Graduate School Class Reminders

- Maintain six feet of distancing
- Please sit in the same chair each class time
- Observe entry/exit doors as marked
- Use hand sanitizer when you enter/exit the classroom
- Use a disinfectant wipe/spray to wipe down your learning space before and after class
- Media Services: 414 955-4357 option 2

Documentation on the web

- CRAN: `http://cran.r-project.org`
- R manuals: `https://cran.r-project.org/manuals.html`
- SAS: `http://support.sas.com/documentation`
- SAS 9.3: `https://support.sas.com/en/documentation/documentation-for-SAS-93-and-earlier.html`
- Step-by-Step Programming with Base SAS 9.4 (SbS): `https://documentation.sas.com/api/docsets/basess/9.4/content/basess.pdf`
- SAS 9.4 Programmer s Guide: Essentials (PGE): `https://documentation.sas.com/api/docsets/lepg/9.4/content/lepg.pdf`
- Wiki: `https://wiki.biostat.mcw.edu` (MCW/VPN)

Concatenating data sets

- Concatenating data sets: creates a single data set from a series of data sets in the order which they appear *stacking* them one on top of another, etc.
- `data NEW; set OLD1 ... OLDn; run;`

  in a diagram, `NEW` looks like this:

  `OLD1`
  ⋮
  `OLDn`
- For large data sets, use `proc append`
- see `NTDB/sas/traumactr.sas`

Interleaving data sets

- Interleaving data sets: similar to concatenation
  but their observations appear in the order of the `by` clause
  the order the data sets appear does NOT matter
  (except for ties of the `by` clause variables)
- `data NEW; set OLD1 ... OLDn; by VAR1 ... VARm; run;`
- see `NTDB/sas/traumactr.sas`

Merging data sets

- Merging two or more data sets requires some more consideration than concatenation/interleaving
- `data NEW; merge OLD1 ... OLDn;`
  `by VAR1 ... VARm; run;`
- Each of the data sets `OLD1 ... OLDn` must be sorted according to the `by` clause

Merging data sets

- Generally, each of the data sets EXCEPT ONE should have unique keys according to the `by` clause
- You can tell if a key is unique by the last variable in the clause the automatic variables `first.VARm=last.VARm=1`
- One data set (typically the last for program readability) does not have to be unique
  there may be multiple observations for each key
- The data set option `in` can be useful since NOT every data set will necessarily contribute observations
- `merge OLD1(in=NAME1) ... OLDn(in=NAMEn);`
  these automatic variables are 1 for contributed variables to a particular observation and 0 otherwise
- see `PTB/merge.sas`

Summarizing data sets: START HERE

- To calculate summaries, these are the PROCs commonly used `proc freq`, `proc means` and `proc univariate`
- They all accept `by` and `where` statements
- `proc freq` for categorical variables
- Common `OPTION1`: `order=freq` by descending frequency
- Common `OPTION2`: `list` for multi-way tables and `missing` to create a missing category

```
proc freq OPTION1 data=NAME; *for display in .lst;
tables Z Y*X / OPTION2;
run;
proc freq NOPRINT OPTION1 data=NAME; *not in .lst;
tables Y*X / OPTION2 out=NEW1; *saved to a data set;
run;
proc freq noprint OPTION1 data=NAME;
tables Z / OPTION2 out=NEW2;
run;
```

Summarizing data sets with `proc means`

- Some overlap of `means` and `univariate` for continuous summaries
- `means` mainly for simple statistics
  `minid/maxid` captures information about min/max
- `univariate` mainly for more complex summaries like quantiles, hypothesis tests and `histrograms` which we will see later

```
proc means data=NAME; *for display in .lst;
var x y z;
run;

proc means noprint data=NAME; *summaries in a new data set;
var VAR1 ... VARn;
output out=NEW STAT=NAME1 ... NAMEn
       minid(VARi(XVAR1 ... XVARk))=XVAL1 ... XVALk
       maxid(VARj(YVAR1 ... YVARm))=YVAL1 ... YVALm;
run;
```

