

Lecture 8

1. Multiple imputation
2. Making missing-value datasets: MCAR and non-MAR
3. Indicator variables
4. Proc MI and Proc MIanalyze
5. Spline smoothing
6. Jittering

1

MCAR and MAR

Missing completely at random (MCAR): data are missing independently of both observed and unobserved data.

Example: a participant flips a coin to decide whether to complete the depression survey.

Missing at random (MAR): given the observed data, data are missing independently of unobserved data.

Example: male participants are more likely to refuse to fill out the depression survey, but it does not depend on the level of their depression.

2

The first 10 observations from the Depression Study data:

Obs	baseline	final	change	drug	center
1	27	4	23	D	1
2	27	9	18	D	1
3	26	8	18	D	1
4	27	5	22	D	1
5	36	8	28	D	1
6	39	18	21	D	1
7	25	14	11	D	1
8	33	8	25	D	1
9	38	9	29	D	1
10	39	21	18	D	1
. . .					

How can we make it 10% MCAR for an illustrative example?

3

1. Randomly select 10% of the observations to change to missing.
2. For a selected observation, randomly choose baseline or final to be missing (optional).

For step 1, we need to choose a random number, and we need to know its distribution so we can divide the range into two parts:

one part has 10% probability, the other has 90%

Easy choice: random number from Uniform[0, 1] distribution, equal probability for any number in the interval [0, 1].

Could we use a standard Normal random number?

4

RANUNI(seed)

Arguments

seed

is an integer. If **seed** \leq 0, the time of day is used to initialize the seed stream.

Range:	seed < 2 ³¹ -1
See:	Seed Values for more information about seed values

Details

The RANUNI function returns a number that is generated from the uniform distribution on the interval (0,1) using a prime modulus multiplicative generator with modulus 2³¹- and multiplier 397204094 (Fishman and Moore 1982) (See [References](#)).

You can use a multiplier to change the length of the interval and an added constant to move the interval. For example,

```
random_variate=a*ranuni(seed)+b;
```

returns a number that is generated from the uniform distribution on the interval (b,a+b).

5

```
data HamD_MCAR;
  set ph6470.hamd2;
  if (ranuni(-1) < .10 ) then do; select a 10% random sample
    if (rannor(-1) > 0) then baseline=. ; coin flip
  else final =. ;
  change = .;
end;
proc print data=MCAR (obs=10);
  var baseline final change drug center;
```

Obs	baseline	final	change	drug	center
1	27	4	23	D	1
2	27	9	18	D	1
3	26	8	18	D	1
4	27	5	22	D	1
5	36	.	.	D	1
6	39	18	21	D	1
7	25	14	11	D	1
8	33	8	25	D	1
9	38	9	29	D	1
10	39	.	.	D	1

6

How could we make an example data set where missing does not happen at random?

Suppose participants with small changes were more likely to drop out = missing final value.

Identify “small” changes. Randomly select among these to delete.

The UNIVARIATE Procedure
Variable: `change`

Quantile	Estimate
100% Max	29.0
99%	28.5
95%	25.5
90%	23.5
75% Q3	19.0
50% Median	16.0
25% Q1	11.5
10%	9.0
5%	6.5
1%	1.0
0% Min	-2.0

7

```
data not_MAR;  
  set ph6470.hamd2;  
  if (change LE 11.5) then do; select smallest 25% of changes  
    if (ranuni(-1) < .5) then do; flip coin  
      final = . ;  
      change = . ;  
    end;  
  end;
```

Obs	baseline	final	change	drug	center
1	27	4	23	D	1
2	27	9	18	D	1
3	26	8	18	D	1
4	27	5	22	D	1
5	36	8	28	D	1
6	39	18	21	D	1
7	25	.	.	D	1 <i>change = 11</i>
8	33	8	25	D	1
9	38	9	29	D	1
10	39	21	18	D	1

8

Multiple Imputation

Suppose we have collected data in order to estimate a parameter θ , such as the population mean or a vector of regression coefficients.

1. **Proc MI** For each missing value Y_i , generate M estimates y_{im} , $m = 1, \dots, M$ using the distribution of observed values.

This is where we rely on the Missing At Random property.

How do we estimate this distribution?

Fill in missing values in the data using each set $\{y_{im}\}$, to produce M complete data sets.

2. Analyze each of the M complete data sets to get a parameter estimate $\hat{\theta}_m$ with variance W_m (squared standard error).

9

3. **Proc MIanalyze** Combine the results of the M analyses to draw conclusions. Combined estimate of θ is the average of the M estimates $\{\hat{\theta}_m\}$:

$$\bar{\theta}_M = \frac{1}{M} \sum_1^M \hat{\theta}_m.$$

Variance of this estimate comes from the *within-imputation* variance, estimated by the mean \bar{W}_M of the $\{W_m\}$,

and the *between-imputation* variance

$$B_M = \frac{1}{M-1} \sum_1^M (\hat{\theta}_m - \bar{\theta}_M)^2,$$

and so its standard error is:

$$SE(\bar{\theta}_M) = \sqrt{\bar{W}_M + \frac{M+1}{M} B_M}.$$

Little & Rubin (2002) *Statistical Analysis with Missing Data, Second Edition*

The SAS code will have 3 steps:

1. Proc MI generates M complete data sets, indexed by `_Imputation_`
2. Proc GLM fits the model, BY `_Imputation_`, and outputs the results as a dataset (in addition to writing them to the output window)
3. Proc MIanalyze reads the output dataset and produce the combined estimate

An additional problem is that drug and center are CLASS variables and MIanalyze has problems with these.

