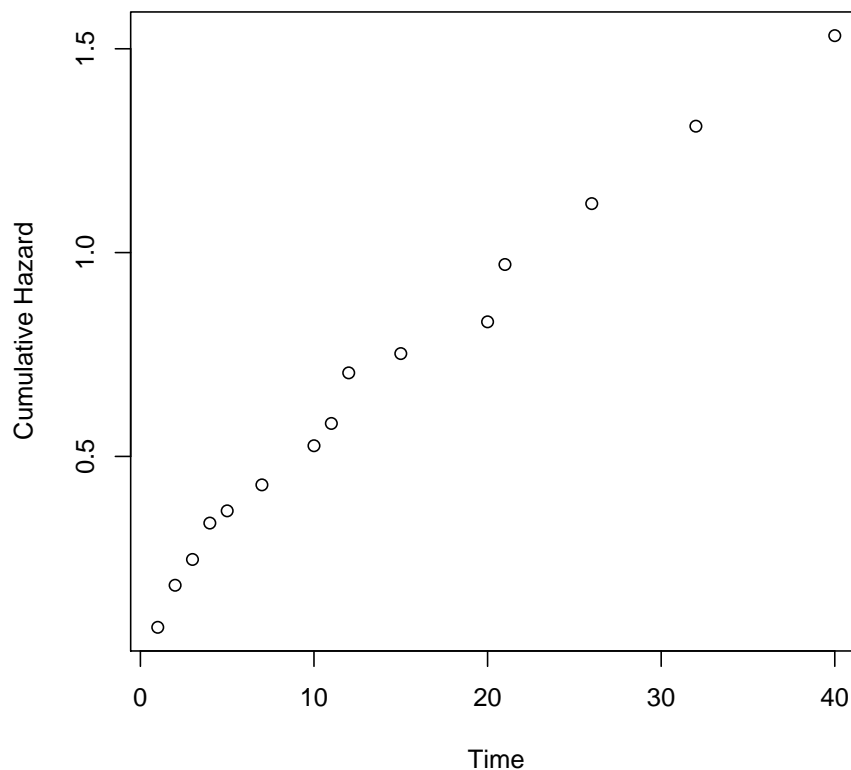
We can also recover the cumulative baseline hazard, computed with all the covariates set to their average values:

R Code

```

1 > foo <- basehaz(cox2d)  ## default is covariates at average values
2 > plot(foo$time,
3 +     foo$hazard,
4 +     xlab="Time",
5 +     ylab="Cumulative Hazard")

```



3. The file [wagepan.dta](#) contains data (Stata format) from 545 men who worked every year from 1980 to 1987. Consider the wage equation

$$\log(\text{wage}_{it}) = \theta_t + \beta_1 \text{educ}_i + \beta_2 \text{black}_i + \beta_3 \text{hispan}_i + \beta_4 \text{exper}_{it} + \beta_5 \text{exper}_{it}^2 + \beta_6 \text{married}_{it} + \beta_7 \text{union}_{it} + c_i + u_{it}$$

Notice that education is time-invariant. In the data set, the variable `lwage` is the log of the wages, while the other variables have obvious names (`educ`, `black`, `hisp`, etc). The variable `nr` is a unique identifier for each subject. `exper` is the experience variable, measured as years that a person has been in the labor market. Summarize the models you fit in the following questions in a publication-quality table.

- (a) How much of the variation in log wages is cross-sectional, and how much is longitudinal? **Answer:**

R Code

```

1 > require(foreign)
2 > wages <- read.dta(file="wagepan.dta")
3 > names(wages)

```

```

[1] "nr"      "year"    "agric"   "black"   "bus"     "construc"
[7] "ent"     "exper"   "fin"     "hisp"    "poorhlth" "hours"
[13] "manuf"   "married" "min"     "nrthcen" "nrtheast" "occ1"
[19] "occ2"    "occ3"    "occ4"    "occ5"    "occ6"    "occ7"
[25] "occ8"    "occ9"    "per"     "pro"     "pub"     "rur"
[31] "south"   "educ"    "tra"     "trad"    "union"   "lwage"
[37] "d81"     "d82"     "d83"     "d84"     "d85"     "d86"
[43] "d87"     "expersq"

```

```

_____ R Code _____
1 > between <- lm(lwage ~ as.factor(nr),
2 + data=wages)
3 > summary(between)$r.squared

```

```
[1] 0.5373721
```

```

_____ R Code _____
1 > within <- lm(lwage ~ as.factor(year),
2 + data=wages)
3 > summary(within)$r.squared

```

```
[1] 0.07518366
```

It is obvious that most the variation in these data is across subject variation.

- (b) Estimate this model by pooled OLS (i.e., ignoring the unit-specific term c_i). Are the OLS standard errors reliable, even if c_i is uncorrelated with all explanatory variables? Explain. Compute appropriate standard errors.

