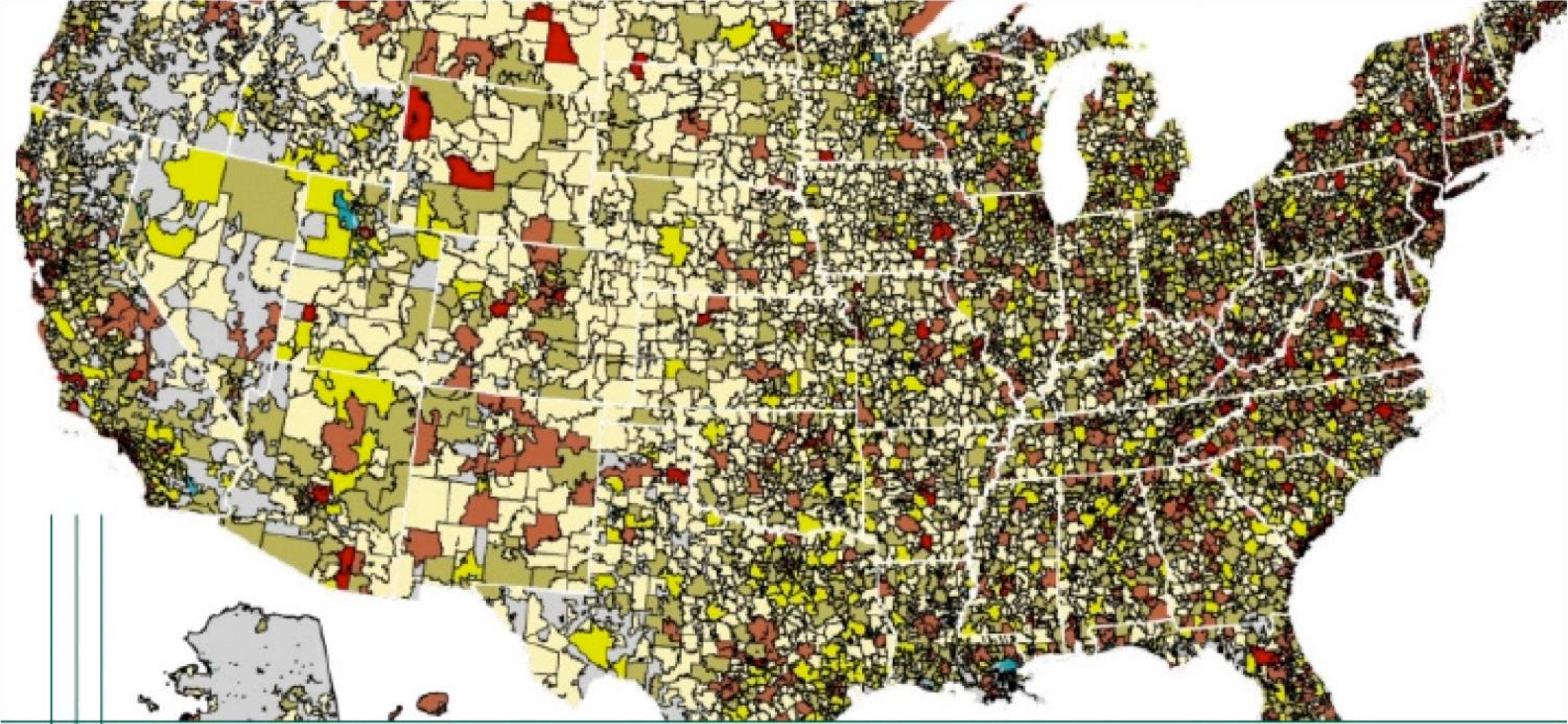
WHERE YOUR MIND REALLY IS



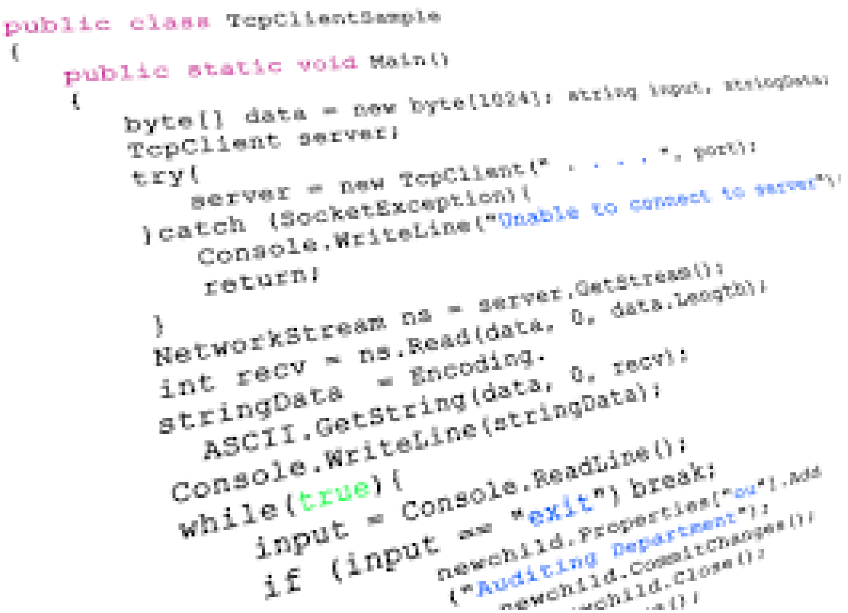
MANY YEARS OF CODING

What I’ve learned (or still need to…)

July 15, 2016

DART-SUG

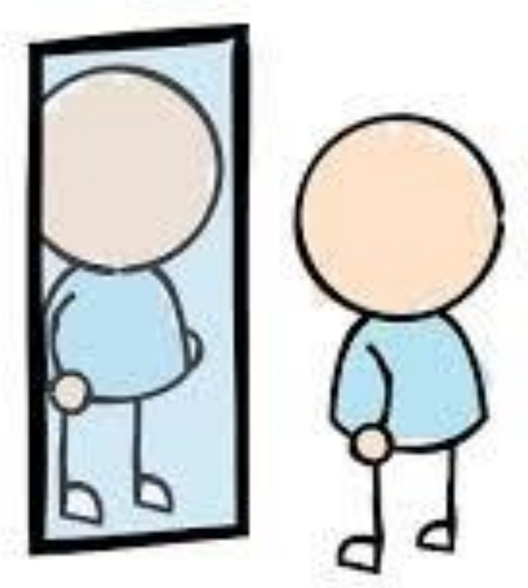
INTRO



```
public class TcpClientSample
{
    public static void Main()
    {
        byte[] data = new byte[1024]; string input, stringData;
        TcpClient server;
        try{
            server = new TcpClient(" . . . . ", port);
        }catch (SocketException){
            Console.WriteLine("Unable to connect to server");
            return;
        }
        NetworkStream ns = server.GetStream();
        int recv = ns.Read(data, 0, data.Length);
        stringData  = Encoding.
            ASCII.GetString(data, 0, recv);
        Console.WriteLine(stringData);
        while(true){
            input = Console.ReadLine();
            if (input == "exit") break;
                newchild.Properties["ou"].Add
                ("Auditing Department");
                newchild.CommitChanges();
                newchild.Close();
```

THE Dartmouth INSTITUTE FOR HEALTH POLICY & CLINICAL PRACTICE GEISEL SCHOOL OF MEDICINE AT DARTMOUTH

INTRO



INTRO

A FEW TRUISMS

- Specifications are flexible

A FEW TRUISMS

- There are N! ways to solve N questions

$$
\begin{aligned}
1! &= 1 \\
2! &= 2(1) = 2 \\
3! &= 3(2)(1) = 6 \\
4! &= 4(3)(2)(1) = 24 \\
5! &= 5(4)(3)(2)(1) = 120
\end{aligned}
$$

A FEW TRUISMS

- The journey can be more rewarding than the finish line
-

A FEW TRUISMS

- Hard work helps

A FEW TRUISMS

- Just because the program ran w/o syntax errors does not mean the results are correct!