Summarizing data sets with `proc means` and `proc univariate`

```
proc means OPTIONS data=NAME;
where ...; * subsetting data set;
by ...;    * for summaries of each by-group;
class ...; * by-like summaries for unsorted data sets;
var VAR1 ... VARn;
output out=NEW STAT1=X1 ... Xn ... STATm=Y1 ... Yn;
run;

proc univariate OPTIONS data=NAME;
where ...; * subsetting data set;
by ...;    * for summaries of each by-group;
class ...; * by-like summaries for unsorted data sets;
var VAR1 ... VARn;
output out=NEW STAT1=X1 ... Xn ... STATm=Y1 ... Yn;
run;
```

Summarizing data sets with `proc corr`

- Correlation is a very important summary for pairs of variables
- `proc corr` computes correlation for all possible pairs
- `proc corr PEARSON` is the default assuming Normality
  `proc corr PEARSON outp=NEW` to output them to a data set
- `proc corr SPEARMAN` for nonparametric correlations of the ranks rather than the values themselves
  `proc corr SPEARMAN outs=NEW` to output them

```
proc corr OPTIONS data=NAME;
var VAR1 ... VARn;
run;
```

Example: Electronic health records (EHR)

Context: Diabetes and recurrent hospital admissions

- We have IRB approval to study a cohort of newly diagnosed diabetes patients from a single health care system
- We have the electronic health records (EHR) for these patients from 2007-2012: prior records may, or may not, be available
- EHR are an omnibus of digital health care information
- We focus on 82 covariates: patient demographics, health insurance, health care charges, diagnoses, procedures, anti-diabetic therapy, laboratory values and vital signs
- By its nature, EHR data is fundamentally time-varying
- EHR covariates are occasionally missing even when carrying the last value forward
- Imputed 15 continuous variables with Sequential BART (Xu, Daniels & Winterstein 2016 *Biostatistics*)

Electronic health records (EHR)

Diabetes and recurrent hospital admissions

- 488 patients followed 5 years from 2008-2012
  the survival rate was high 0.939 (noninformative censoring)
  yet experienced a high rate of hospital admissions: 525 total
- For diabetes, which covariates increase the risk of admission? What about the number of previous admissions or an acutely recent admission?
- What are the functional forms of the covariates i.e. linear, quadratic, logarithm, etc.? Are the covariate effects additive or multiplicative?
- Are there interactions? Are these effects constant with respect to time, i.e., proportionality assumption?
- We want to avoid precarious restrictive assumptions hence we chose to use Bayesian Additive Regression Trees (BART)

Electronic health records (EHR)

Diabetes and recurrent hospital admissions

	Patients		Admissions	
Number of Admissions	488		525	
0	308	(63.0)	0	
1	79	(16.2)	79	(15.0)
2-3	50	(10.3)	115	(21.9)
4-16	51	(10.5)	331	(63.1)

EHR: Diabetes and recurrent hospital admissions

	Patients		Admissions	
Gender	488		525	
M	216	(44.3)	228	(43.4)
F	272	(55.7)	297	(56.6)
Race	488		525	
Black	174	(35.7)	265	(50.5)
White	314	(64.3)	260	(49.5)
Age	488		525	
Mean, SD	60.9	15.0	60.3	15.7
ZIP3 area	488		525	
532/urban	378	(77.5)	454	(86.5)
530/suburb	110	(22.5)	71	(13.5)
Insurance and Age	488		525	
Government 65+	191	(39.1)	224	(42.7)
Government <65	138	(28.3)	208	(39.6)
Commercial <65	143	(29.3)	71	(13.5)
Other <65	16	(3.3)	22	(4.2)