Answer:

```

_____ R Code _____
1 > pooled <- lm(lwage ~ as.factor(year) + educ + black + hisp +
2 + exper + expersq + married + union,
3 + data=wages)
4 > summary(pooled)

```

Call:

```
lm(formula = lwage ~ as.factor(year) + educ + black + hisp +
    exper + expersq + married + union, data = wages)
```

Residuals:

```

      Min       1Q   Median       3Q      Max
-5.26573 -0.24838  0.03192  0.29475  2.52912

```

Coefficients:

```

              Estimate Std. Error t value Pr(>|t|)
(Intercept)    0.092056   0.078270   1.176 0.239608
as.factor(year)1981  0.058320   0.030354   1.921 0.054753 .
as.factor(year)1982  0.062774   0.033214   1.890 0.058825 .
as.factor(year)1983  0.062012   0.036660   1.692 0.090807 .
as.factor(year)1984  0.090467   0.040091   2.257 0.024085 *
as.factor(year)1985  0.109246   0.043353   2.520 0.011773 *
as.factor(year)1986  0.141960   0.046423   3.058 0.002242 **
as.factor(year)1987  0.173833   0.049433   3.517 0.000442 ***
educ             0.091350   0.005237  17.442 < 2e-16 ***
black            -0.139234   0.023580  -5.905 3.80e-09 ***
hisp             0.016019   0.020797   0.770 0.441179
exper            0.067235   0.013695   4.909 9.47e-07 ***
expersq          -0.002412   0.000820  -2.941 0.003286 **
married          0.108253   0.015689   6.900 5.96e-12 ***
union            0.182461   0.017157  10.635 < 2e-16 ***
---

```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Residual standard error: 0.4803 on 4345 degrees of freedom
 Multiple R-Squared: 0.1893, Adjusted R-squared: 0.1867
 F-statistic: 72.46 on 14 and 4345 DF, p-value: < 2.2e-16

OLS ignores the possible subject-wise heteroskedasticity, i.e., the possibility that $\text{var}(u_{it}) \neq \text{var}(u_{jt})$, for some $i \neq j$. OLS also ignores the possibility of within-unit autocorrelated errors: i.e., $E(u_{it}u_{is}) \neq 0$, for some i , and for some $t \neq s$. We consider heteroskedasticity-robust and autocorrelation-robust estimates of the OLS standard errors, using the `pvcovHC` function in `library(plm)`. We consider two robust VC estimators: the `white1` option which provides a separate error variance for each cross-sectional unit, and the more general `arellano` option, which provides for a general structure with respect to heteroskedasticity and serial correlation. We first re-fit the OLS model with the `plm` function, and then pass the fitted model to the functions that compute the various robust VC estimators.

```

R Code
1 > require(plm)
2 > pdata.frame(wages, "nr", "year", name="pwages")
3 > models <- plm(lwage ~ d81 + d82 + d83 + d84 + d85 + d86 + d87 +
4 + educ + black + hisp +
5 + exper + expersq + married + union,
6 + model="pooling",
7 + data=pwages)
8 > summary(models)

```

Model Description

Oneway (individual) effect
 Pooling Model
 Model Formula : lwage ~ d81 + d82 + d83 + d84 + d85 +
 d86 + d87 + educ + black + hisp +
 exper + expersq + married + union

Panel Dimensions

Balanced Panel
 Number of Individuals : 545
 Number of Time Observations : 8
 Total Number of Observations : 4360

Residuals

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-5.27e+00	-2.48e-01	3.19e-02	1.21e-17	2.95e-01	2.53e+00

Coefficients

	Estimate	Std. Error	z-value	Pr(> z)
(intercept)	0.09205578	0.07827010	1.1761	0.2395431
d81	0.05831999	0.03035363	1.9214	0.0546875 .
d82	0.06277442	0.03321407	1.8900	0.0587586 .
d83	0.06201174	0.03666013	1.6915	0.0907354 .
d84	0.09046719	0.04009071	2.2566	0.0240354 *
d85	0.10924630	0.04335248	2.5200	0.0117370 *
d86	0.14195959	0.04642297	3.0580	0.0022285 **
d87	0.17383343	0.04943305	3.5165	0.0004372 ***
educ	0.09134979	0.00523738	17.4419	< 2.2e-16 ***
black	-0.13923421	0.02357956	-5.9049	3.529e-09 ***
hisp	0.01601951	0.02079714	0.7703	0.4411369
exper	0.06723450	0.01369484	4.9095	9.132e-07 ***
expersq	-0.00241170	0.00081995	-2.9413	0.0032688 **

```

married      0.10825295  0.01568942  6.8997 5.210e-12 ***
union       0.18246128  0.01715677 10.6349 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

----- Overall Statistics -----
Total Sum of Squares      : 1236.5
Residual Sum of Squares   : 1002.5
Rsq                       : 0.18928
F                         : 72.4588
P(F>0)                   : 1.44547e-11

```

```

----- R Code -----
1 > require(lmtest)
2 > coeftest(models,vcov=function(x)pvcovHC(x,type="white1"))

```

t test of coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(intercept)	0.09205578	0.07932045	1.1606	0.2458865
d81	0.05831999	0.03203042	1.8208	0.0687108 .
d82	0.06277442	0.03413723	1.8389	0.0660004 .
d83	0.06201174	0.03636620	1.7052	0.0882281 .
d84	0.09046719	0.04103813	2.2045	0.0275438 *
d85	0.10924630	0.04388407	2.4894	0.0128320 *
d86	0.14195959	0.04708063	3.0152	0.0025826 **
d87	0.17383343	0.04896102	3.5504	0.0003887 ***
educ	0.09134979	0.00528218	17.2939	< 2.2e-16 ***
black	-0.13923421	0.02450289	-5.6824	1.415e-08 ***
hispanic	0.01601951	0.01968545	0.8138	0.4158191
exper	0.06723450	0.01326572	5.0683	4.181e-07 ***
expersq	-0.00241170	0.00076884	-3.1368	0.0017194 **
married	0.10825295	0.01522147	7.1119	1.333e-12 ***
union	0.18246128	0.01625551	11.2246	< 2.2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

