

432 Class 09

<https://thomaseLove.github.io/432-2023/>

2023-02-14

Today's Agenda

- A New NHANES Example
- Logistic Regression Analyses in Project A
 - Establishing a Research Question
 - Identifying / Tidying Outcome and Candidate Predictors
 - Dealing with Missing Data
 - Building a “Main Effects” Model and Plotting Effects
 - Considering Non-Linear Terms
 - Fitting an “Augmented” Model and Plotting Effects
 - Summarizing/Presenting a Final Model
 - In-Sample and Validated Model Summaries
 - Selecting Model Y or Model Z
 - Describing a Meaningful Effect (see Notes, Chapter 22)
 - ROC curve for the Final Model
 - Nomogram for the Final Model

Today's R Setup

```
knitr::opts_chunk$set(comment = NA)
```

```
library(nhanesA)
```

```
library(broom)
```

```
library(caret)
```

```
library(janitor)
```

```
library(knitr)
```

```
library(mosaic)
```

```
library(naniar)
```

```
library(pROC)
```

```
library(rms)
```

```
library(simputation)
```

```
library(tidyverse)
```

```
theme_set(theme_bw())
```

Section 1

The Data

NHANES Data

We'll use data from the 2011-2012 administration of NHANES here, just because I don't want to overlap with any studies people might be thinking about for Project B.

Data bases we'll use are:

- **DEMO_G** for Demographic Variables
- **CDQ_G** for Cardiovascular Health Questionnaire
- **HSQ_G** for Current Health Status Questionnaire
- **BPX_G** for Blood Pressure Examination Results
- **BMX_G** for Body Measures Examination Results
- **MCQ_G** for Medical Conditions Questionnaire

Pulling the Data, I

```
demo_g <- nhanes("DEMO_G") |> tibble(); dim(demo_g)
```

```
[1] 9756    48
```

```
cdq_g <- nhanes("CDQ_G") |> tibble(); dim(cdq_g)
```

```
[1] 3603    17
```

```
hsq_g <- nhanes("HSQ_G") |> tibble(); dim(hsq_g)
```

```
[1] 8956    14
```

Pulling the Data, II

```
bpx_g <- nhanes("BPX_G") |> tibble(); dim(bpx_g)
```

```
[1] 9338    27
```

```
bmh_g <- nhanes("BMH_G") |> tibble(); dim(bmh_g)
```

```
[1] 9338    26
```

```
mcq_g <- nhanes("MCQ_G") |> tibble(); dim(mcq_g)
```

```
[1] 9364    92
```

Merging the Tibbles

```
df_mlist <- list(demo_g, cdq_g, hsq_g, bpx_g, bmx_g, mcq_g)

nh_merge <- df_mlist |>
  reduce(left_join, by = "SEQN") # reduce is from purrr

dim(nh_merge)
```

```
[1] 9756 219
```

We had 262 variables in our original seven tibbles, but that counts the SEQN variable seven times, and we only have it once in our `nh_merge` tibble.

Now, which of these 256 variables are we actually going to use?

Variables We'll Use Today

NHANES	Description	Source
SEQN	Identifying code	All 6 files
CDQ010	Shortness of breath on stairs/inclines?	CDQ_G
RIDAGEYR	Age in years at screening	DEMO_G
HSD010	General health condition (E/VG/G/F/P)	HSQ_G
LBXSCLSI	Serum Chloride (mmol/L)	BIOPRO_G
BPXDI1	Diastolic BP (first reading, in mm Hg)	BPX_G
BMXBMI	Body Mass Index (kg/m^2)	BMX_G
MCQ010	Ever been told you have asthma	MCQ_G

Selecting Today's Variables

```
nh_today <- nh_merge |>
  select(SEQN, CDQ010, age = RIDAGEYR, sroh = HSD010,
         dbp = BPXDI1, bmi = BMXBMI,
         asthma = MCQ010) |>
  filter(CDQ010 < 3) |>
  filter(age < 80)

dim(nh_today)
```

```
[1] 3234    7
```

Check the variables, 1

- 1 SEQN should be unique for each row in the data.

```
identical(nrow(nh_today), n_distinct(nh_today$SEQN))
```

```
[1] TRUE
```

Check the variables, 2

② CDQ010 was 1 for Yes, 2 for No. We want 1 for Yes, 0 for No.

```
nh_today <- nh_today |> mutate(sbreath = 2 - CDQ010)

nh_today |> count(CDQ010, sbreath) # sanity check
```

```
# A tibble: 2 x 3
  CDQ010 sbreath      n
  <dbl>   <dbl> <int>
1     1     1    1015
2     2     0    2219
```

```
nh_today <- nh_today |> select(-CDQ010) |>
  relocate(sbreath, .after = "SEQN")
```

Check the variables, 3

- 3 Age should be between 40 and 79 years
- 4 Body mass index should be between 12.4 and 82.1 kg/m²
- 5 Diastolic BP should be between 30 and 120 mm Hg (treat values below 30 as NA)

```
df_stats(~ age + bmi + dbp, data = nh_today) |>  
  rename(var = response) |> kable(digits = 1)
```

var	min	Q1	median	Q3	max	mean	sd	n	missing
age	40.0	48.0	57.0	65.0	79.0	57.3	10.7	3234	0
bmi	13.6	24.7	28.3	32.9	82.1	29.5	6.9	3062	172
dbp	0.0	66.0	74.0	80.0	120.0	73.1	12.4	2854	380