EHR: Diabetes and recurrent hospital admissions

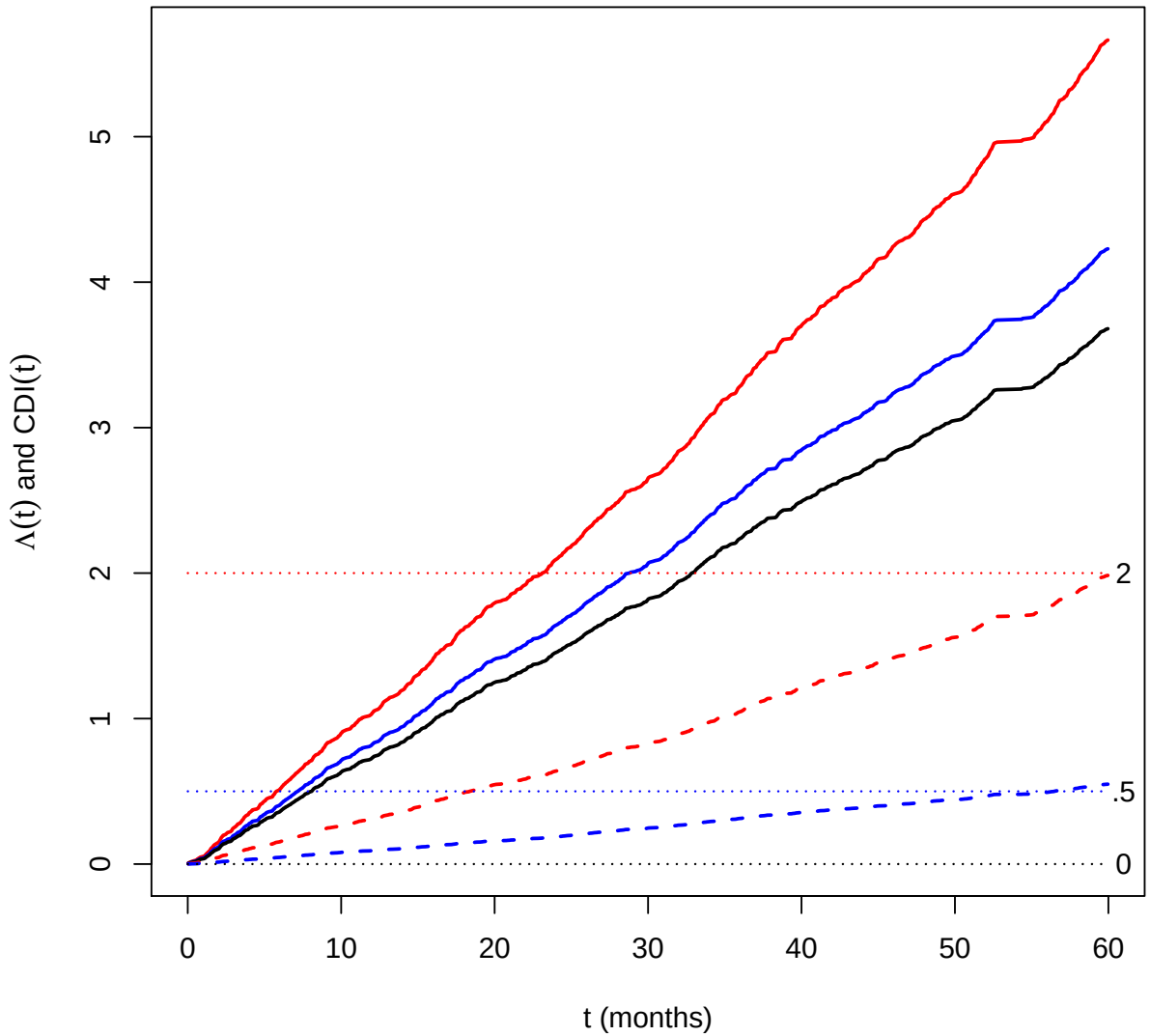
	Patients		Admissions		Relative Intensity	95% Credible Interval
Insulin	488		525		2.39	1.56, 3.25
Yes	206	(42.2)	391	(74.5)		
No	282	(57.8)	134	(25.5)		
PVD	488		525		2.90	2.00, 3.89
Yes	272	(55.7)	488	(93.0)		
No	216	(44.3)	37	(7.0)		