----- R Code -----
1 > coeftest(models,vcov=function(x)pvcovHC(x,type="arellano"))

```

t test of coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(intercept)	0.0920558	0.1605304	0.5734	0.5663712
d81	0.0583200	0.0281568	2.0713	0.0383936 *
d82	0.0627744	0.0368802	1.7021	0.0888046 .
d83	0.0620117	0.0461313	1.3442	0.1789397
d84	0.0904672	0.0578417	1.5640	0.1178790
d85	0.1092463	0.0666787	1.6384	0.1014111
d86	0.1419596	0.0760424	1.8668	0.0619902 .
d87	0.1738334	0.0849906	2.0453	0.0408828 *
educ	0.0913498	0.0110542	8.2638	< 2.2e-16 ***
black	-0.1392342	0.0503963	-2.7628	0.0057551 **
hispanic	0.0160195	0.0389795	0.4110	0.6811130
exper	0.0672345	0.0195464	3.4397	0.0005877 ***
expersq	-0.0024117	0.0010226	-2.3584	0.0183993 *
married	0.1082529	0.0259683	4.1687	3.124e-05 ***
union	0.1824613	0.0273742	6.6654	2.969e-11 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

The use of the Arellano robust VC estimator produces considerably smaller standard errors than either pooled OLS or the White robust VC estimator,

though none of the estimated t -statistics cross a critical threshold with either estimator.

- (c) Estimate the wage equation treating the c_i as “random effects”. Compare your estimates with the pooled OLS estimates.

Answer: plm was being unhelpful for this, so I used lme in library(nlme):

```

R Code
1 > require(nlme)
2 > re <- lme(lwage ~ d81 + d82 + d83 + d84 + d85 + d86 + d87 +
3 +       educ + black + hisp +
4 +       exper + expersq + married + union,
5 +       random = ~ 1 | nr,
6 +       data=wages)
7 > summary(re)

```

Linear mixed-effects model fit by REML

Data: wages
 AIC BIC logLik
 4507.075 4615.48 -2236.537

Random effects:
 Formula: ~1 | nr
 (Intercept) Residual
 StdDev: 0.331687 0.3511073

Fixed effects: lwage ~ d81 + d82 + d83 + d84 + d85 + d86 + d87 + educ + black + hisp + exper

	Value	Std.Error	DF	t-value	p-value
(Intercept)	0.02306038	0.15306813	3804	0.150654	0.8803
d81	0.04034397	0.02473202	3804	1.631244	0.1029
d82	0.03070747	0.03255533	3804	0.943239	0.3456
d83	0.01999917	0.04198927	3804	0.476292	0.6339
d84	0.04279589	0.05192106	3804	0.824249	0.4098
d85	0.05744986	0.06203603	3804	0.926072	0.3545
d86	0.09158051	0.07223291	3804	1.267850	0.2049
d87	0.13464692	0.08251683	3804	1.631751	0.1028
educ	0.09188953	0.01083351	541	8.481970	0.0000
black	-0.13938308	0.04849707	541	-2.874052	0.0042
hisp	0.02178418	0.04330356	541	0.503057	0.6151
exper	0.10603825	0.01548829	3804	6.846349	0.0000
expersq	-0.00474006	0.00068847	3804	-6.884914	0.0000
married	0.06346140	0.01678561	3804	3.780703	0.0002
union	0.10531873	0.01786082	3804	5.896635	0.0000

Correlation:

	(Intr)	d81	d82	d83	d84	d85	d86	d87	educ	black
d81		0.282								
d82		0.469	0.667							
d83		0.557	0.655	0.817						
d84		0.605	0.636	0.821	0.889					
d85		0.632	0.616	0.814	0.892	0.926				
d86		0.649	0.596	0.802	0.887	0.926	0.947			
d87		0.659	0.577	0.787	0.877	0.921	0.946	0.960		
educ		-0.960	-0.249	-0.380	-0.444	-0.482	-0.507	-0.525	-0.538	
black		-0.038	0.032	0.049	0.056	0.061	0.063	0.065	0.066	0.016
hisp		-0.184	0.003	0.005	0.007	0.008	0.009	0.010	0.012	0.163
exper		-0.588	-0.485	-0.710	-0.792	-0.819	-0.820	-0.807	-0.785	0.390
expersq		0.120	0.226	0.302	0.302	0.274	0.232	0.182	0.128	0.039
married		0.049	-0.014	-0.007	-0.014	-0.008	-0.002	0.001	0.002	-0.049
union		-0.009	0.018	0.020	0.028	0.025	0.032	0.033	0.019	-0.058

d81
 d82
 d83

```

d84
d85
d86
d87
educ
black
hisp
exper   -0.001
expersq -0.015 -0.682
married  0.006 -0.115  0.081
union   -0.024 -0.034  0.034 -0.034

Standardized Within-Group Residuals:
      Min          Q1          Med          Q3          Max
-12.24292387  -0.34809092   0.05419966   0.46199003   4.29614070

Number of Observations: 4360
Number of Groups: 545

```

- (d) Now estimate the equation by fixed effects. Why is $exper_{it}$ redundant in the model even though it changes over time? What happens to the marriage and union wage premiums as compared with their corresponding random effects estimates?

Answer: Note that with fixed effects (using the “within” variation), coefficients over time-invariant variables are unidentified. Thus we can’t estimate coefficients on *educ*, *black*, *hisp*. We also can’t estimate a coefficient on *exper* due to the presence of the fixed effects for year (i.e., experience is colinear with time).

I fit the model with `plm` and the `twoways` option (fixed effects for both individual and time):