Replace dbp values below 30 with NA

```
nh_today |> count(dbp < 30)
```

```
# A tibble: 3 x 2
  `dbp < 30`      n
  <lgl>         <int>
1 FALSE         2842
2 TRUE           12
3 NA            380
```

```
nh_today <- replace_with_na_at(nh_today, "dbp", ~ .x < 30)

favstats(~ dbp, data = nh_today) |> kable(digits = 1)
```

min	Q1	median	Q3	max	mean	sd	n	missing
30	66	74	80	120	73.4	11.6	2842	392

Check the variables, 4

- ⑥ `asthma` should be a two-level factor (currently 1 = Yes, 2 = No, 9 = Don't Know, which we'll treat as missing)

```
nh_today |> count(asthma)
```

```
# A tibble: 3 x 2
  asthma      n
  <dbl> <int>
1      1   452
2      2  2779
3      9     3
```

Recoding Asthma as a factor with 3 missing values

```
nh_today <- nh_today |>
  mutate(asthma = fct_recode(
    factor(asthma), "Yes" = "1", "No" = "2", NULL = "9"),
    asthma = fct_relevel(asthma, "No"))

nh_today |> tabyl(asthma) |> adorn_pct_formatting()
```

asthma	n	percent	valid_percent
No	2779	85.9%	86.0%
Yes	452	14.0%	14.0%
<NA>	3	0.1%	-

Check the variables, 5

- Self-reported overall health should be a five-level factor

```
nh_today <- nh_today |>
  mutate(sroh = fct_recode(factor(sroh), "E" = "1", "VG" = "2",
                                "F" = "4", "P" = "5"))

nh_today |> tabyl(sroh) |> adorn_pct_formatting() |> kable()
```

sroh	n	percent	valid_percent
E	238	7.4%	8.5%
VG	684	21.2%	24.5%
G	1119	34.6%	40.1%
F	612	18.9%	21.9%
P	136	4.2%	4.9%
NA	445	13.8%	-

Which Variables are Missing?

```
miss_var_summary(nh_today)
```

```
# A tibble: 7 x 3
  variable n_miss pct_miss
  <chr>      <int>    <dbl>
1 sroh      445    13.8
2 dbp       392    12.1
3 bmi       172     5.32
4 asthma     3    0.0928
5 SEQN       0     0
6 sbreath    0     0
7 age        0     0
```

How Many Missing Values?

```
miss_case_table(nh_today)
```

```
# A tibble: 4 x 3
```

	n_miss_in_case	n_cases	pct_cases
	<int>	<int>	<dbl>
1	0	2555	79.0
2	1	475	14.7
3	2	75	2.32
4	3	129	3.99

Updated Codebook

We have 3234 rows and 7 in the `nh_today` data now. 2555, or 79% of the rows have complete data on these 7 variables.

Name	Description	NHANES Source
SEQN	Identifying code	All 6 files
sbreath	Shortness of breath on stairs/inclines?	CDQ_G (CDQ010)
age	Age in years at screening	DEMO_G (RIDAGEYR)
sroh	Self-reported health (E/VG/G/F/P)	HSQ_G (HSD010)
dbp	Diastolic BP (1st reading, in mm Hg)	BPX_G (BPXDI1)
bmi	Body Mass Index (kg/m^2)	BMX_G (BMXBMI)
asthma	Ever been told you have asthma?	MCQ_G (MCQ010)

- Inclusions/Exclusions: Valid (1 or 0) response to `sbreath`, age between 40 and 79 years, inclusive.

Section 2

Project A Tasks

Establishing a Research Question

How effectively can we predict whether or not an adult subject has experienced “shortness of breath when hurrying on the level or walking up a slight hill” on the basis of their age, self-reported overall health, diastolic blood pressure, body mass index and whether or not they have been told they have asthma?

- Our data come from NHANES 2011-12, and describe a total of 3234 (unweighted) adult (ages 40-79) subjects.
- We will not use survey weights in this work.

Identifying / Tidying Outcome

Our outcome is the subject's response to the following question:

Have you had shortness of breath either when hurrying on the level or walking up a slight hill?

This was asked of adults ages 40 years and up, as question CDQ010 on the CDQ_G questionnaire in NHANEZS 2011-12, and we've included subjects who gave either a Yes or No response.

```
nh_today |> tabyl(sbreath) |> adorn_totals() |>
  adorn_pct_formatting()
```

sbreath	n	percent
0	2219	68.6%
1	1015	31.4%
Total	3234	100.0%

Identifying Candidate Predictors

The five predictors we will examine for this outcome are age, sroh, dbp, bmi and asthma.

Name	Description	Missing?
SEQN	Identifying code	None
sbreath	Shortness of breath on stairs/inclines?	None
age	Age in years at screening	None
sroh	Self-reported health (E/VG/G/F/P)	445
dbp	Diastolic BP (1st reading, in mm Hg)	392
bmi	Body Mass Index (kg/m^2)	172
asthma	Ever been told you have asthma?	3

Dealing with Missing Data

We have excluded all cases with missing `sbreath` so our outcome is complete.

We will assume MAR for the remaining missing values and then use single imputation both to:

- build a Spearman ρ^2 plot
- fit our models Y and Z

If you wanted to use multiple imputation in the project, I would do that at the end, by refitting the “winning” model and summarizing those results (after imputation) only as part of your **Final Model** materials. I might also use `aregImpute()` to get my single imputation, although I won't here.