So we need to make indicators for both variables, which is what Proc GLM does.

11

```
data indicators;
  set ph6470.hamd2;
  drugD = (drug="D"); logical variables
  center1=(center=1);
  center2=(center=2);
  center3=(center=3);
  center4=(center=4);
  drugcenter_1 = drugD * center1; multiply to get interactions
  drugcenter_2 = drugD * center2;
  drugcenter_3 = drugD * center3;
  drugcenter_4 = drugD * center4;

proc print data=indicators;
  var drug drugD center center1-center4 drugcenter_1-drugcenter_4;
```

12

First, multiple imputation where it should work, in MCAR dataset:

```
Proc MI data=ph6470.hamd_mcar out=C output data set
  nimpute=25 number of filled-in datasets
  seed=74950631 seed for randomization
  minimum= 0 0 maximum= 40 40 reject values outside 0 - 40
  round=1.0; round to integer
  var baseline final; variables to fill in
Proc GLM data=C;
  model final = baseline drugD center1 center2 center3 center4
    / inverse solution;
  by _Imputation_;
  ODS output ParameterEstimates=glmparms InvXPX=glmxpxi; output data
Proc MIanalyze parms=glmparms xpxi=glmxpxi;
  modeleffects Intercept baseline drugD center1 center2
    center3 center4;
```

15

Default: assume missing data is Normally distributed

The MI Procedure

Model Information

Data Set	PH6470.HAMD_MCAR
Method	MCMC
Multiple Imputation Chain	Single Chain
Initial Estimates for MCMC	EM Posterior Mode
Start	Starting Value
Prior	Jeffreys
Number of Imputations	25
Number of Burn-in Iterations	200
Number of Iterations	100
Seed for random number generator	74950631

Missing Data Patterns

Group	baseline	final	Freq	Percent	-----Group Means-----	
					baseline	final
1	X	X	85	85.00	29.882353	14.352941
2	X	.	5	5.00	31.600000	.
3	.	X	10	10.00	.	13.800000

16

From MCAR data set:

The MIANALYZE Procedure

Model Information

PARMS Data Set WORK.GLMPARMS
XPXI Data Set WORK.GLMXPXI
Number of Imputations 25

Multiple Imputation Parameter Estimates

Parameter	Estimate	Std Error	95% Confidence Limits		DF
Intercept	-4.728238	2.717329	-10.0611	0.60462	923.56
baseline	0.718213	0.075861	0.5693	0.86709	948.85
drugD	-5.441965	1.058178	-7.5172	-3.36669	1948.1
center1	0.494309	1.906681	-3.2495	4.23811	668.4
center2	1.167676	2.049106	-2.8514	5.18671	1702.7
center3	0.111310	1.805349	-3.4317	3.65429	940.91
center4	0.366387	2.056796	-3.6750	4.40773	484.5

17

From not_MAR, data not missing at random:

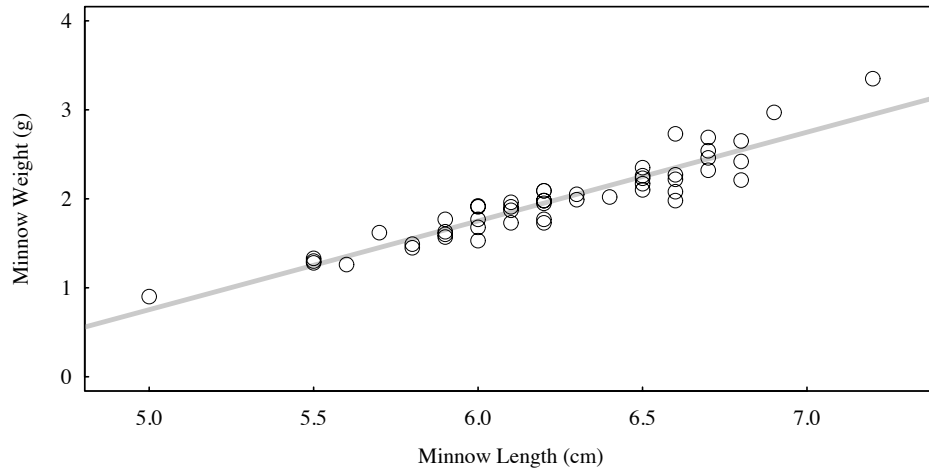
Multiple Imputation Parameter Estimates

Parameter	Estimate	Std Error	95% Confidence Limits		DF
Intercept	-4.501912	2.306115	-9.02448	0.02066	2052.8
baseline	0.681527	0.067575	0.54886	0.81419	725.02
drugD	-4.527452	0.980670	-6.45381	-2.60110	544.97
center1	-0.128483	1.547977	-3.16308	2.90611	5967.2
center2	1.167716	1.751370	-2.26603	4.60146	3694.7
center3	-0.436672	1.470987	-3.31993	2.44659	19341
center4	-1.577074	1.659070	-4.82999	1.67584	3293.2

Recall that small changes were more likely to be missing.

18

Scatterplot Smoothing



To investigate whether weight could be used to predict length in adult minnows, the researcher recorded the length (cm) and weight (g) of 50 adult minnows. We know that weight is closely related to length in minnows. We have several weights at most of the measured lengths, so it makes sense to think about the *average weight at each length*.