A FEW TRUISMS

- You’ll get bored and lose focus

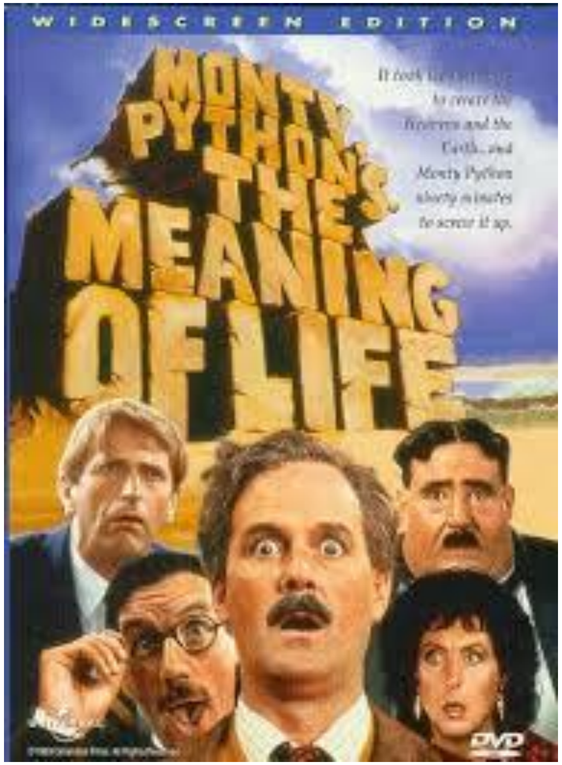
STAYING PRODUCTIVE

- Read a SAS global paper
- Write documentation
- Review a manuscript
- Draft a methods section
- Go for a walk

STAYING PRODUCTIVE

Love what you do

We are what we do



SO WHEN ARE YOU GOING TO START TALKING DAN…

Tips and Hints

1. Determine what’s feasible
2. Adjust the specifications
3. Break the project into components and lay them out
   1. Modularization allows for team work.
   2. Allows for independent testing and work to be done in parallel
   3. Describing what you’re doing often yields incites and inefficiencies.
4. If you’re on the wrong path, STOP

" I do good work . . unfortunately. I don't do it here. "

TIPS AND TRICKS

- Try Enterprise Guide if possible
  - Look at all those options!
- Use a log checker
- Name programs using numbers and names, data sets the same way.
  - P10_create_cohort creates d10_ami2006_2010_&sysdate (or &ver)

TIPS AND TRICKS

- Use data set labels (256 characters)
  - Path/program name (&syprocessname) and basic info (exclusions, # of recs/person, unit of observation (e.g. 1 observation per person age 65-99). Use if %length(&label)>256 %then %let label=%substr(&label,1,256)
- Auto label variables/use variable labels
  - Label dx_test="&&test&l &daysp prior to &dayspost post &outcome"
  - Consider short variable names and longer labels.

TIPS AND TRICKS

- When creating analytic files, keep all observations and flag instead of removing
  - Create multiple flags (ex_parta ex_partb ex_age,etc) exclude=sum(of ex_:); use dictionary tables to auto label.
  - Easier to run sensitivity analyses
  - You might be able to use the file for another project with a different cohort.
  - Create an exclusion tree for QC/publication
  - Often identifies problems (e.g. excluding 75% of the initial cohort).

TIPS AND TRICKS

- Look for a solution online before coding a complex algorithm/macro.
- Expand your toolkit- STATA and R can do some things better/simpler
- Consider counting process methods when working with temporal data.

NAMING CONVENTIONS

- Consider UPPERCASE for provided data, first letter uppercase for derivation, lowercase for created variables
- SEX, Female, spending_l6m_life

COUNTING PROCESS

- Question: Identify the number of people taking drug A and B within a month of a hospitalization or during a hospitalization.
  - Scripts are often for weeks.
  - Hospitalizations are multiple days.