Single Imputation via simulation

```
set.seed(43212345)
nh_today_i <- nh_today |> data.frame() |>
  impute_rhd(asthma ~ age) |>
  impute_rlm(dbp ~ age + asthma) |>
  impute_rlm(bmi ~ dbp + age + asthma) |>
  impute_cart(sroh ~ age + bmi) |>
  as_tibble()

n_miss(nh_today_i) # should now have no missing data
```

```
[1] 0
```

Resulting nh_today_i tibble

```
nh_today_i
```

```
# A tibble: 3,234 x 7
```

	SEQN	sbreath	age	sroh	dbp	bmi	asthma
	<dbl>	<dbl>	<dbl>	<fct>	<dbl>	<dbl>	<fct>
1	62164	0	44	G	56	23.2	No
2	62172	0	43	G	70	33.3	No
3	62177	0	51	G	68	20.1	No
4	62179	0	55	VG	78	27.6	No
5	62182	1	75	G	69.6	28.5	No
6	62191	0	70	G	68	28.5	No
7	62199	1	57	VG	70	28	No
8	62200	0	42	VG	88	27.6	No
9	62201	0	58	G	73.3	28.6	No
10	62209	1	62	F	60	26	No

```
# ... with 3,224 more rows
```

Section 3

Model Y: The “Main Effects”

Building a “Main Effects” Model and Plotting Effects

First, we'll assume MAR and do our analysis on the (singly) imputed data `nh_today_i`)

```
d <- datadist(nh_today_i)
options(datadist = "d")

modY_si <- lrm(sbreath ~ age + sroh + dbp + bmi + asthma,
               data = nh_today_i, x = TRUE, y = TRUE)
```

modY_si results (from lrm fit)

```
> modY_si
Logistic Regression Model

lrm(formula = sbreath ~ age + sroh + dbp + bmi + asthma, data = nh_today_i,
     x = TRUE, y = TRUE)
```

		Model Likelihood	Discrimination	Rank Discrim.
		Ratio Test	Indexes	Indexes
Obs	3234	LR chi2 426.39	R2 0.174	C 0.718
0	2219	d.f. 8	R2(8,3234)0.121	Dxy 0.436
1	1015	Pr(> chi2) <0.0001	R2(8,2089.3)0.181	gamma 0.436
max deriv	2e-10		Brier 0.187	tau-a 0.188

	Coef	S.E.	Wald Z	Pr(> Z)
Intercept	-4.1132	0.4878	-8.43	<0.0001
age	0.0149	0.0039	3.79	0.0002
sroh=VG	0.4849	0.2337	2.07	0.0380
sroh=G	1.1641	0.2179	5.34	<0.0001
sroh=F	1.8890	0.2269	8.33	<0.0001
sroh=P	1.9771	0.2780	7.11	<0.0001
dbp	-0.0062	0.0038	-1.63	0.1033
bmi	0.0544	0.0062	8.79	<0.0001
asthma=Yes	0.7981	0.1111	7.18	<0.0001

Key Fit Summary Statistics for Model Y (modY_si)

```
temp <- modY_si$stats  
temp["C"]
```

C

0.7181358

```
temp["R2"]
```

R2

0.1735251

The Nagelkerke R^2 for this model is 0.718 and the C statistic is 0.174.

glm version of this same fit

```
modY_si_g <- glm(sbreath ~ age + sroh + dbp + bmi + asthma,  
  data = nh_today_i,  
  family = binomial(link = logit))
```

```
> modY_si_g  
  
Call:  glm(formula = sbreath ~ age + sroh + dbp + bmi + asthma, family = binomial(link = logit),  
  data = nh_today_i)  
  
Coefficients:  
(Intercept)      age      srohVG      srohG      srohF      srohP      dbp  
-4.113185    0.014921    0.484919    1.164107    1.888952    1.977082   -0.006171  
      bmi      asthmaYes  
 0.054416    0.798134  
  
Degrees of Freedom: 3233 Total (i.e. Null); 3225 Residual  
Null Deviance: 4024  
Residual Deviance: 3598      AIC: 3616
```

```
glance(modY_si_g) |> select(AIC, BIC) |> kable(dig = 1)
```

AIC	BIC
3615.7	3670.4

Tidied Table of Model Y (Exponentiated) Coefficients

Here's the code: result is on next slide.

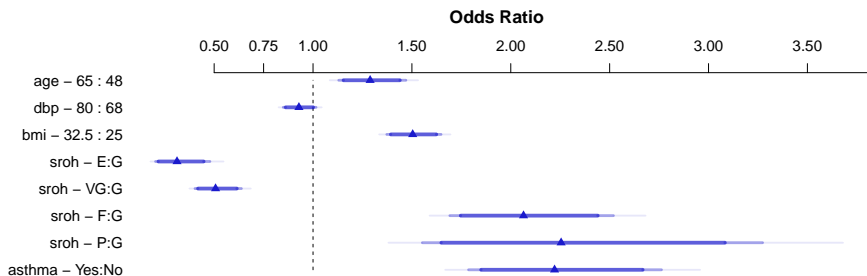
```
tidy(modY_si_g, exponentiate = TRUE,  
      conf.int = TRUE, conf.level = 0.90) |>  
  select(term, estimate, std.error,  
          low90 = conf.low, high90 = conf.high,  
          p = p.value) |> kable(digits = 3)
```

Tidied Table of Model Y (Exponentiated) Coefficients

term	estimate	std.error	low90	high90	p
(Intercept)	0.016	0.488	0.007	0.036	0.000
age	1.015	0.004	1.008	1.022	0.000
srohVG	1.624	0.234	1.117	2.415	0.038
srohG	3.203	0.218	2.267	4.652	0.000
srohF	6.612	0.227	4.607	9.737	0.000
srohP	7.222	0.278	4.609	11.519	0.000
dbp	0.994	0.004	0.988	1.000	0.103
bmi	1.056	0.006	1.045	1.067	0.000
asthmaYes	2.221	0.111	1.850	2.667	0.000