19

The general idea here is a **mean function**: the mean weight at each length

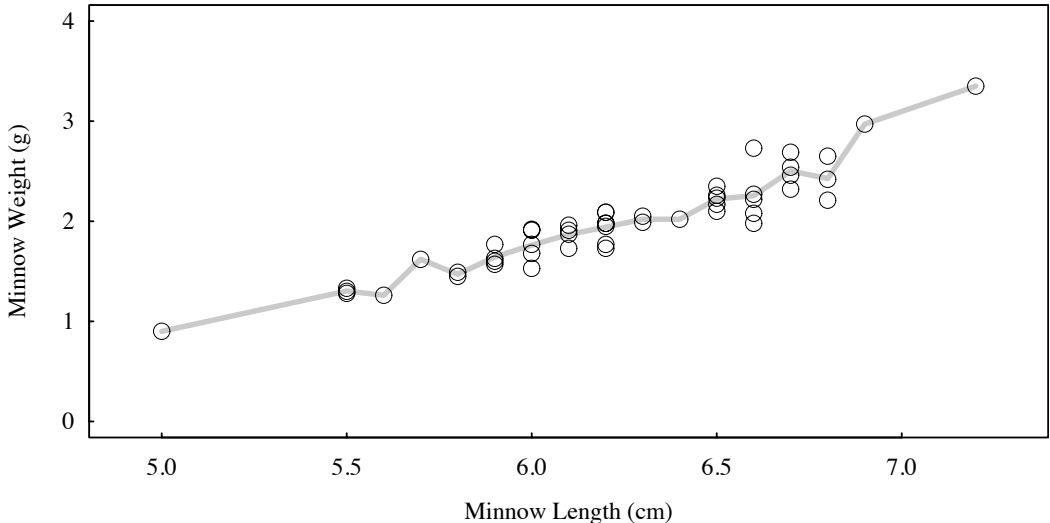
$$\text{minnow weight} = \mu(\text{minnow length}) + \text{error},$$

Fitted regression line above estimates the mean function *under the assumption that it is a line*.

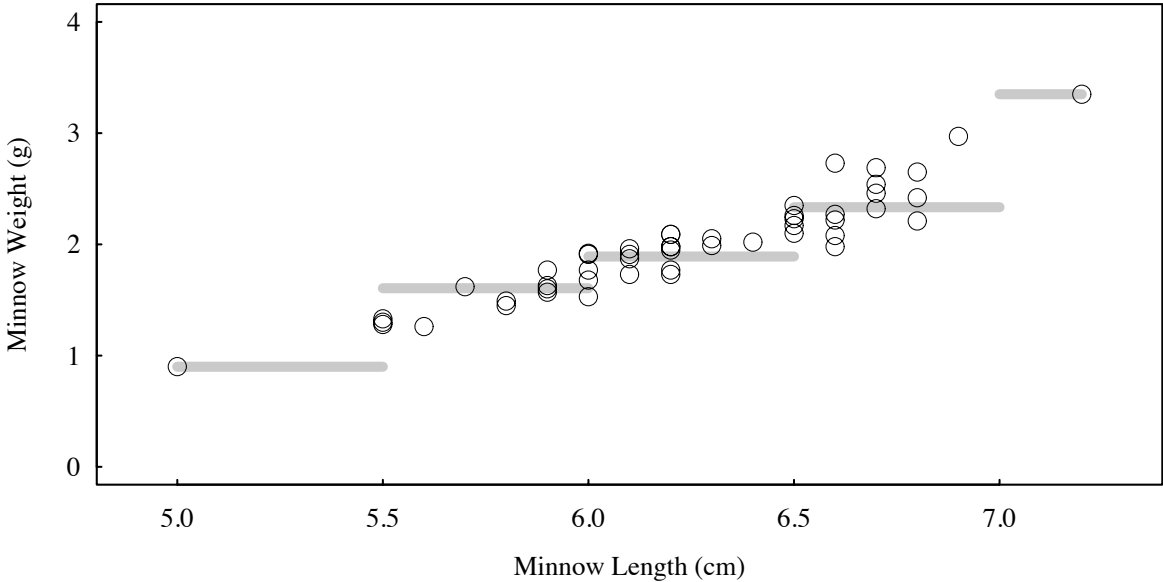
We'll look at other ways to estimate mean functions.

20

1. Find the mean at each x -value (length). Connect these means with line segments.



2. Define categories for x (length), such as 5–5.5, 5.5–6, 6–6.5, 6.5–7 cm. Draw a horizontal line at the mean in each category.



3. **Spline smoothing.** This estimates the mean function by joining together **splines** (a segment of cubic polynomial) so that they have a continuous second derivative at the join points, called **knots**.

In SAS, the amount of smoothness is set by the parameter N in the `interpol` option: `interpol = smNs`:

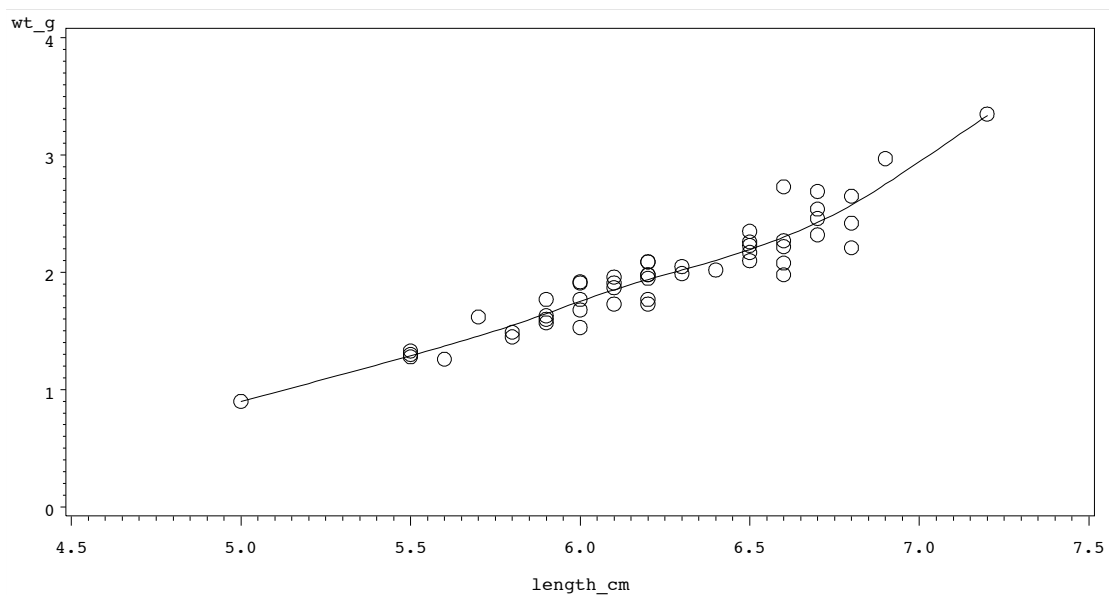
$N = 00$ (roughest) makes the spline go through mean at every x

$N = 99$ (smoothest) gives linear regression.

```
Goptions reset=all vsize=4in ftext=simplex device=pdf lfactor=2
noborder gsfname=graphout gsfmode= replace;
filename graphout "C: ... minnows.pdf";
symbol1 value=circle h=1.5;
```

```
Proc Gplot data=pubh.minnows ;
plot wt_g * length_cm / haxis = 4.5 to 7.5 by 0.5;
symbol1 interpol = sm60s width=2;
```

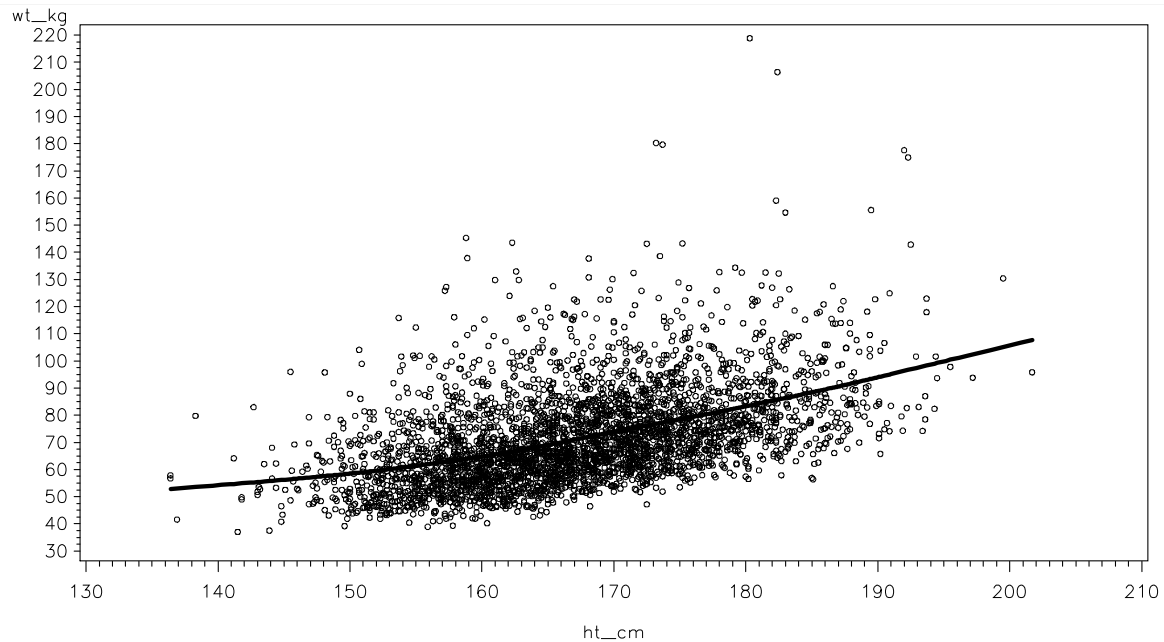
23



Weight (g) and length (cm) of 50 adult minnows with cubic spline smooth.

