

Logistic Regression Fitting and Multiple Imputation: Frequently Asked Questions after the Quiz 1 Honors Opportunity

Thomas E. Love for 432

To be discussed 2019-04-02: version 2019-03-29

```
library(rms); library(broom); library(NHANES)
library(tidyverse)
```

1 A Sample Data Set

We'll pull a set of NHANES data from the 2011-12 administration.

```
nh <- NHANES %>%
  filter(SurveyYr == "2011_12",
         Age >= 21, Age <= 64,
         Work %in% c("Working", "NotWorking"),
         !is.na(Diabetes)) %>%
  droplevels() %>%
  select(ID, SurveyYr, Age, HomeOwn, Education, BMI, Pulse, Work, Diabetes, SleepTrouble)

summary(nh)
```

ID	SurveyYr	Age	HomeOwn
Min. :62172	2011_12:2757	Min. :21.00	Own :1675
1st Qu.:64582		1st Qu.:31.00	Rent :1012
Median :67014		Median :43.00	Other: 58
Mean :67055		Mean :42.43	NA's : 12
3rd Qu.:69537		3rd Qu.:53.00	
Max. :71915		Max. :64.00	

Education	BMI	Pulse	Work
8th Grade :127	Min. :16.70	Min. : 40.00	NotWorking: 725
9 - 11th Grade:304	1st Qu.:24.10	1st Qu.: 64.00	Working :2032
High School :505	Median :27.80	Median : 72.00	
Some College :887	Mean :28.76	Mean : 73.03	
College Grad :934	3rd Qu.:32.00	3rd Qu.: 80.00	
	Max. :80.60	Max. :128.00	
	NA's :20	NA's :95	

Diabetes	SleepTrouble
No :2550	No :2033
Yes: 207	Yes: 724

2 m1 = A Simple Logistic Regression Model with lrm

In Model m1, let's predict the log odds of `Diabetes` being "Yes" across the 2,757 subjects in these data, on the basis of `Age`, alone.

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

m1 <- lrm((Diabetes == "Yes") ~ Age,
          data = nh, x = TRUE, y = TRUE)

m1
```

Logistic Regression Model

```
lrm(formula = (Diabetes == "Yes") ~ Age, data = nh, x = TRUE,
     y = TRUE)
```

		Model Likelihood		Discrimination		Rank Discrim.	
		Ratio Test		Indexes		Indexes	
Obs	2757	LR chi2	121.40	R2	0.104	C	0.723
FALSE	2550	d.f.	1	g	1.016	Dxy	0.445
TRUE	207	Pr(> chi2)	<0.0001	gr	2.762	gamma	0.455
max deriv	1e-09			gp	0.062	tau-a	0.062
				Brier	0.066		

	Coef	S.E.	Wald Z	Pr(> Z)
Intercept	-5.7914	0.3625	-15.98	<0.0001
Age	0.0700	0.0070	10.01	<0.0001

2.1 What is the effect of Age in model m1?

By default, `summary` within `lrm` shows the impact of moving from the 25th percentile of a quantitative predictor (like `Age`) to the 75th percentile.

```
summary(m1)
```

	Effects					Response : (Diabetes == "Yes")
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95 Upper 0.95
Age	31	53	22	1.5411	0.15391	1.2394 1.8427
Odds Ratio	31	53	22	4.6697	NA	3.4537 6.3139

OK. That's the default. We can plot that, and so forth. The estimated odds ratio is 4.67 with 95% confidence interval (3.45, 6.31). This describes the impact of moving from Age 31 to Age 53, which represent the 25th and 75th percentiles of `Age`, respectively.

2.1.1 What if we wanted a different confidence level?

```
summary(m1, conf.int = .90)
```

	Effects					Response : (Diabetes == "Yes")
--	---------	--	--	--	--	--------------------------------

Factor	Low	High	Diff.	Effect	S.E.	Lower 0.9	Upper 0.9
Age	31	53	22	1.5411	0.15391	1.2879	1.7942
Odds Ratio	31	53	22	4.6697	NA	3.6253	6.0149

2.1.2 What if we wanted to show the effect of a one-year change in Age?

Suppose that instead of knowing the impact of moving from Age 31 to 53, we want to know the impact of moving from Age 31 to 32?