Model Y Effects Plot on Odds Ratio Scale

```
plot(summary(modY_si))
```



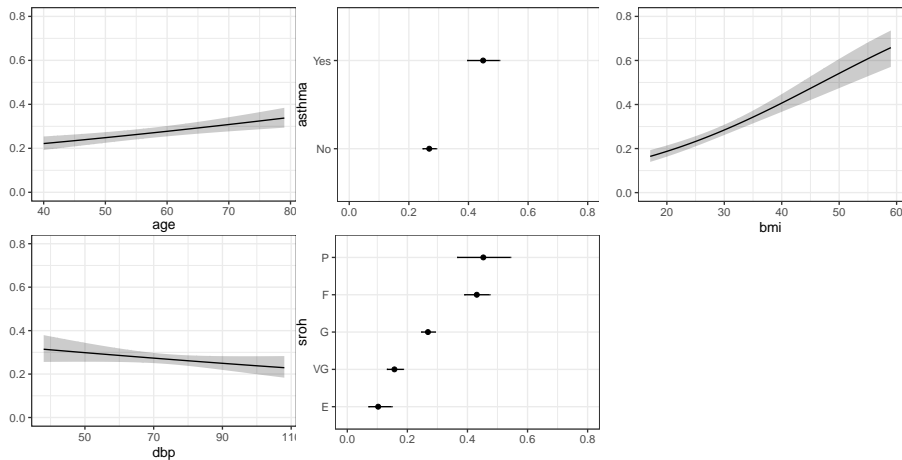
Details of Effects Plot (Model Y)

```
> summary(modY_si)
```

Effects				Response : sbreath			
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95
age	48	65.0	17.0	0.253660	0.066962	0.12242	0.384900
Odds Ratio	48	65.0	17.0	1.288700	NA	1.13020	1.469500
dbp	68	80.0	12.0	-0.074048	0.045455	-0.16314	0.015043
Odds Ratio	68	80.0	12.0	0.928630	NA	0.84947	1.015200
bmi	25	32.5	7.5	0.408120	0.046411	0.31716	0.499080
Odds Ratio	25	32.5	7.5	1.504000	NA	1.37320	1.647200
sroh - E:G	3	1.0	NA	-1.164100	0.217900	-1.59120	-0.737040
Odds Ratio	3	1.0	NA	0.312200	NA	0.20369	0.478530
sroh - VG:G	3	2.0	NA	-0.679190	0.116900	-0.90831	-0.450060
Odds Ratio	3	2.0	NA	0.507030	NA	0.40320	0.637590
sroh - F:G	3	4.0	NA	0.724850	0.101520	0.52588	0.923810
Odds Ratio	3	4.0	NA	2.064400	NA	1.69190	2.518900
sroh - P:G	3	5.0	NA	0.812980	0.190200	0.44019	1.185800
Odds Ratio	3	5.0	NA	2.254600	NA	1.55300	3.273200
asthma - Yes:No	1	2.0	NA	0.798130	0.111120	0.58035	1.015900
Odds Ratio	1	2.0	NA	2.221400	NA	1.78670	2.761900

Prediction Plot for Model Y

```
ggplot(Predict(modY_si, fun = plogis))
```



Confusion Matrix for Model Y

How well does our Model Y classify subjects using a decision rule at 0.5?

```
modY_aug <- augment(modY_si_g, type.predict = "response")

modY_aug <- modY_aug |>
  mutate(pred = ifelse(.fitted >= 0.5,
                        "Predict SB", "Predict No SB"))

modY_aug |> tabyl(pred, sbreath) |>
  adorn_totals(where = c("row", "col")) |> adorn_title()
```

	sbreath		
	0	1	Total
Predict No SB	2040	726	2766
Predict SB	179	289	468
Total	2219	1015	3234

- What fraction of our predictions are correct with this decision rule?

Summaries of Classification Accuracy, 1

		sbreath		
pred		0	1	Total
Predict No SB		2040	726	2766
Predict SB		179	289	468
Total		2219	1015	3234

- **Accuracy** is $(2040 + 289) / 3234 = 0.720$
 - 72.0% of this model's predictions were accurate.
- **Sensitivity** is $289 / 1015 = 0.285$
 - 28.5% of those who actually were short of breath are predicted to be short of breath.
- **Specificity** is $2040 / 2219 = 0.919$
 - 91.9% of those who actually weren't short of breath were predicted not to be short of breath.

Summaries of Classification Accuracy, 2

		sbreath		
pred		0	1	Total
Predict No SB		2040	726	2766
Predict SB		179	289	468
Total		2219	1015	3234

- **Positive Predictive Value (PPV)** is $289 / 468 = 0.618$
 - 61.8% of those predicted to be short of breath actually were short of breath.
- **Negative Predictive Value (NPV)** is $2040 / 2766 = 0.738$
 - 73.8% of those predicted to not be short of breath actually were not short of breath.

Using the caret package to get a confusion matrix

```
cmY <- confusionMatrix(  
  data = factor(modY_aug$.fitted >= 0.5),  
  reference = factor(modY_aug$sbreath == 1),  
  positive = "TRUE")
```

Result on the next slide.