```

_____ R Code _____
1 > fe <- plm(lwage ~ expersq + married + union,
2 +       model="within",
3 +       effect="twoways",
4 +       data=pwages)
5 > summary(fe)

```

```

_____ Model Description _____
Twoways effects
Within Model
Model Formula      : lwage ~ expersq + married + union

_____ Panel Dimensions _____
Balanced Panel
Number of Individuals      : 545
Number of Time Observations : 8
Total Number of Observations : 4360

_____ Residuals _____
      Min.   1st Qu.   Median     Mean   3rd Qu.    Max.
-4.16e+00 -1.25e-01  1.13e-02 -1.89e-17  1.55e-01  1.49e+00

_____ Coefficients _____
      Estimate Std. Error z-value Pr(>|z|)
expersq -0.0051855  0.0006583 -7.8771 3.331e-15 ***
married  0.0466804  0.0171113  2.7280 0.006371 **
union    0.0800019  0.0180457  4.4333 9.280e-06 ***

```

```

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

	Overall Statistics
Total Sum of Squares	: 479.09
Residual Sum of Squares	: 468.75
Rsq	: 0.021568
F	: 27.959
P(F>0)	: 0.00905434

(e) What does a Hausman test say about the plausibility of the random effects assumption?

Answer: To do this I estimated the following model by fixed and random effects, and implemented the Hausman test accordingly.

```

R Code
1 > models <- plm(lwage ~ d81 + d82 + d83 + d84 + d85 + d86 + d87 +
2 + expersq + married + union,
3 + data=pwages)
4 > summary(models)

```

Model Description

Oneway (individual) effect

Model Formula : lwage ~ d81 + d82 + d83 + d84 + d85 +
d86 + d87 + expersq + married + union

Panel Dimensions

Balanced Panel
Number of Individuals : 545
Number of Time Observations : 8
Total Number of Observations : 4360

	within	wse	random	rse
(intercept)	.	.	1.38998437	0.0227
d81	0.15119121	0.02052797	0.13362558	0.0217
d82	0.25297086	0.02283762	0.21456153	0.0228
d83	0.35444374	0.02734876	0.29090406	0.0251
d84	0.49011479	0.03388132	0.39810581	0.0288
d85	0.61748227	0.04231448	0.49410474	0.0338
d86	0.76549657	0.05249406	0.60647753	0.0401
d87	0.92502493	0.06432077	0.72461617	0.0477
expersq	-0.00518550	0.00065883	-0.00313866	0.0005
married	0.04668036	0.01712503	0.07803392	0.0168
union	0.08000186	0.01806017	0.10397363	0.0180

Tests

Hausman Test : chi2(10) = 315.4914 (p.value=0)
F Test : F(544,3805) = 9.156772 (p.value=0)
Lagrange Multiplier Test : chi2(1) = 3745.325 (p.value=0)

We overwhelmingly reject the null of the Hausman test, and would prefer the fixed effects estimator in this case.

(f) Estimate an education effect specific to each year. How well does this model fit the data compared to the model with a time-invariant education

effect? Has the return to education increased over time? Offer a substantive interpretation of the returns to education.

Answer: Apologies: this isn't a well posed question, since the only way we can estimate an education effect is with the pooling model or with a random effects model, and we just got a strong message from the data that fixed effects is preferred to random effects, and that fixed effects is preferred to the pooling model (see the *F* test).

For the sake of answering the question, let's go back to random effects:

```

R Code
1 > re2 <- lme(lwage ~ as.factor(year)*educ + black + hisp
2 +           + exper + expersq + married + union,
3 +           random = ~ 1 | nr,
4 +           data=wages)
5 > summary(re2)$tTable

```

	Value	Std.Error	DF	t-value
(Intercept)	-0.0308471563	0.1888270627	3797	-0.16336195
as.factor(year)1981	-0.0287793851	0.1455050660	3797	-0.19778957
as.factor(year)1982	-0.0100370492	0.1470552672	3797	-0.06825359
as.factor(year)1983	0.0177508539	0.1498403747	3797	0.11846509
as.factor(year)1984	0.1133786607	0.1540346095	3797	0.73605965
as.factor(year)1985	0.1172454894	0.1598159641	3797	0.73362815
as.factor(year)1986	0.1793951624	0.1672944401	3797	1.07233189
as.factor(year)1987	0.2562971376	0.1766042948	3797	1.45125088
educ	0.0946598854	0.0137547507	541	6.88197756
black	-0.1396152088	0.0485457677	541	-2.87595017
hisp	0.0224139969	0.0433490356	541	0.51705872
exper	0.1156006859	0.0170741514	3797	6.77050843
expersq	-0.0053693945	0.0008346633	3797	-6.43300659
married	0.0639564436	0.0168019530	3797	3.80648867
union	0.1043563356	0.0178845934	3797	5.83498508
as.factor(year)1981:educ	0.0054331827	0.0122234645	3797	0.44448795
as.factor(year)1982:educ	0.0026893826	0.0123247892	3797	0.21820922
as.factor(year)1983:educ	-0.0008096129	0.0124927814	3797	-0.06480645
as.factor(year)1984:educ	-0.0071166490	0.0127269521	3797	-0.55917937
as.factor(year)1985:educ	-0.0062127005	0.0130194655	3797	-0.47718553
as.factor(year)1986:educ	-0.0084982408	0.0133664103	3797	-0.63579081
as.factor(year)1987:educ	-0.0111660284	0.0137684241	3797	-0.81098812
	p-value			
(Intercept)	8.702422e-01			
as.factor(year)1981	8.432203e-01			
as.factor(year)1982	9.455874e-01			
as.factor(year)1983	9.057054e-01			
as.factor(year)1984	4.617399e-01			
as.factor(year)1985	4.632207e-01			
as.factor(year)1986	2.836391e-01			
as.factor(year)1987	1.467926e-01			
educ	1.639146e-11			
black	4.186900e-03			
hisp	6.053265e-01			
exper	1.480873e-11			
expersq	1.406760e-10			
married	1.431771e-04			
union	5.829219e-09			
as.factor(year)1981:educ	6.567152e-01			
as.factor(year)1982:educ	8.272778e-01			
as.factor(year)1983:educ	9.483315e-01			
as.factor(year)1984:educ	5.760723e-01			
as.factor(year)1985:educ	6.332575e-01			

```

as.factor(year)1986:educ 5.249511e-01
as.factor(year)1987:educ 4.174233e-01
R Code _____
1 > AIC(re)
[1] 4507.075
R Code _____
1 > AIC(re2)
[1] 4569.896