partial dependence function

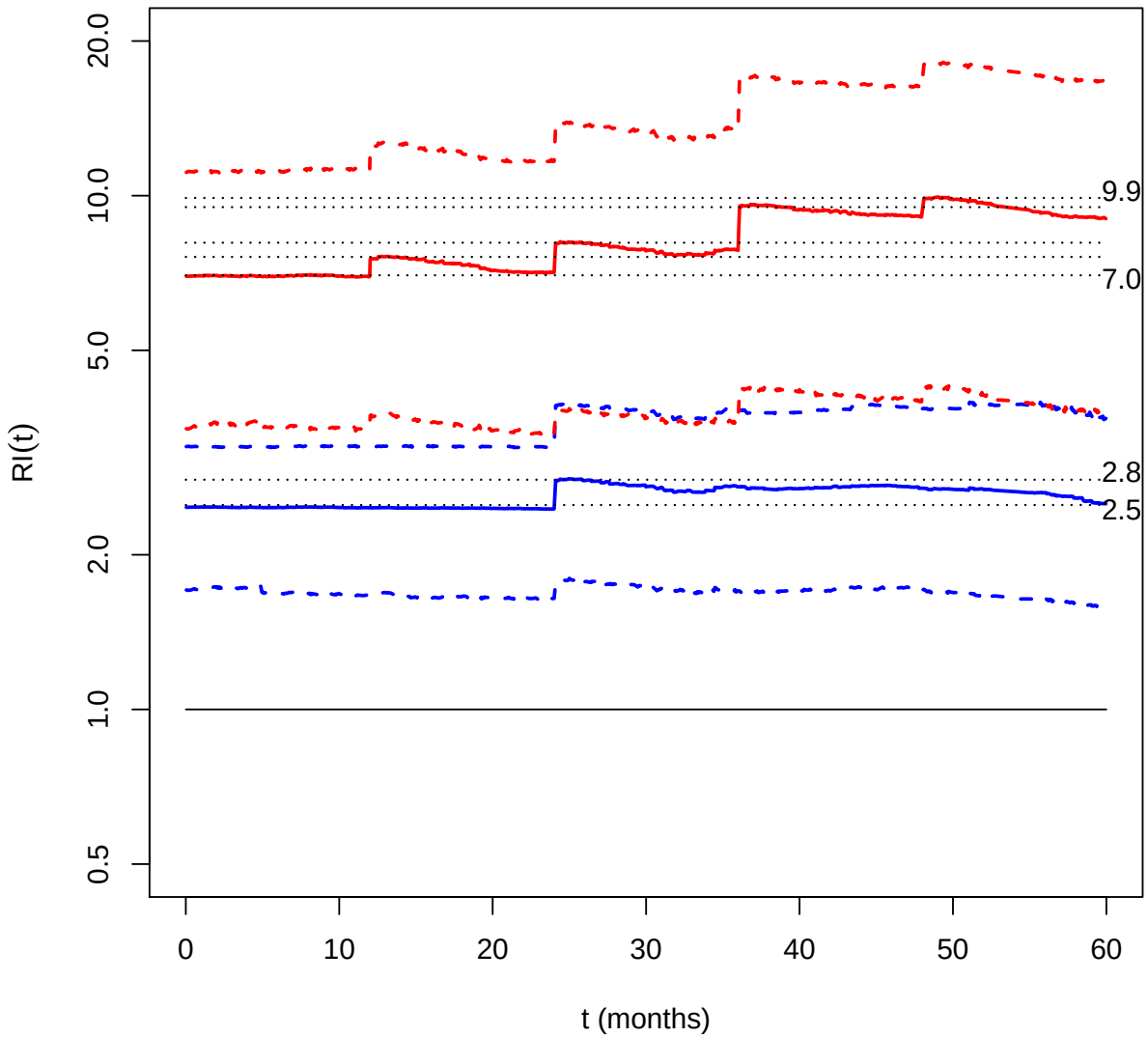
EHR: Hospital admission risk profiles

Risk	Insulin	PVD	$N_i(t)$ with time in months 0	12	24	36	48	60