Confusion Matrix Output (from caret) for Model Y

```
> cmY
Confusion Matrix and Statistics

          Reference
Prediction FALSE TRUE
FALSE      2040   726
TRUE       179   289

      Accuracy : 0.7202
      95% CI   : (0.7043, 0.7356)
No Information Rate : 0.6861
P-Value [Acc > NIR] : 1.393e-05

      Kappa : 0.239

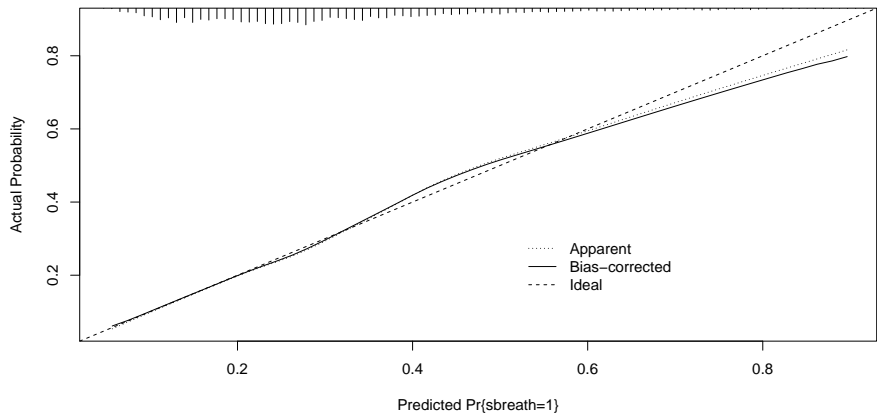
McNemar's Test P-Value : < 2.2e-16

      Sensitivity : 0.28473
      Specificity : 0.91933
      Pos Pred Value : 0.61752
      Neg Pred Value : 0.73753
      Prevalence : 0.31385
      Detection Rate : 0.08936
      Detection Prevalence : 0.14471
      Balanced Accuracy : 0.60203

      'Positive' Class : TRUE
```

Calibration Plot for Model Y

```
plot(calibrate(modY_si))
```



B= 40 repetitions, boot

Mean absolute error=0.009 n=3234

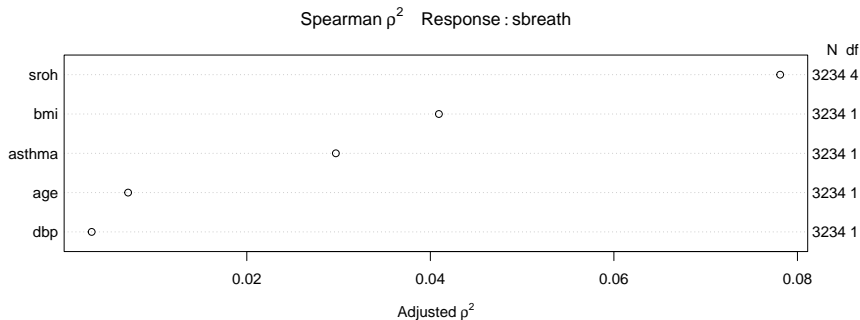
Section 4

Model Z: The “Augmented” model

Considering Non-Linear Terms

- Use 3-6 additional degrees of freedom to account for non-linearity, and add 1-3 non-linear terms.
- We'll start with the Spearman ρ^2 plot...

```
plot(spearman2(sbreath ~ age + sroh + dbp + bmi + asthma,  
              data = nh_today_i))
```



Fitting an “Augmented” Model

We'll include the interaction of `sroh` and `bmi`, which will add 4 df, and a restricted cubic spline with three knots in `bmi` (which will add one more), and stop there.

```
## note: datadist has already been set up

modZ_si <- lrm(sbreath ~ age + sroh + rcs(bmi,3) +
               sroh %ia% bmi + dbp + asthma,
               data = nh_today_i, x = TRUE, y = TRUE)
```

modZ_si results (from lrm fit)

```
> modZ_si
Logistic Regression Model

lrm(formula = sbreath ~ age + sroh + rcs(bmi, 3) + sroh %ia%
     bmi + dbp + asthma, data = nh_today_i, x = TRUE, y = TRUE)


```

		Model Likelihood	Discrimination	Rank Discrim.			
		Ratio Test	Indexes	Indexes			
Obs	3234	LR chi2	428.25	R2	0.174	C	0.719
0	2219	d.f.	13	R2(13,3234)	0.120	Dxy	0.438
1	1015	Pr(> chi2)	<0.0001	R2(13,2089.3)	0.180	gamma	0.438
max deriv	4e-09			Brier	0.187	tau-a	0.189

	Coef	S.E.	Wald Z	Pr(> Z)
Intercept	-4.2765	1.1558	-3.70	0.0002
age	0.0148	0.0039	3.75	0.0002
sroh=VG	0.4052	1.1722	0.35	0.7295
sroh=G	1.5974	1.0686	1.49	0.1350
sroh=F	2.1630	1.0985	1.97	0.0489
sroh=P	2.9370	1.2318	2.38	0.0171
bmi	0.0596	0.0387	1.54	0.1235
bmi'	0.0095	0.0207	0.46	0.6461
sroh=VG * bmi	0.0023	0.0402	0.06	0.9539
sroh=G * bmi	-0.0151	0.0368	-0.41	0.6811
sroh=F * bmi	-0.0100	0.0376	-0.27	0.7907
sroh=P * bmi	-0.0323	0.0414	-0.78	0.4352
dbp	-0.0061	0.0038	-1.62	0.1047
asthma=Yes	0.8017	0.1111	7.21	<0.0001

glm version of Model Z

```
modZ_si_g <- glm(sbreath ~ age + sroh + rcs(bmi, 3) +  
                  sroh %ia% bmi + dbp + asthma,  
                  data =nh_today_i,  
                  family =binomial(link =logit))
```

```
> modZ_si_g
```

```
Call: glm(formula = sbreath ~ age + sroh + rcs(bmi, 3) + sroh %ia%  
        bmi + dbp + asthma, family = binomial(link = logit), data = nh_today_i)
```