  CP method: create events (QC each event separately)

```
BENE_ID  STARTDATE     ENDDATE       EVENT         EDATE         DAYS
      1  2/27/2009     12/31/2010    HOSPSTAY 3/21/2009     6
      1  2/27/2009     12/31/2010    BETABLOCKER   3/25/2009     90
      1  2/27/2009     12/31/2010    CACHBLOCKER   3/25/2009     30
      1  2/27/2009     12/31/2010    REHAB         3/27/2009     14
      1  2/27/2009     12/31/2010    CACHBLOCKER   4/20/2009     30
```

COUNTING PROCESS

- Data events; set events;
- If event=‘HOSP’ then do;
- *** now a 30 day hosp lookback – change the edate too ;
    - Event=”prehosp30”;
    - Days=30;
    - EDATE=edate-30;
    - OUTPUT;
- End;

```
Run;
```

```
BENE_ID STARTDATE       WINSTART WINEND         ENDDATE           BETABLOCKER CACHBLOCKER HOSPSTAY
      1     02/27/2009  20090227 20090321       12/31/2010  0           0           0           0
      1     02/27/2009  20090321 20090325       12/31/2010  0           0           1           0
      1     02/27/2009  20090325 20090327       12/31/2010  1           1           1           0
      1     02/27/2009  20090327 20090410       12/31/2010  1           1           0           1
      1     02/27/2009  20090410 20090520       12/31/2010  1           1           0           0
      1     02/27/2009  20090520 20090623       12/31/2010  1           0           0           0
      1     02/27/2009  20090623 20101231       12/31/2010  0           0           0           0
```

QC

- Create ‘self checking’ code
- Don’t go over the top!
- if _n_<1000 then put “QC in loop” var1= var2=; in data step
- Create short summaries at multiple steps and describe expectations in title statements.

QC

- Create histograms for time associated data or big data.
  - – Proc sgplot;
  - – Histogram date;
  - – Format date mmyy.
  - – Proc sgpanel;
  - – Panelby age / rows=2 columns=2;
  - – Format age agefive. ;
  - – Histogram date;

QC

- Create output that can be checked by a non-SAS person.

```
Proc means nmiss min max mean median maxdec=2 nway; class gender; var bp;
```

Title should not have any missing values. Values should be similar by gender. Should be >30 and <300

QC

- Self-checking version
- Proc format; value badbp
- ., low-50,250-high=‘bad’ other=‘OK’
- %macro qc;

```
Data qc;
    Set data (keep=bp end=fin);  if put(bp,badbp.)=‘bad’ then badbp+1;
    If fin then output;
    Run;
    Proc sql;
    Select badbp into :badbp from qc;
    Quit;
    %if &badpb>0 %then %do;
    %put &badbp obs with invalid blood pressure- fix or flag for deletion- quitting;
    %abort abend;
    %mend;
```

HOW WE GET TRIPPED UP

- Merging- remove observations or create duplicates.

```
%macro prepost(pre,post,id=bene);
Proc sql; create table _pre as select distinct &id, count(*) as nrecs_pre from &pre group by &id order by &id;
Create table _post as select distinct &id, count(*) as nrecs_post from &post group by &id order by &id;
Data _compare;  merge _pre (in=inpre) _post (in=inpost); by &id;
If _inpre or _inpost; Run;
Proc sql; select count(*) as inpre_notpost as select count(*) as n_inpre , sum(nrecs_pre) as total_recs from _compare where missing(nrecs_post);
```

Same for post->pre

HOW WE GET TRIPPED UP

- Missing values
  - Often the result of one sided joins (or simply missing data).
  - Exacerbated by many SAS procedures not showing results

```
Proc tabulate; class gender; var age female;
Proc tabulate; class gender /missing; var age female;
Tables gender, (age female)*mean*f=7.2;
```

HOW WE GET TRIPPED UP

- Public analytic files often use odd conventions for missing/refused

  -77 = missing, -88=refused

  Check (or create) codebooks.