24

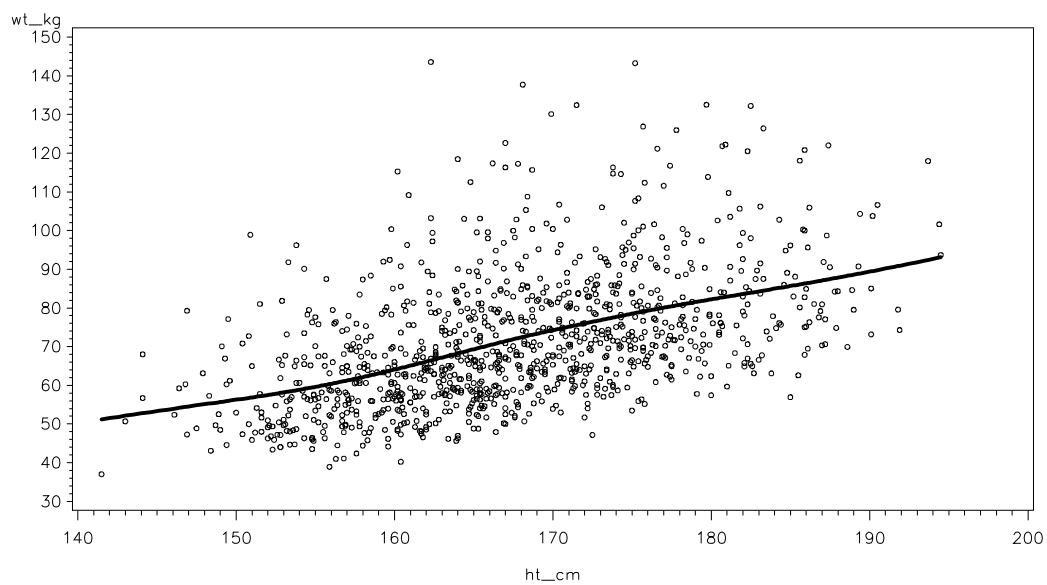
Spline smooth of the NHANES20 data:



If too many points on the graph reduces clarity, we could take a random sample of the data to plot.

25

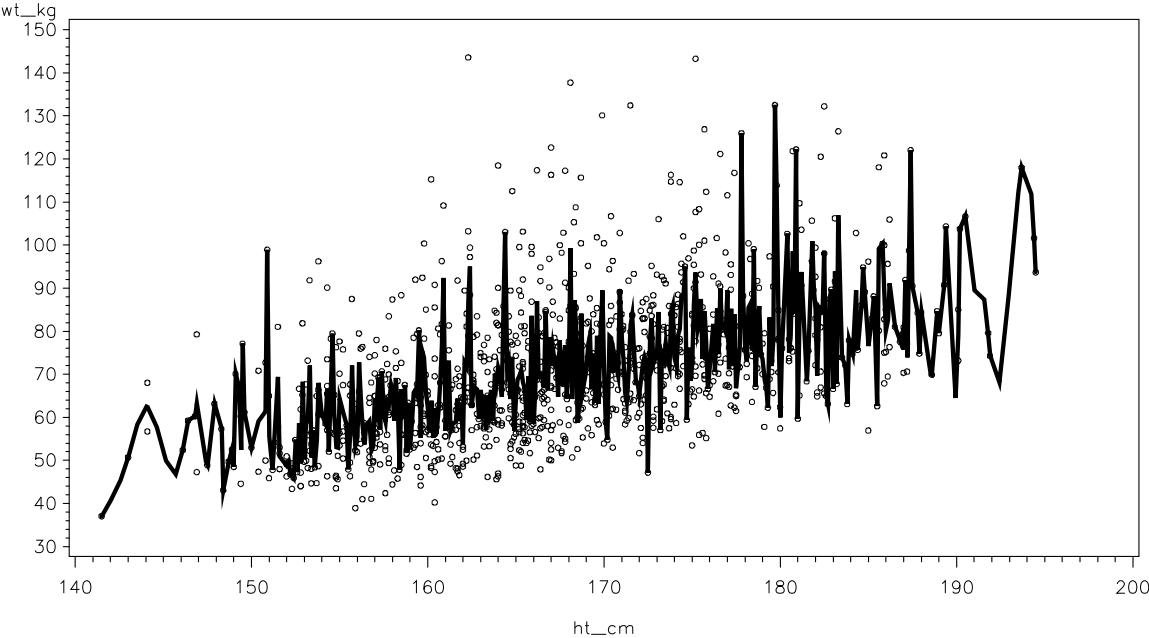
Spline smooth (sm60s) of 30% random sample from NHANES20:



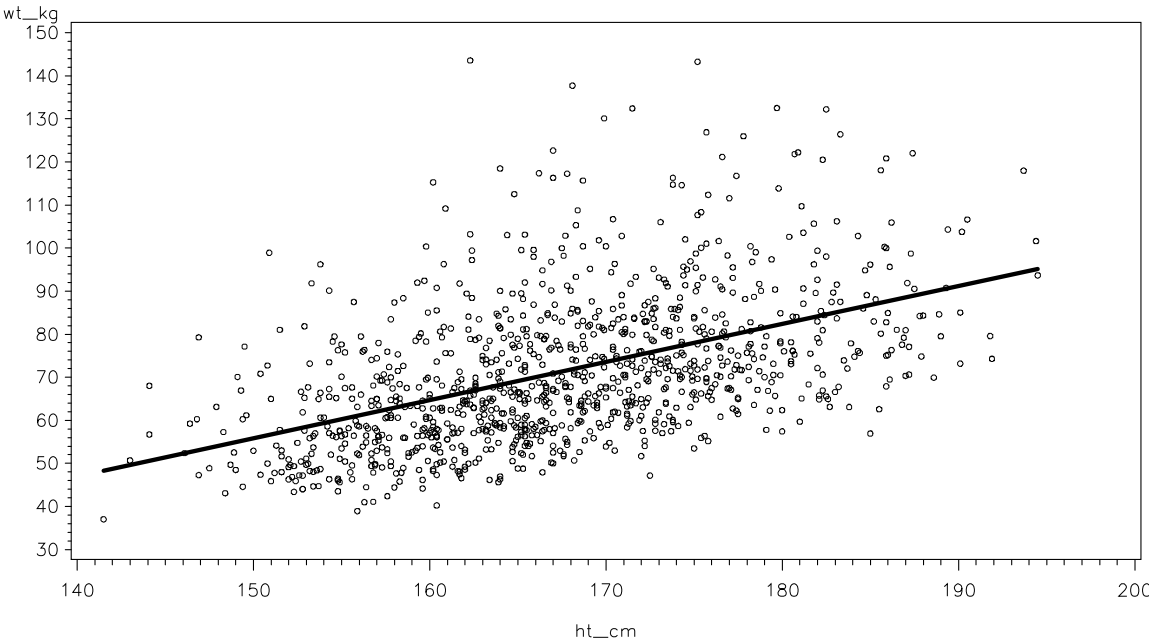
Spline fit to sample data, *not original data*. Main flaw of SAS graphics: every element is calculated within Gplot using one data set. Cannot calculate parts separately and add to graph.