Coefficients:

(Intercept)	age	srohVG	srohG
-4.276453	0.014819	0.405248	1.597374
srohF	srohP	rcs(bmi, 3)bmi	rcs(bmi, 3)bmi'
2.163030	2.937031	0.059582	0.009525
sroh %ia% bmisroh=VG * bmi	sroh %ia% bmisroh=G * bmi	sroh %ia% bmisroh=F * bmi	sroh %ia% bmisroh=P * bmi
0.002326	-0.015125	-0.009987	-0.032268
dbp	asthmaYes		
-0.006146	0.801692		

Degrees of Freedom: 3233 Total (i.e. Null); 3220 Residual

Null Deviance: 4024

Residual Deviance: 3596

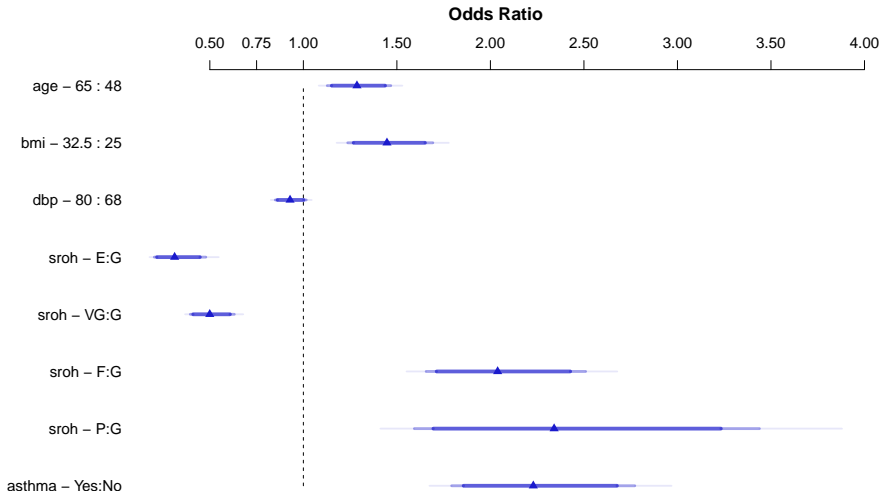
AIC: 3624

Tidied Table of Model Z (Exponentiated) Coefficients

term	estimate	std.error	low90	high90
(Intercept)	0.014	1.156	0.002	0.092
age	1.015	0.004	1.008	1.022
srohVG	1.500	1.172	0.215	10.758
srohG	4.940	1.068	0.836	30.246
srohF	8.697	1.098	1.402	55.705
srohP	18.860	1.232	2.436	147.704
rsc(bmi, 3)bmi	1.061	0.039	0.995	1.132
rsc(bmi, 3)bmi'	1.010	0.021	0.976	1.045
sroh %ia% bmisroh=VG * bmi	1.002	0.040	0.937	1.072
sroh %ia% bmisroh=G * bmi	0.985	0.037	0.926	1.049
sroh %ia% bmisroh=F * bmi	0.990	0.038	0.930	1.055
sroh %ia% bmisroh=P * bmi	0.968	0.041	0.904	1.038
dbp	0.994	0.004	0.988	1.000
asthmaYes	2.229	0.111	1.857	2.677

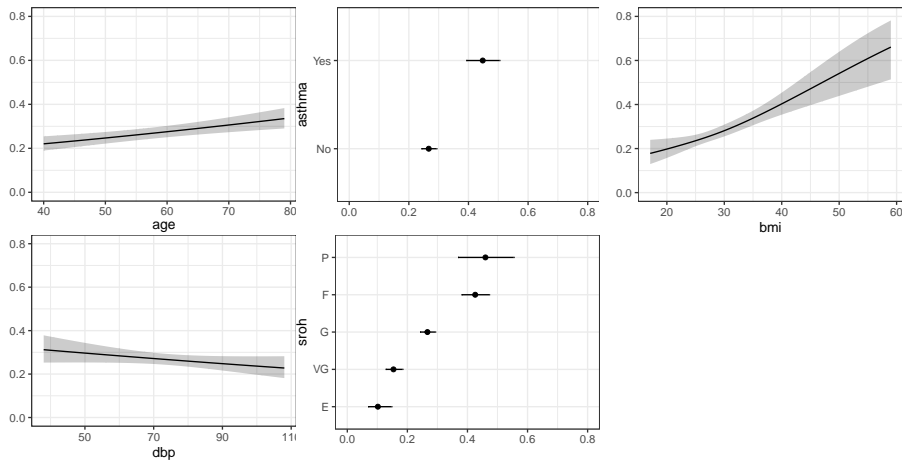
Model Z Effects Plot on Odds Ratio Scale

```
plot(summary(modZ_si))
```



Prediction Plot for Model Z

```
ggplot(Predict(modZ_si, fun = plogis))
```



Confusion Matrix for Model Z

How well does our Model Z classify subjects using a decision rule at 0.5?

```
modZ_aug <- augment(modZ_si_g, type.predict = "response")

modZ_aug <- modZ_aug |>
  mutate(pred = ifelse(.fitted >= 0.5,
                        "Predict SB", "Predict No SB"))

modZ_aug |> tabyl(pred, sbreath) |>
  adorn_totals(where = c("row", "col")) |> adorn_title()
```

	sbreath		
pred	0	1	Total
Predict No SB	2040	725	2765
Predict SB	179	290	469
Total	2219	1015	3234