A SIMPLE MACRO TO CHECK NUMERIC VALUES

```
%macro checkvals(datain,outcomes=,covs=,byvar=);
* Can use dictionary tables to separate into character and numeric;
%if &byvar= %then %let byvar2=%*;
%else %let byvar=class &byvar;
Proc means %if &byvar ne %then nway; nmiss min max p1 p99 mean median maxdec=2;
&byvar2;
Title QC of outcome &outcome and covariates &covs from data &datain;
Var &outcomes &covs;
Run;
%mend;
```

USE FORMATS SMARTLY

- Add actual value to the label- reduce errors (create 2 versions, one with and one w/o values)

  Proc format;

  Value place 11=‘11:office’ instead of

  11=‘office’

  To modify compiled formats:

  Proc format library=formats.medpfmts cntlout=fmts;

USING FORMATS SMARTLY

- As an alternative to merging (big data)
- Proc sql; create table ppl as select distinct bene_id, ‘$hipfx’ as fmtname, ‘YES’ as label from cohort where event=‘hip’;
- Proc format cntlin=ppl; run;

```
Data claims;
    Set bigdata;
    Where put(bene_id,$hipfx.)=‘YES’;
    Run;
```

USING FORMATS SMARTLY

- For suppression

```
Proc format; value suppress 1-11=‘<11:suppress’;
Proc print;  var hrr nppl; format nppl suppress. ;
```

As an alternative to creating new variables..

```
Proc tabulate;
Class age;
Format age agefive.;
Var spending;
Tables age all , spending*mean
```

FORMAT LIBRARIES

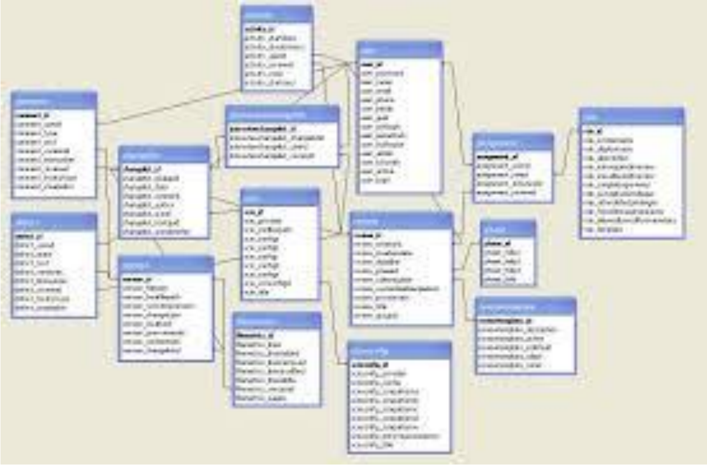
- Save sets of related formats to individual format libraries, save excel versions to easy review

```
Proc format library=fmts.denom;
Value $sex ‘1=‘1:male’ ‘2’=‘2:female’ other=‘other’;
Value $race ‘1’=‘1:white’ 2=‘2:black’;
Proc format library=fmts.denom cntlout=fmts; proc sort; by fmtname;
Ods excel file=‘denomfmts.xlsx’ options(index=‘on’
sheet_interval=‘byvar’ embedded_titles=‘on’)
proc print noobs;  title format libe fmts.denom created &sysdate;
by fmtname;
Var fmtname start end label;
```

FORMATS

- Make them readily available
- Options append=(fmtsearch=(fmts.denom fmts.medpar myfmts.pac));
- Use as an alternative to

Table links when working
With database data



USE THE EXCEL DESTINATION

- No more clunky, oversized XML files.
- Allows a mix of graphics and text.
- Name tabs
- Allows titles
- Can stoplight values
- Create a single report ‘package’

EXCEL DESTINATION

- Add an index
- Paste program into a tab.
- Include proc contents of analytic file.
- Date label file name (e.g. p110_poisson_&sysdate..xlsx)