Low	0	0	0	0	0	0	0	0
Medium	1	0	0	0	1	1	1	1
High	1	1	0	1	2	3	4	4

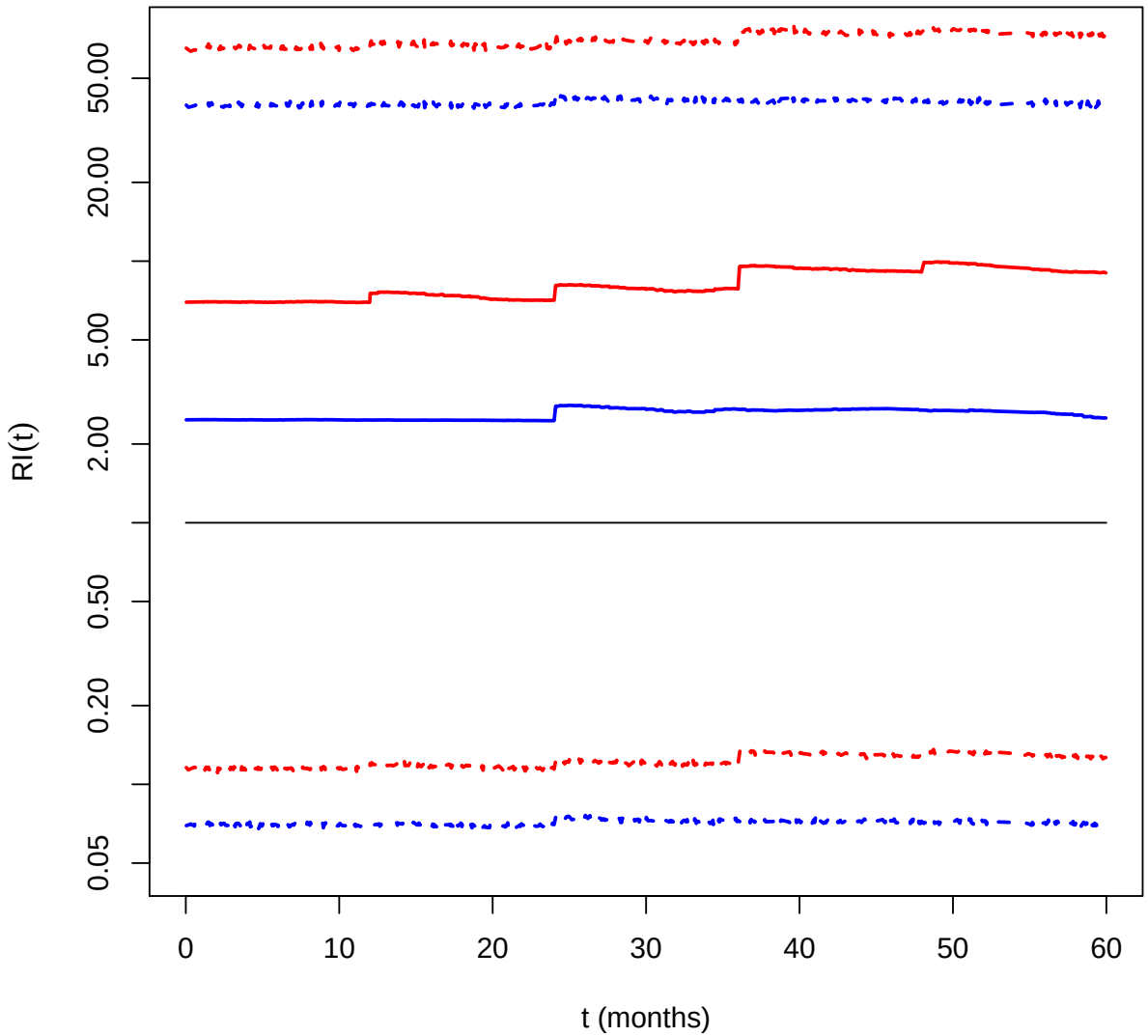
EHR: Risk profiles: Cumulative Intensity partial dependence function