```

There is very little evidence to suggest that the education effects are varying by year. The improvement in fit from estimating the time-specific education effects is not worth the 7 degrees of freedom consumed: the AIC comparisons suggest the model with the time invariant education effects is the preferred specification.

4. Suppose that we have the unobserved effects model

$$y_{it} = \alpha + \mathbf{x}_{it}\boldsymbol{\beta} + \mathbf{z}_i\boldsymbol{\gamma} + h_i + u_{it}$$

where the \mathbf{x}_{it} are time-varying, the \mathbf{z}_i are time-constant, $E(u_{it}|\mathbf{x}_{it}, \mathbf{z}_i, h_i) = 0$, $t = 1, \dots, T$ and $E(h_i|\mathbf{x}_i, \mathbf{z}_i) = 0$. Let $\sigma_h^2 = \text{var}(h_i)$ and $\sigma_u^2 = \text{var}(u_{it})$. If we estimate $\boldsymbol{\beta}$ by fixed effects, we are estimating the equation

$$y_{it} = \mathbf{x}_{it}\boldsymbol{\beta} + c_i + u_{it}$$

where $c_i = \alpha + \mathbf{z}_i\boldsymbol{\gamma} + h_i$.

- (a) Find $\sigma_c^2 = \text{var}(c_i)$. Show that σ_c^2 is at least as large as σ_h^2 and usually strictly larger.

Answer: Since $c_i = \alpha + \mathbf{z}_i\boldsymbol{\gamma} + h_i$,

$$\begin{aligned}
\sigma_c^2 &= \text{var}(\alpha + \mathbf{z}_i\boldsymbol{\gamma} + h_i) \\
&= \text{var}(\mathbf{z}_i\boldsymbol{\gamma} + h_i) \\
&= \boldsymbol{\gamma}'\text{var}(\mathbf{z}_i)\boldsymbol{\gamma} + \text{var}(h_i) + 2\boldsymbol{\gamma}'\text{cov}(\mathbf{z}_i, h_i) \\
&= \boldsymbol{\gamma}'\text{var}(\mathbf{z}_i)\boldsymbol{\gamma} + \text{var}(h_i) \\
&\quad \text{since the assumption } E(h_i|\mathbf{x}_i, \mathbf{z}_i) = 0 \Rightarrow \text{cov}(\mathbf{z}_i, h_i) = \mathbf{0} \\
&\geq \text{var}(h_i) \equiv \sigma_h^2
\end{aligned}$$

with a strong inequality for non-degenerate \mathbf{z}_i .

- (b) Explain why estimation of the model by fixed effects will lead to a larger estimated variance of the unobserved effect than if we estimate the model by random effects.

Answer: The question is relatively easy to handle if we consider the \mathbf{z}_i as observed, time-invariant covariates. By treating the unit-specific effects as

random effects we can include the time-invariant predictors in the model, meaning we could estimate

$$y_{it} = \mathbf{x}_{it}\boldsymbol{\beta} + \mathbf{z}_i\boldsymbol{\gamma} + \alpha + \varepsilon_{it}$$

where $\varepsilon_{it} = h_i + u_{it}$ and has variance $\sigma_h^2 + \sigma_u^2$. With this model the variance of the unit-specific term is just $\text{var}(h_i) = \sigma_h^2$ which in the previous question we showed to be generally smaller than σ_c^2 .

5. The data frame `bacteria` in the R package `MASS` contains data from a drug trial administered to children with a history of otitis media (infection of the middle ear). Fifty children participated in 5 regular screenings for the presence of *H. influenzae* (bacteria responsible for a wide range of diseases, including some nasties like acute bacterial meningitis, but more run of the mill stuff like conjunctivitis, sinusitis, and ear infections such as otitis media). Children were randomly assigned to one of four conditions at the initial screening in a crossed experimental design: (a) a drug or a placebo; (b) encouragement to take their assigned medication, or no such encouragement. The data frame has 220 observations; not all children were monitored at all 5 time points. The variables are

- `y`: presence or absence of *H. influenzae*, a factor with levels `n` and `y`;
- `ap`: active drug or placebo, a factor with levels `a` and `p`;
- `hilo`: hi/low encouragement to comply with treatment, a factor with levels `hi` and `lo`;
- `week`: numeric, the week of the screening;
- `ID`: subject ID, a factor;
- `trt`: a factor with levels `placebo`, `drug`, `drug+`, a re-coding of `ap` and `hilo`

We are interested in the efficacy of the treatments, taking into account the fact that we don't expect to see any response to treatment until the second screening (since treatments were assigned at the initial screening).