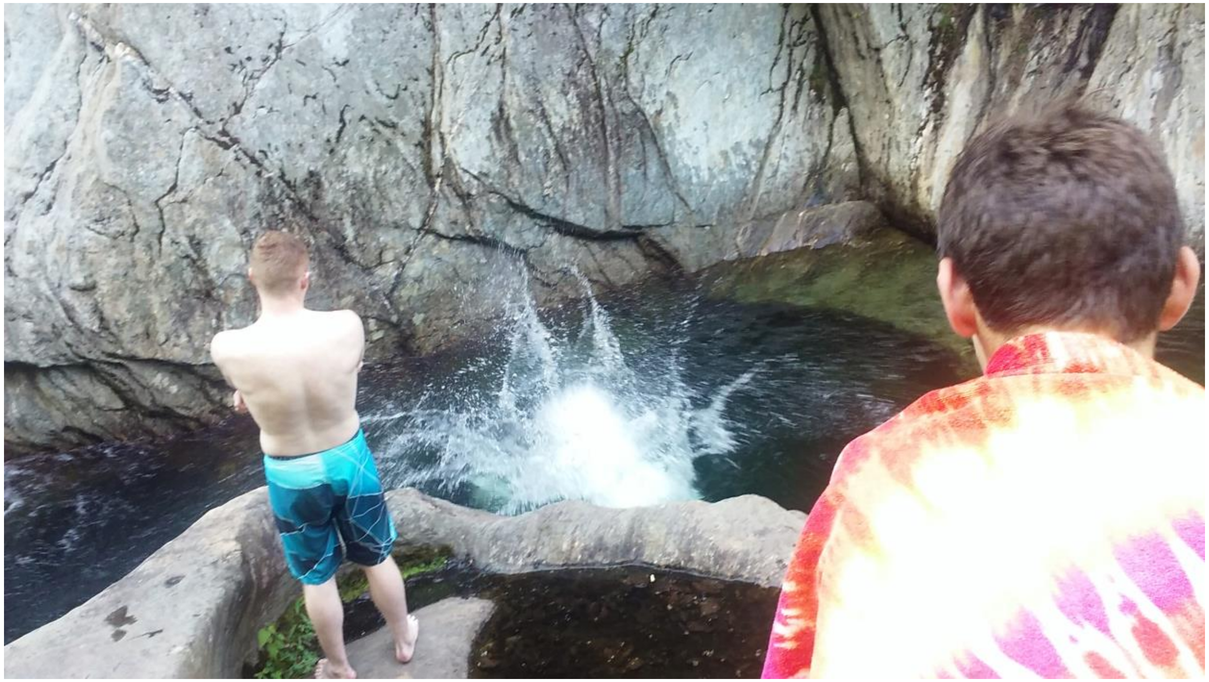
EXCEL

```
ods excel file="./output/p110_poisson_WPRAC_&oc._&sysdate..xlsx"
options (sheet_interval='none' sheet_name="input file"  embedded_titles='on'
index='on');

%macro savexls(mod,descript,num=93,random=Y,or=N);
proc contents data= save.d45d_fullcoh2  ;
title input file used;
run;
ods excel options (sheet_name="overall mod1 &mod &lab" sheet_interval='proc' );
data descript;
info="Run &sysdate  &saspath outcome &lab";
output;
info="all models fixed effects straight poisson just 3 races main ACO effects.  save practice fixed effects";
output;
info="&descript "; output;
info="has practice level effects";
run;
proc print noobs;
title summary info;
rds excel options (sheet_interval='none' );

proc means mean median p90 p95 p99 maxdec=2 data=sample;
          var &oc;
          title overall prevalence of outcome;
run;      proc print noobs data=fit1&num;
title fit statistics;
run;
```