26

Spline smooth (sm00 s) of 30% random sample from NHANES-20:



Spline smooth (sm99 s) of 30% random sample from NHANES20:

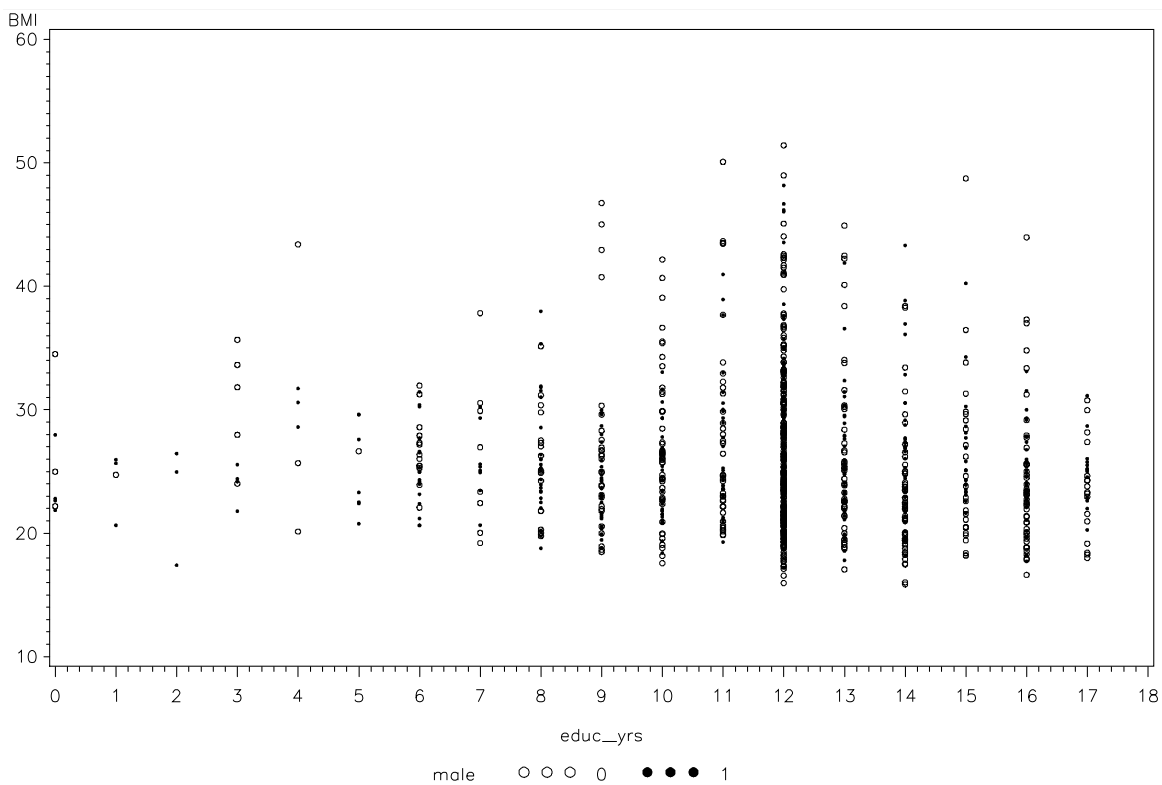


NHANES III example: Is education associated with obesity? Is it the same for men and women?

```
Goptions reset=all vsize=5in ftext=simplex device=pdf lfactor=2
noborder gsfname=graphout gsfmode= replace;
filename graphout "C: ... NHANES.pdf";
symbol1 value=circle h=0.5;
symbol2 value=dot h=0.35;
```

```
Proc Gplot data=pubh.NHANES20 ;
plot BMI * educ_years = male / haxis = 0 to 18 by 1;
```