(a) Provide some kind of assessment as to how much of the variation in these data is “between”, and how much is “within”.

Answer: This is a little rough and ready, using OLS regression to look at these sources of variation:

```
----- R Code -----
1 > require(MASS)
2 > data(bacteria)
3 > ynum <- as.numeric(bacteria$y=="y")
4 > within <- lm(ynum ~ as.factor(week), data=bacteria)
5 > summary(within)$r.squared
```

```
[1] 0.04734063
```

```
_____ R Code _____  
1 > between <- lm(ynum ~ ID,data=bacteria)  
2 > summary(between)$r.squared
```

```
[1] 0.3457715
```

- (b) Estimate a logit model for these data with y as the (binary) response, and the treatment variables as the only predictors. Briefly comment on what you find.

```
_____ R Code _____  
1 > logit1 <- glm(ynum ~ ap*hilo,  
2 + data=bacteria,  
3 + family=binomial)  
4 > summary(logit1)
```

Call:

```
glm(formula = ynum ~ ap * hilo, family = binomial, data = bacteria)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-2.1460	0.4590	0.6039	0.6860	0.8282

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	1.3269	0.3120	4.253	2.11e-05 ***
app	0.8704	0.5315	1.638	0.102
hilolo	-0.4331	0.4191	-1.033	0.301
app:hilolo	-0.1547	0.7488	-0.207	0.836

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 217.38 on 219 degrees of freedom
Residual deviance: 209.83 on 216 degrees of freedom
AIC: 217.83

Number of Fisher Scoring iterations: 4

```
_____ R Code _____  
1 > logit2 <- update(logit1, ~ trt)  
2 > anova(logit2,logit1,test="Chi")
```

Analysis of Deviance Table

```
Model 1: ynum ~ trt  
Model 2: ynum ~ ap * hilo  
  Resid. Df Resid. Dev  Df Deviance P(>|Chi|)  
1      217    210.720  
2      216    209.831  1    0.890    0.346
```

This naive analysis doesn't provide a lot of support for the hypothesis that the treatments work, but of course, this includes the baseline values. If we exclude the initial observations:

```
_____ R Code _____  
1 > logit1 <- glm(ynum ~ ap*hilo,  
2 + data=bacteria,  
3 + subset=(week>0),  
4 + family=binomial)  
5 > summary(logit1)
```

```
Call:
glm(formula = ynum ~ ap * hilo, family = binomial, data = bacteria,
     subset = (week > 0))
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1169   0.4743   0.6272   0.7679   0.9005
```

```
Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.0704     0.3345   3.200  0.00137 **
app            1.0578     0.5794   1.826  0.06790 .
hilolo        -0.3773     0.4535  -0.832  0.40542
app:hilolo    -0.2249     0.8203  -0.274  0.78398
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(Dispersion parameter for binomial family taken to be 1)

```
Null deviance: 180.66 on 169 degrees of freedom
Residual deviance: 172.64 on 166 degrees of freedom
AIC: 180.64
```

```
Number of Fisher Scoring iterations: 4
R Code
```

```
1 > logit2 <- update(logit1, ~ trt)
2 > anova(logit2, logit1, test="Chi")
```

Analysis of Deviance Table

```
Model 1: ynum ~ trt
Model 2: ynum ~ ap * hilo
  Resid. Df Resid. Dev Df Deviance P(>|Chi|)
1      167      173.41
2      166      172.64  1     0.77     0.38
```

Note that the test of the two models is a test of whether we require coefficients for all four possible experimental conditions (including the “placebo”+ “encouraged” condition), versus the three level coding that collapses the two placebo conditions. These models fit better.

- (c) Estimate the unconditional transition probabilities for y , week by week. That is, at each week (other than week 0) what is the probability of testing positive for *H. influenzae* given the status of a child’s test in the previous week. Hint: this is nothing other than computing 2-by-2 tables, but you have to set them up the right way (or rather, set up the lagged variables the right way).
- (d) A plausible rival hypothesis is *maturation*: that over time, the children’s immune systems would attack the bacteria, and absent treatment. What does your answer to the previous question suggest about this possibility?
- (e) Augment your logit model with some kind of control for this maturation hypothesis. Does this alter your conclusion about the effects of the treatments? What specification of time dependence is best supported by the data: a linear time trend (on the log-odds scale), or fixed effects for each time period, something else, or nothing?