TAKE SOME RISKS



Try something new

IT’S MAINLY A GRIND

USE HISTOGRAMS

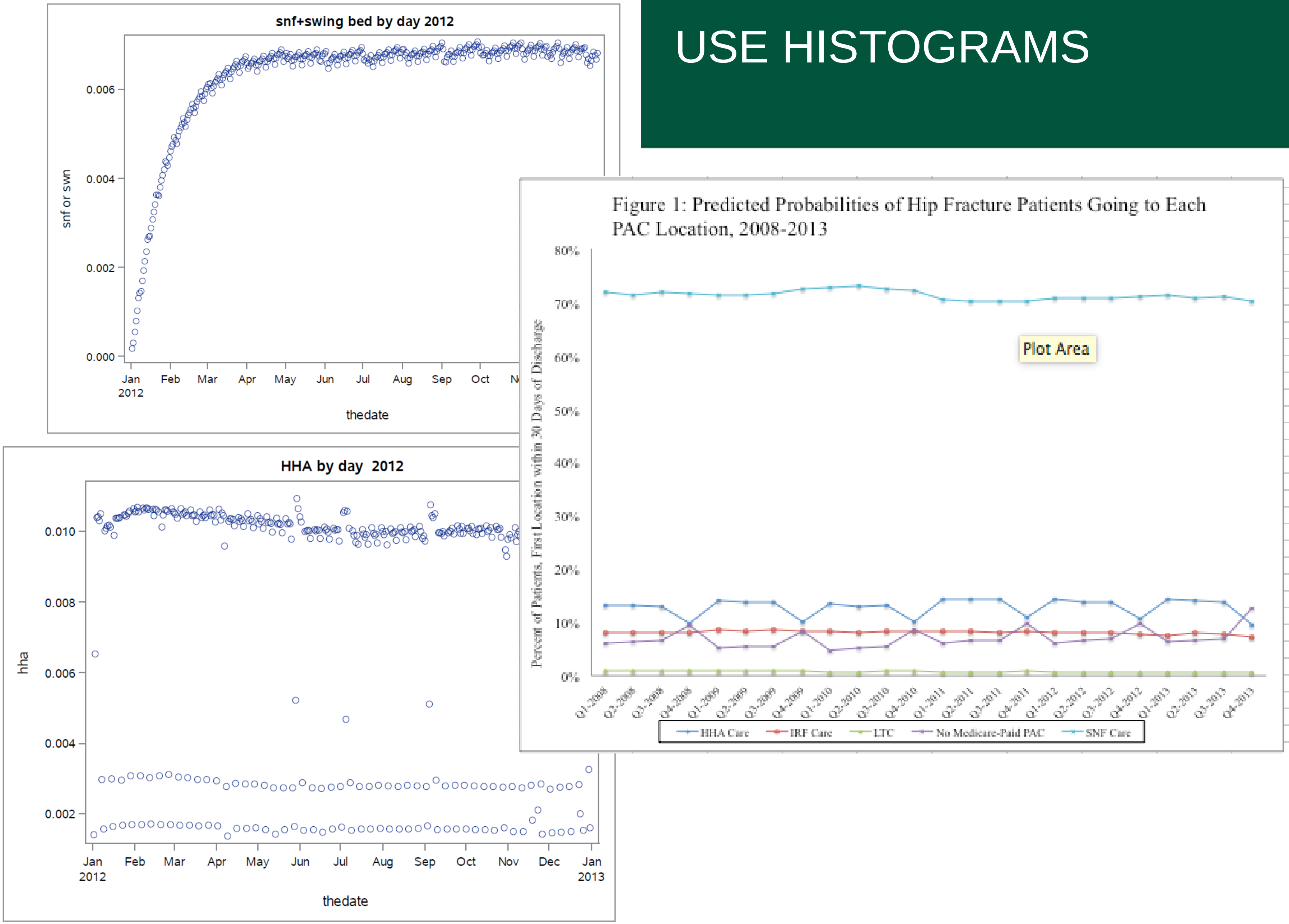


Figure 1: Predicted Probabilities of Hip Fracture Patients Going to Each PAC Location, 2008-2013