29



30

Jittering

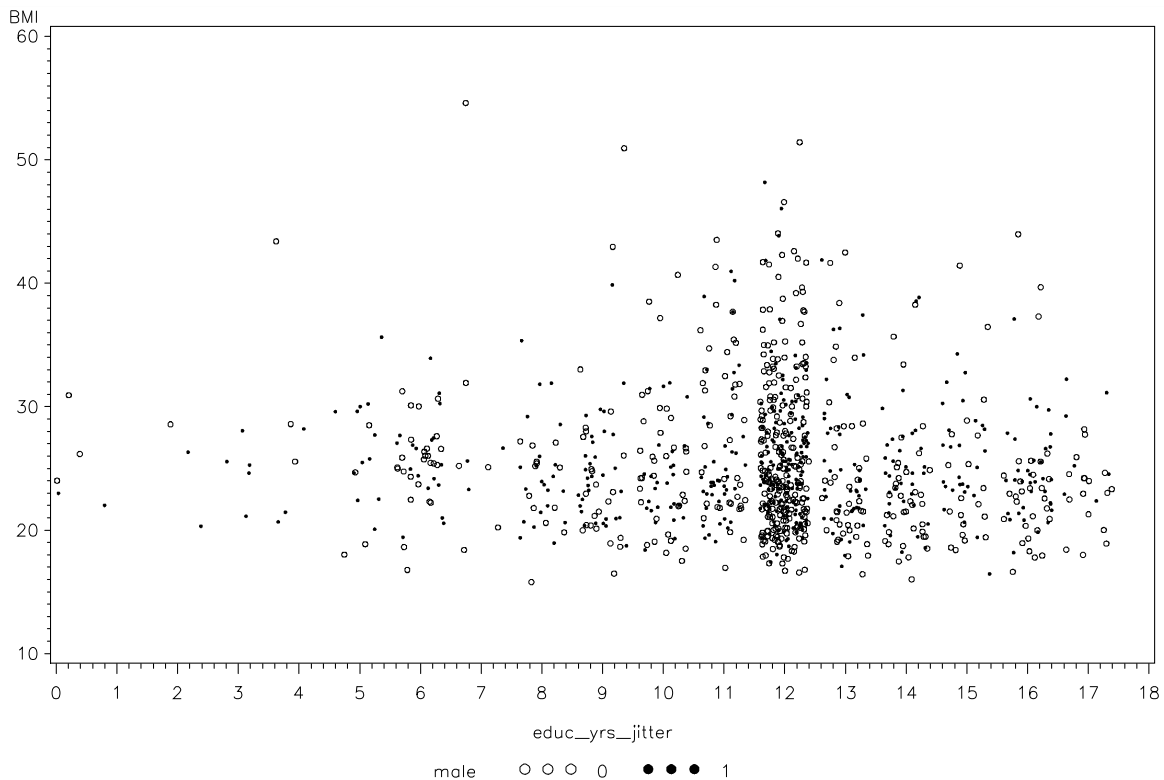
One solution to the vertical strips is to jitter years of education by adding horizontal random noise. This must be done in the data step:

```
data sample;
  set bmi;
  if (educ_yrs LE 18) then do;
    educ_yrs_jitter = educ_yrs + 0.8*ranuni(684112) - .4;
    if (ranuni(339087) LE 0.3) then output; select 30% random sample
  end;
```

If U is Uniform[0,1], then $a * U + b$ is Uniform[$b, a + b$]

$(.8 * U - .4)$ is Uniform[-.4, +.4] so we don't get overlap between strips.

31



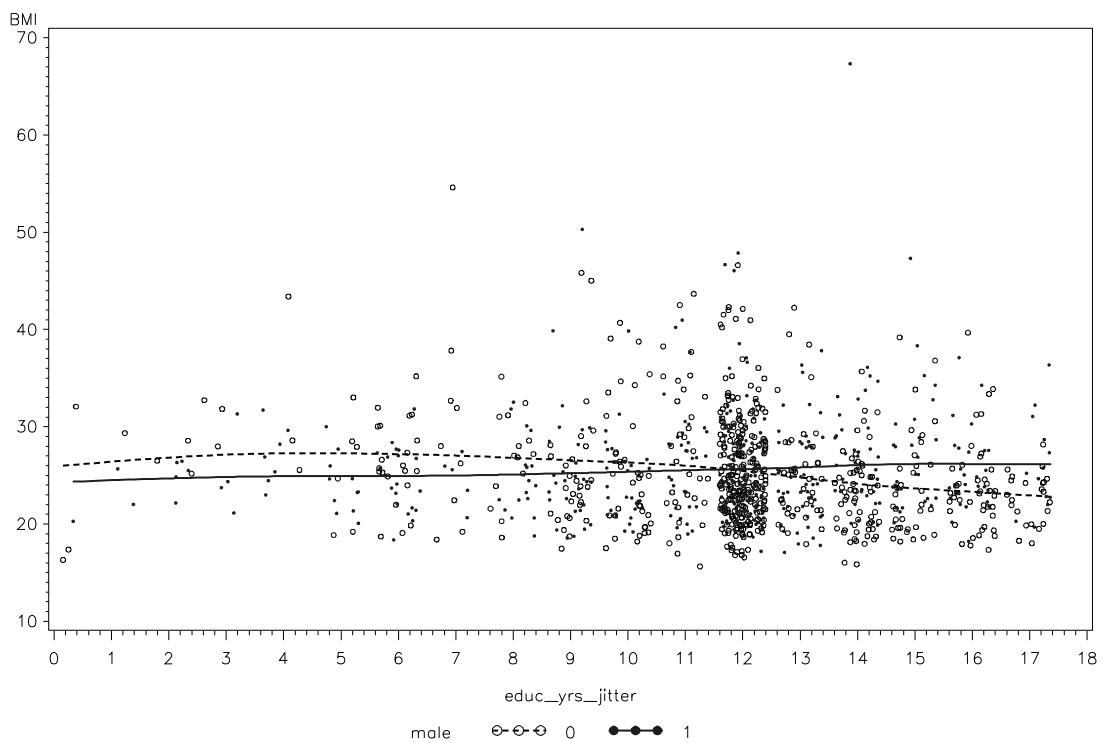
32

Smooth is particularly helpful when it is hard to visualize a mean function.

Plot the data with different symbols for men and women, and add two spline smooths for each gender.