Answer: We fit some alternative models to the data.

```

R Code
1 > ## fixed effects by time for each treatment regime
2 > logit1 <- glm(ynum ~ trt*as.factor(week),
3 +             data=bacteria,
4 +             family=binomial)
5 > ## setp function after treatment begins for each treatment
6 > logit2 <- glm(ynum ~ trt*I(week>0),
7 +             data=bacteria,
8 +             family=binomial)
9 > ## interactive, linear trend for each treatment
10 > logit3 <- glm(ynum ~ trt*week,
11 +            data=bacteria,
12 +            family=binomial)
13 > ## additive, fixed effects for week
14 > logit1a <- glm(ynum ~ trt + as.factor(week),
15 +            data=bacteria,
16 +            family=binomial)
17 > ## additive, step function after treatment begins
18 > logit2a <- glm(ynum ~ trt + I(week>0),
19 +            data=bacteria,
20 +            family=binomial)
21 > ## additive, linear time trend
22 > logit3a <- glm(ynum ~ trt + week,
23 +            data=bacteria,
24 +            family=binomial)
25 > models <- list(logit1,logit2,logit3,
26 +              logit1a,logit2a,logit3a)
27 > lapply(models,extractAIC)
```

```
[[1]]
[1] 15.0000 223.6624
```

```
[[2]]
[1] 6.0000 217.4481
```

```
[[3]]
[1] 6.0000 215.1206
```

```
[[4]]
[1] 7.0000 213.1259
```

```
[[5]]
[1] 4.0000 214.2877
```

```
[[6]]
[1] 4.0000 211.8061
```

AIC prefers the last model fitted, with a linear time trend constant across treatments:

```

R Code
1 > summary(logit3a)
```

```
Call:
glm(formula = ynum ~ trt + week, family = binomial, data = bacteria)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2899  0.3885  0.5400  0.7027  1.1077
```

```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.54629    0.40555   6.279 3.42e-10 ***
```

```

trtdrug      -1.10667    0.42519   -2.603   0.00925 **
trtdrug+     -0.65166    0.44615   -1.461   0.14412
week         -0.11577    0.04414   -2.623   0.00872 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 217.38  on 219  degrees of freedom
Residual deviance: 203.81  on 216  degrees of freedom
AIC: 211.81

Number of Fisher Scoring iterations: 4

```

- (f) The data available for analysis have no individual level covariates, but it is plausible that unobserved subject-level heterogeneity is a factor in these data. Comment on this possibility, keeping in mind that we have an experiment.

Answer: This is surely a possibility in these data. Since we have an experiment, subject-level factors ought to be uncorrelated with treatment status, and so our estimates of the treatment effects that ignore the subject-level factors are unbiased. On the other hand, we can probably get some efficiency gains from random effects estimation. We also note that the random subject-level effects are uncorrelated with the treatment effects, so estimates of the treatment effects remain consistent after we implement the random effects estimator (or at least this is the way the theory works with in the linear case; note that here we have a non-linear, logistic regression set up).

- (g) The R package `glmmML` has a function with the same name for fitting GLMs with random intercepts (“random effects”). The argument `cluster` in the `glmmML` function specifies the grouping variable for the random effects (i.e., in this case, ID). Use this function to augment your preferred logit model from the previous steps.

Answer:

```

----- R Code -----
1 > require(glmmML)
2 > mod <- glmmML(ynum ~ trt + week,
3 +           cluster=ID,
4 +           data=bacteria,
5 +           family=binomial)
6 > summary(mod)

```

Call: `glmmML(formula = ynum ~ trt + week, family = binomial, data = bacteria, cluster = ID)`

```

              coef se(coef)      z Pr(>|z|)
(Intercept)  3.1440  0.61916  5.078 3.82e-07
trtdrug      -1.3202  0.64199 -2.056 3.97e-02
trtdrug+     -0.7955  0.65172 -1.221 2.22e-01
week         -0.1437  0.05092 -2.822 4.77e-03

```

```

Standard deviation in mixing distribution: 1.147
Std. Error:                               0.3782

```

The results don't change that much. We obtain a slightly larger estimates of the treatment effects with random effects (-1.32 versus -1.10 for the drug/placebo; and an additional -.80 versus -.65 for the encouragement effect), and also a larger estimate of the time effects, but these larger estimates generally come with larger standard errors. The standard deviation of the random effects distribution is quite large, 1.14 (on the log-odds scale), which indicates a lot of subject-to-subject heterogeneity.

- (h) Why is random effects appropriate here? How do your results change, if at all, via the introduction of the random effects?

Answer: See the answers to parts f and g.

- (i) Provide a short statement as to the efficacy of the treatments deployed in this case. In a graph or two, show how the predicted probability of having *H. influenzae* changes over time, as a function of the two overlapping treatments. Augment your graphs with confidence intervals etc.

Answer:

— R Code —

```

1  > x <- rbind(
2  +       c(1,0,0,0), ## placebo week zero
3  +       c(1,0,0,2), ## placebo week two
4  +       c(1,0,0,4), ## placebo week four
5  +       c(1,0,0,6), ## placebo week six
6  +       c(1,0,0,11), ## placebo week eleven
7  +       c(1,1,0,0), ## trtdrug week zero
8  +       c(1,1,0,2), ## trtdrug week two
9  +       c(1,1,0,4), ## trtdrug week four
10 +       c(1,1,0,6), ## trtdrug week six
11 +       c(1,1,0,11), ## trtdrug week eleven
12 +       c(1,0,1,0), ## trtdrug+ week zero
13 +       c(1,0,1,2), ## trtdrug+ week two
14 +       c(1,0,1,4), ## trtdrug+ week four
15 +       c(1,0,1,6), ## trtdrug+ week six
16 +       c(1,0,1,11) ## trtdrug+ week eleven
17 +     )
18 > b <- coef(mod)
19 > yhat <- x%*%b
20 > phat <- 1/(1+exp(-yhat))
21 > vc <- mod$variance[-5,-5] ## vc matrix of beta
22 > v <- x%*%vc%*%t(x) ## vc of predictions
23 > se <- sqrt(diag(v)) ## std errors
24 > bounds <- cbind(yhat - 1.96*se,
25 +               yhat + 1.96*se)
26 > phatBounds <- 1/(1+exp(-bounds))
27 > ## a plotting function we will use
28 > plotFunc <- function(x,y,bounds,xlab){
29 +   plot(x,y,
30 +       xlab=xlab,
31 +       ylab="Probability",
32 +       ylim=c(0,1),
33 +       axes=FALSE)
34 +   axis(2)
35 +   segments(x0=x,
36 +           x1=x,
```

```

37 +             y0=bounds[,1],
38 +             y1=bounds[,2])
39 +   invisible(NULL)
40 + }
41 > ## look by treatment
42 > par(mfrow=c(3,1),
43 +     las=1)
44 > plotFunc(x=c(0,2,4,6,11),
45 +         y=phat[1:5],
46 +         phatBounds[1:5,],
47 +         xlab="Week")
48 > axis(1)
49 > title("Placebo")
50 > plotFunc(x=c(0,2,4,6,11),
51 +         y=phat[6:10],
52 +         phatBounds[6:10,],
53 +         xlab="Week")
54 > axis(1)
55 > title("Treatment")
56 > plotFunc(x=c(0,2,4,6,11),
57 +         y=phat[11:15],
58 +         phatBounds[11:15,],
59 +         xlab="Week")
60 > axis(1)
61 > title("Treatment Plus")

```

See Figure 1. The treatment effect is apparent, but so too are the large confidence intervals around each point estimate.

END OF EXAM

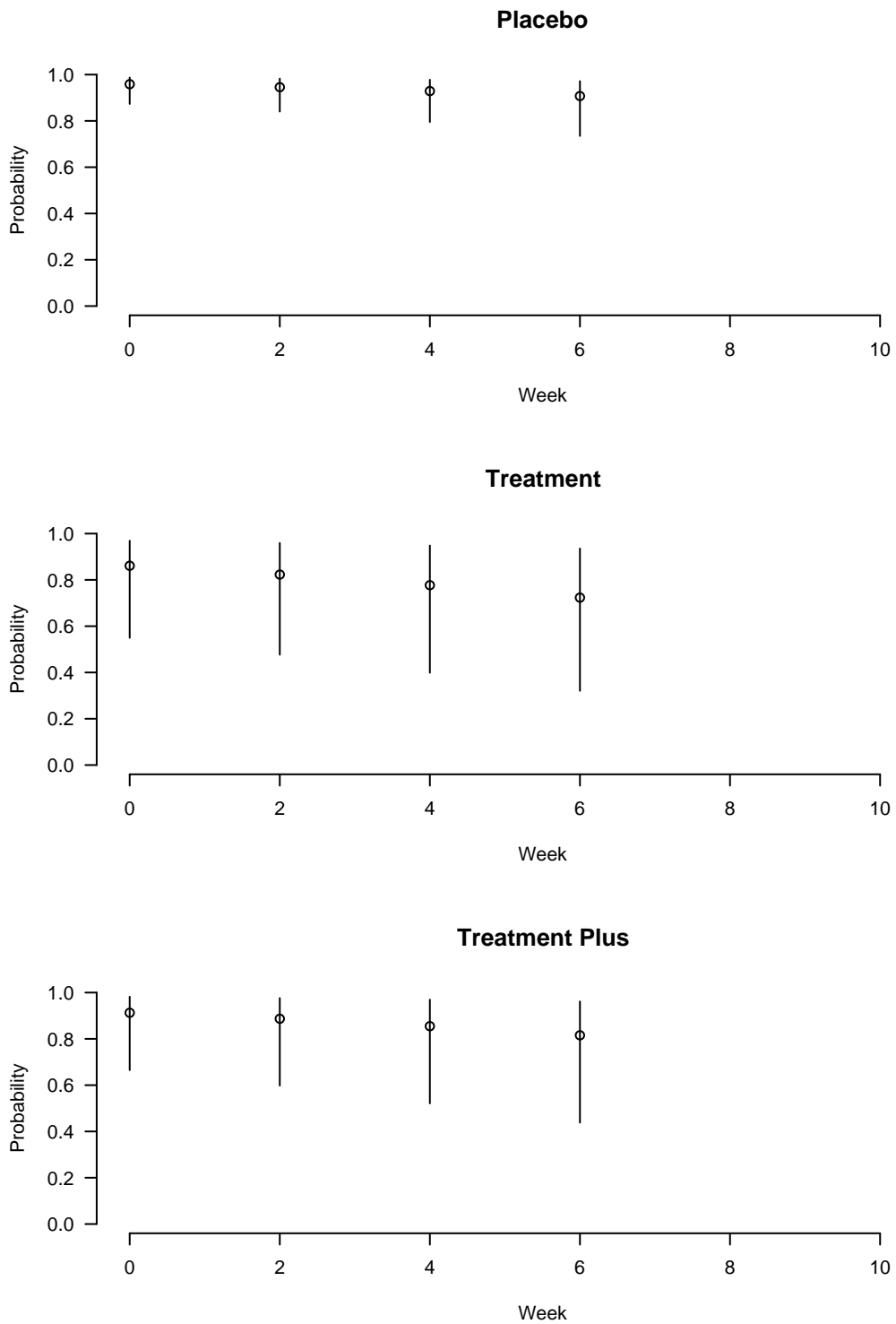


Figure 1: Probability of testing positive for *H. influenzae*, as a function of week of screening and treatment status; lines cover 95% confidence intervals.