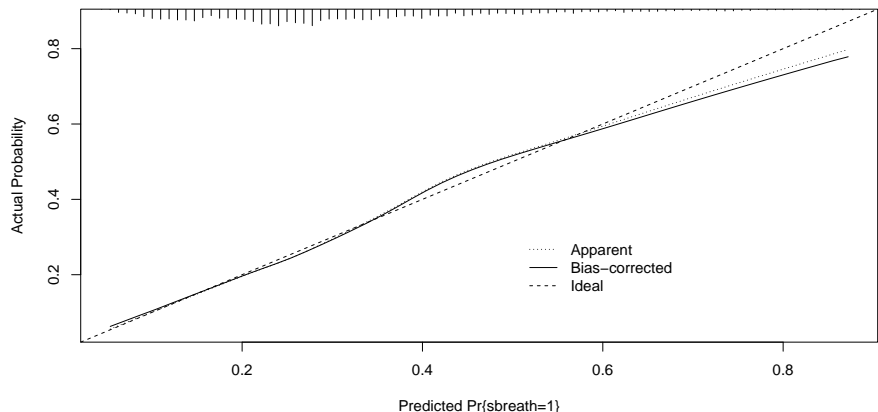
Key Summaries of Classification Accuracy

		sbreath		
		0	1	Total
Predict	No SB	2040	725	2765
	SB	179	290	469
Total		2219	1015	3234

- **Sensitivity** is only slightly changed, to $290 / 1015 = 0.286$
 - 28.6% of those who actually were short of breath are predicted to be short of breath.
- **Specificity** is still $2040 / 2219 = 0.919$
 - 91.9% of those who actually weren't short of breath were predicted not to be short of breath.
- **Positive Predictive Value (PPV)** is $290 / 469 = 0.618$
 - Again, 61.8% of those predicted to be short of breath actually were short of breath.

Calibration Plot for Model Z

```
plot(calibrate(modZ_si))
```



B= 40 repetitions, boot

Mean absolute error=0.01 n=3234

Section 5

Summarizing/Presenting a Final Model

Compare Models Y and Z on Key Summaries

```
temp1 <- bind_rows(glance(modY_si_g), glance(modZ_si_g)) |>
  mutate(model = c("Y", "Z")) |>
  select(model, AIC, BIC)

temp2 <- tibble(model = c("Y", "Z"),
  auc = c(modY_si$stats["C"], modZ_si$stats["C"]),
  r2_nag = c(modY_si$stats["R2"], modZ_si$stats["R2"]))

left_join(temp1, temp2, by = "model") |> kable()
```

model	AIC	BIC	auc	r2_nag
Y	3615.664	3670.398	0.7181358	0.1735251
Z	3623.808	3708.949	0.7188002	0.1742317

ANOVA comparing Model Y to Z

```
> anova(modZ_si)
```

	Wald Statistics		Response: sbreath
Factor	Chi-Square	d.f.	P
age	14.08	1	0.0002
sroh (Factor+Higher Order Factors)	165.26	8	<.0001
All Interactions	1.79	4	0.7745
bmi (Factor+Higher Order Factors)	79.65	6	<.0001
All Interactions	1.79	4	0.7745
Nonlinear	0.21	1	0.6461
sroh * bmi (Factor+Higher Order Factors)	1.79	4	0.7745
dbp	2.63	1	0.1047
asthma	52.03	1	<.0001
TOTAL NONLINEAR + INTERACTION	1.89	5	0.8642
TOTAL	352.90	13	<.0001

```
> |
```

Validating Model Summaries (code)

```
set.seed(432123)

valY <- validate(modY_si, B = 40)
valZ <- validate(modZ_si, B = 40)

val_1 <- bind_rows(valY[1,], valZ[1,]) |>
  mutate(model = c("Y", "Z"),
         AUC_nominal = 0.5 + (index.orig/2),
         AUC_validated = 0.5 + (index.corrected/2)) |>
  select(model, AUC_nominal, AUC_validated)

val_2 <- bind_rows(valY[2,], valZ[2,]) |>
  mutate(model = c("Y", "Z"),
         R2_nominal = index.orig,
         R2_validated = index.corrected) |>
  select(model, R2_nominal, R2_validated)
```

See next slide for the result.

Validating Model Summaries

```
val <- left_join(val_1, val_2, by = "model")
```

```
val |> kable()
```

model	AUC_nominal	AUC_validated	R2_nominal	R2_validated
Y	0.7181438	0.7163131	0.1735251	0.1692934
Z	0.7187991	0.7132489	0.1742317	0.1632280

Which model should we choose?

Describing a Meaningful Effect

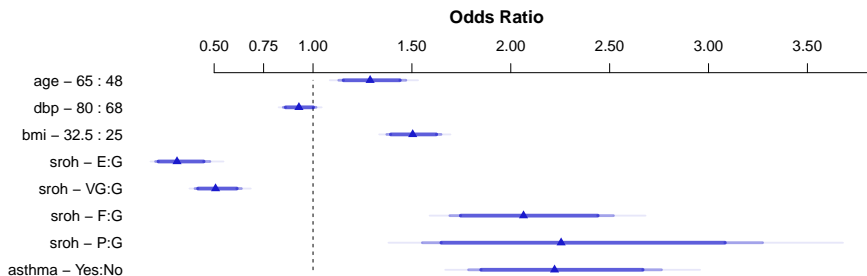
This is for you to do.