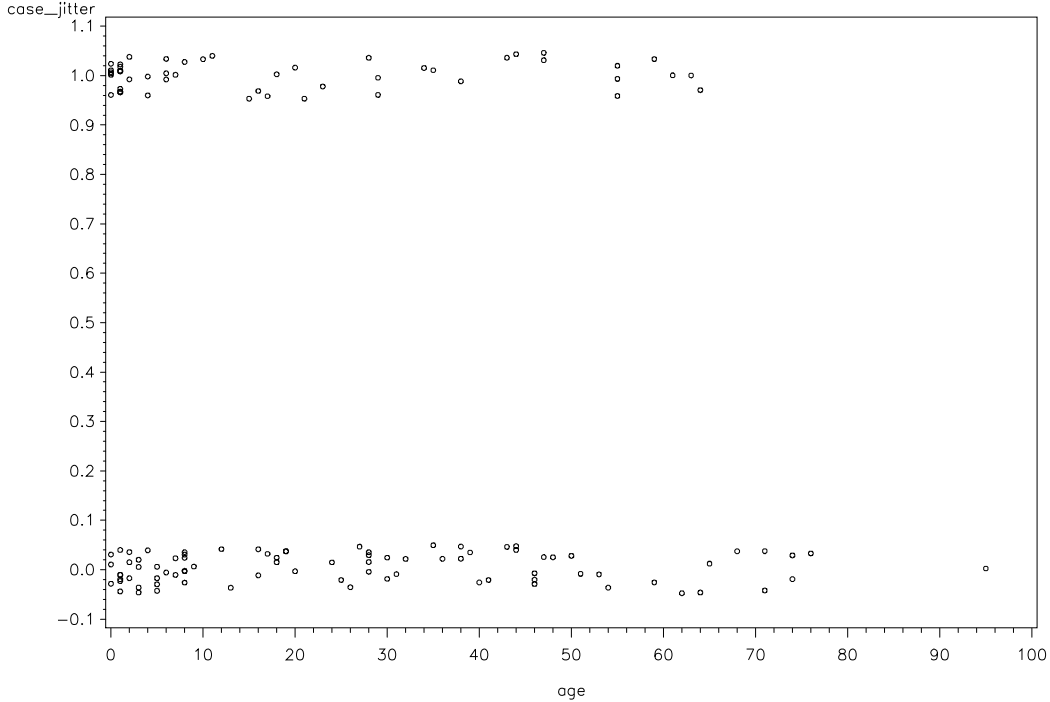
```
goptions reset=all vsize=5in ftext=simplex ctext=black lfactor=1.5;  
symbol1 value=circle c=black h=.5;  
symbol2 value=dot c=blue h=0.35;  
  
proc gplot data=sample;  
  plot BMI * educ_yrs = male / haxis=0 to 18 by 1;  
  symbol1 interpol=sm60s c=black line=2 width=4;  
  symbol2 interpol=sm60s c=blue line=1 width=4;
```

33

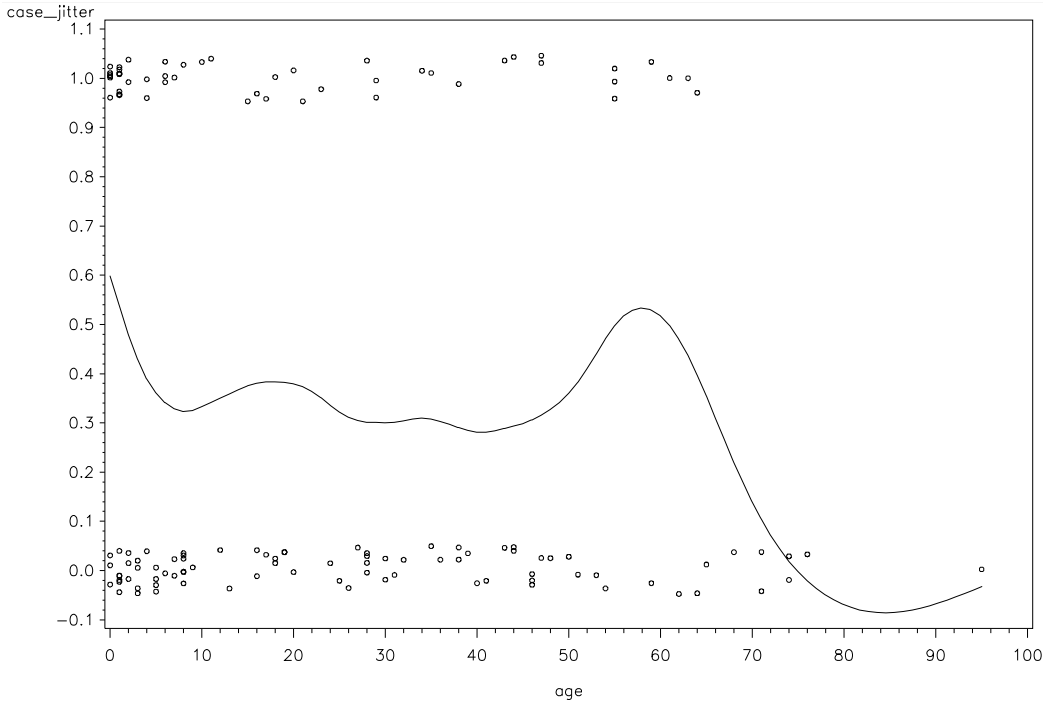


34

Binary (0/1) response is another situation that makes it hard to visualize the mean function. These are vertically jittered 0s (non-cases) and 1s (cases) vs subject age.



35



36