```
summary(m1, Age = c(31,32))
```

Effects					Response : (Diabetes == "Yes")		
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95
Age	31	32	1	0.07005	0.0069959	0.056338	0.083761
Odds Ratio	31	32	1	1.07260	NA	1.058000	1.087400

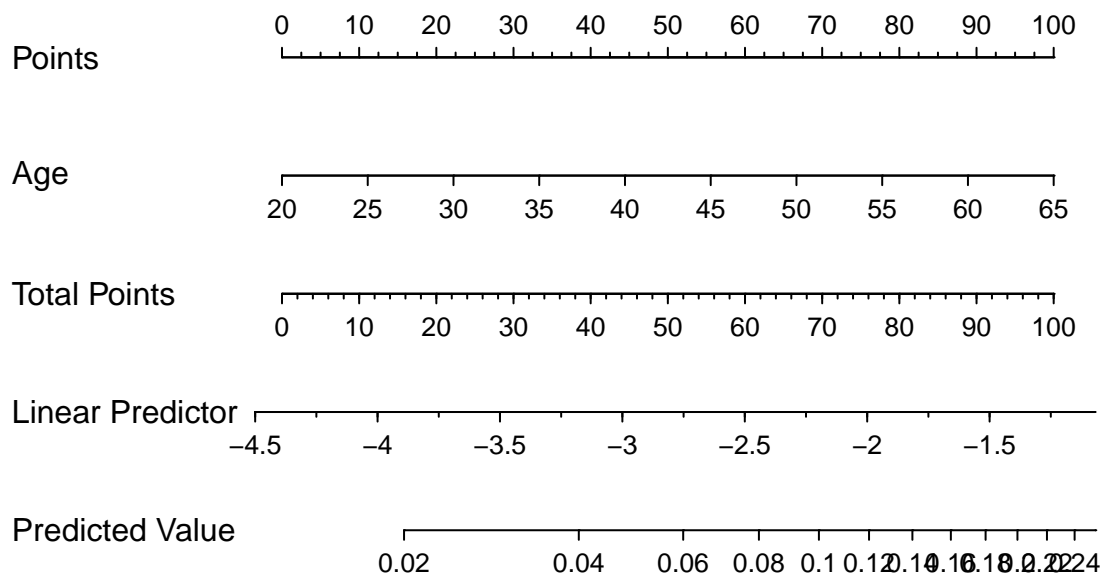
How about moving from 51 to 52? Any difference?

```
summary(m1, Age = c(51,52))
```

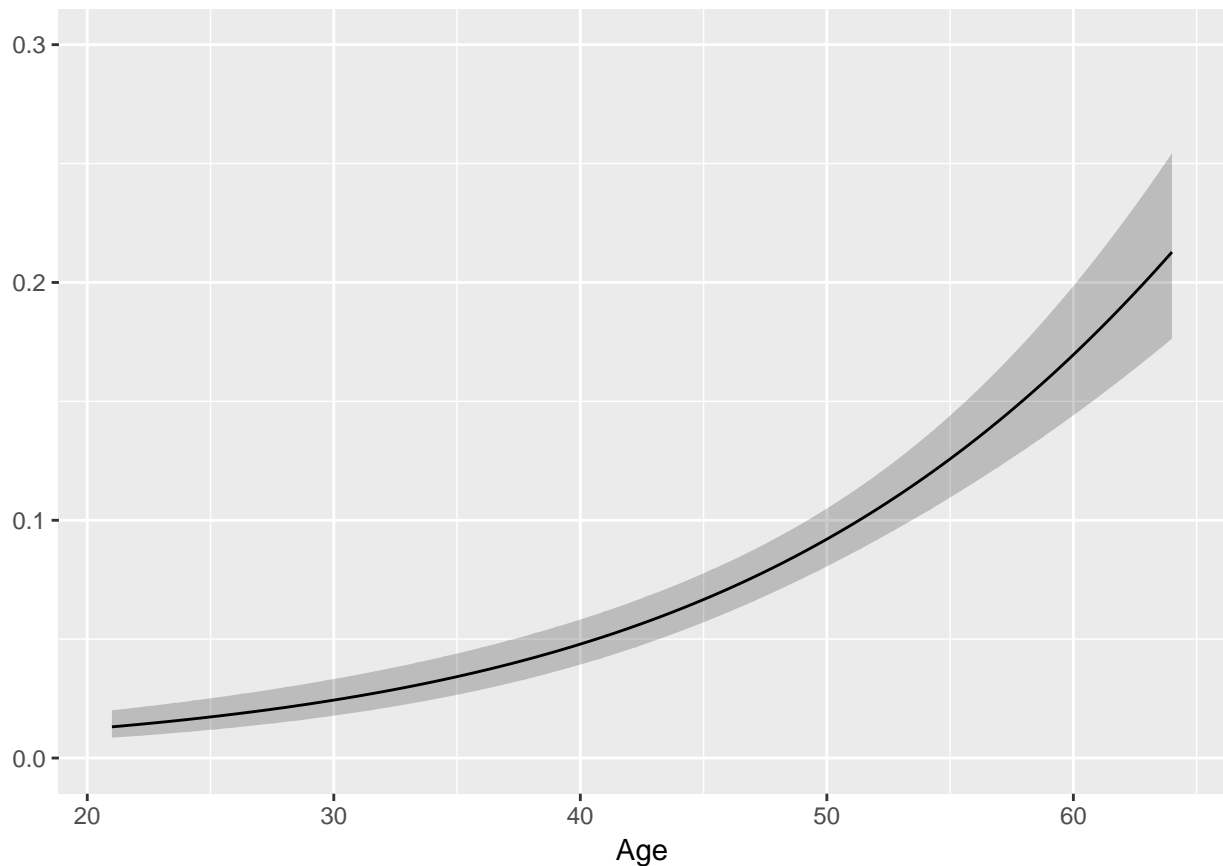
Effects					Response : (Diabetes == "Yes")		
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95
Age	51	52	1	0.07005	0.0069959	0.056338	0.083761
Odds Ratio	51	52	1	1.07260	NA	1.058000	1.087400

Here, the effect of moving from 31 to 32 is the same as moving from 51 to 52, or, indeed, moving by one year from any starting Age, because the model includes only the main effect of Age, and is linear in Age. We can see that easily in, for example, a nomogram, or a prediction plot (`ggplot(Predict())`)...