[W]rite a detailed and correct description of the effect of at least one predictor on your outcome for your chosen logistic regression model, providing all necessary elements of such a description, and link this directly to what the (effects) plot is telling you.

See Chapter 22 of the Notes for more details, and this is also the major task in several questions within Lab 4. The effects plot for Model Y is repeated in the next slide, and you'll want the actual summary as well as the plot so you can specify the numbers. We prefer you discuss a meaningful effect, should one exist. Pick an effect to describe that is interesting to you.

Model Y Effects Plot on Odds Ratio Scale

```
plot(summary(modY_si))
```



ROC Calculations for Model Y

```
roc_modY <- roc(nh_today_i$sbreath ~  
  predict(modY_si_g, type="response"), ci = TRUE)
```

```
roc_modY
```

Call:

```
roc.formula(formula = nh_today_i$sbreath ~ predict(modY_si_g,
```

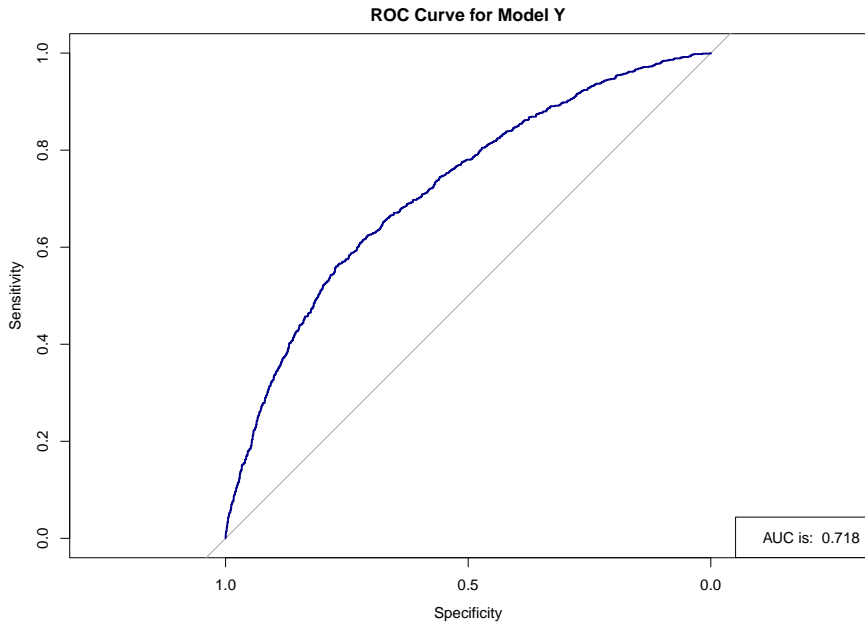
Data: predict(modY_si_g, type = "response") in 2219 controls

Area under the curve: 0.7181

95% CI: 0.6992-0.7371 (DeLong)

```
plot(roc_modY, main = "ROC Curve for Model Y",  
     lwd = 2, col = "salmon")  
legend('bottomright',  
     legend = paste("AUC is: ",round_half_up(auc(roc_modY),3)))
```

ROC plot for Model Y



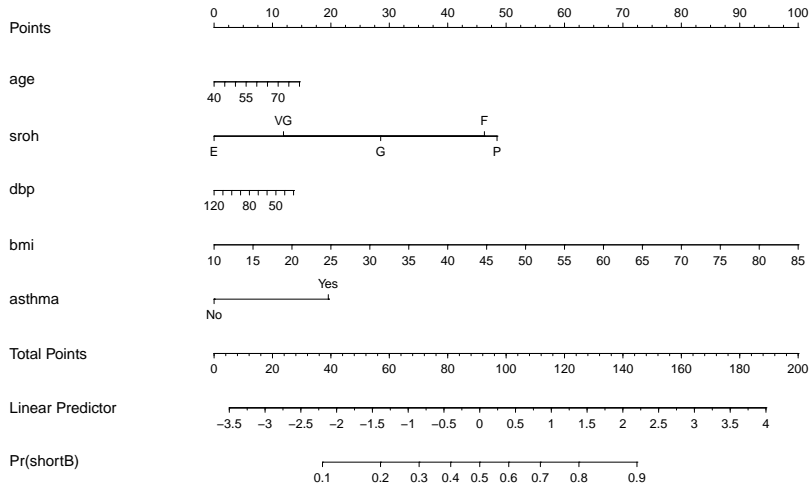
Nomogram for the Final Model (Model Y)

```
plot(nomogram(modY_si, fun = plogis,  
             fun.at=c(seq(0.1, 0.9, by = 0.1)),  
             funlabel = "Pr(shortB)"))
```

Result on next slide.

- The final part of your summary of the final model should be a nomogram **with a demonstration of a predicted probability associated with two new subjects of interest** that differ in terms of some of the parameters in your model.
- Your predictions should describe two different subjects. You don't have to call them Harry and Sally, but it is helpful to give them actual names.

Nomogram for Model Y



Next Time

More on Logistic Regression

- See section 20 for more on confusion matrices and ROC curves and some material on assessing assumptions through residual plots, all of which are in the context of logistic models fit with `glm()`.
- See section 21 for more on using Spearman's ρ^2 plot, Nagelkerke R^2 , the C statistic, its relationship to Somers' d, validation and plotting the results, along with some thoughts on identifying influential points, mostly in the context of logistic models fit with `lrm()`.
- See section 22 for some thoughts on estimating and interpreting effect sizes in logistic and in linear models. Some really useful tips to be found here.