Risk profiles: Relative Intensity and 95% Credible Intervals partial dependence function



Risk profiles: Relative Intensity & 95% Prediction Intervals partial dependence function



EHR: Diabetes and hospital admission risk

- Some diabetes patients are at high risk for hospital admission
  - diagnosed with PVD
  - prescribed insulin therapy
  - with a recent hospital admission
  - and/or several previous hospital admissions
- Health policy implications: Diabetic patients' health care post-discharge should be carefully orchestrated to ensure the delivery of quality clinical care which maximizes healthy outcomes while preventing adverse events and costly unnecessary hospital admissions
- **BART** package contains a roughly 20% random sample
  50 patients from training: `ydm20.train` & `xdm20.train`
  50 patients from validation: `xdm20.test`
- See example: `system.file('demo/dm.recur.bart.R', package='BART')`
- complete data set at `http://www.mcw.edu/FileLibrary/Groups/Biostatistics/TechReports/TechReports5175/tr064.tar`
- `tr064.tar` copied to `/data/shared/04224/EHR`

Cumulative intensity and recurrent events

- In this example, we are ignoring the impact of covariates we are interested in the experience of diabetes patients in aggregate rather than individually
- With the discrete time approach, divide the time line into a grid based on when events were observed
  $0 = t_{(0)} < t_{(1)} < \cdots < t_{(K)} < \infty$ where $t_{(j)}$ are the distinct event times observed
- Suppose we count the number of events in each interval
  k_j is the number of events found in the interval $(t_{(j-1)}, t_{(j)}]$
- The *intensity* of an event falling in the interval $(t_{(j-1)}, t_{(j)}]$ is the probability $p_j = k_j/N$ where N is the number of patients (so few patients died in this study that we can simply ignore them: in other cases you may not be able to)
- And the cumulative intensity by time $t_{(j)}$ is just the sum of these probabilities

$$C_j = \sum_{h=1}^{j} p_h = m_j/N \text{ where } m_j = \sum_{h=1}^{j} k_h$$ number of cumulative events

HW EHR part 1: Cumulative intensity and recurrent events

- N.B. don't confuse the cumulative intensity with a cumulative distribution function (CDF) or a cumulative incidence function e.g., cumulative intensity is not restricted to the interval $(0, 1)$ like the others, i.e., its not a probability
- Calculate the cumulative intensity function based on the 20% sample: see `EHR/sas/dm20.sas`

Variable	Description
`id`	Unique patient identifier: $i = 1, \ldots, N = 100$ (not a unique key)
`t`	The *study day* at the end of the interval: $j = 1, \ldots, K = 798$ `id` and `t` taken together are the unique key
`y`	Was an event observed within this interval? 0 or 1
`n`	the number of *previous* events that have been observed prior to this interval

For each patient, $m_{ij} = y_{ij} + n_{ij}$ cumulative events by time $t_{(j)}$
Similarly, for the whole sample, $m_{+j} = \sum_{i=1}^{N} m_{ij}$
$C_j = N^{-1} m_{+j} = N^{-1} \sum_{i=1}^{N} m_{ij}$