```
plot(nomogram(m1, fun = plogis))
```



```
ggplot(Predict(m1, fun = plogis))
```



2.2 Predicting Alice's probability of diabetes

Suppose Alice is 35 years old. What is her predicted probability of diabetes, according to model `m1`?

```
predict(m1, newdata = tibble(Age = 35),
        type = "fitted")
```

```
1
0.03423479
```

2.3 Comparison to what we get from `glm`

```
g1 <- glm((Diabetes == "Yes") ~ Age,
          data = nh, family = binomial())
```

```
exp(coef(g1)); exp(confint(g1))
```

```
(Intercept)      Age
0.003053669 1.072561288
```

Waiting for profiling to be done...

```
2.5 %      97.5 %
(Intercept) 0.001464804 0.00607422
Age         1.058261055 1.08771521
```

Or use broom!

```
tidy(g1, exponentiate = TRUE, conf.int = TRUE)
```

```
# A tibble: 2 x 7
  term      estimate std.error statistic  p.value conf.low conf.high
  <chr>      <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
1 (Intercept) 0.00305   0.362    -16.0 1.82e-57 0.00146 0.00607
2 Age          1.07    0.00700    10.0 1.34e-23 1.06    1.09
```

and this is, indeed, the same answer we would get from our rms fit: m1 comparing any one-year change in Age for this model.

```
summary(m1, Age = c(41,42))
```

	Effects			Response : (Diabetes == "Yes")				
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95	
Age	41	42	1	0.07005	0.0069959	0.056338	0.083761	
Odds Ratio	41	42	1	1.07260	NA	1.058000	1.087400	

2.3.1 Does the prediction for Alice match up, too?

The prediction for Alice we get from g1 matches the one we saw in m1, as well, once we deal with the fact that the appropriate type of prediction to get a probability uses type = "fitted" for a fit from rms and type = "response" for a glm fit from base R.

```
predict(g1, newdata = tibble(Age = 35),
        type = "response")
```

```
1
0.03423479
```

3 What if there was a non-linear Age effect, as in Model m2?

Let's add a restricted cubic spline with three knots in Age to incorporate a non-linear effect.

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

m2 <- lrm((Diabetes == "Yes") ~ rcs(Age, 3),
        data = nh, x = TRUE, y = TRUE)
```

```
m2
```

Logistic Regression Model

```
lrm(formula = (Diabetes == "Yes") ~ rcs(Age, 3), data = nh, x = TRUE,
    y = TRUE)
```

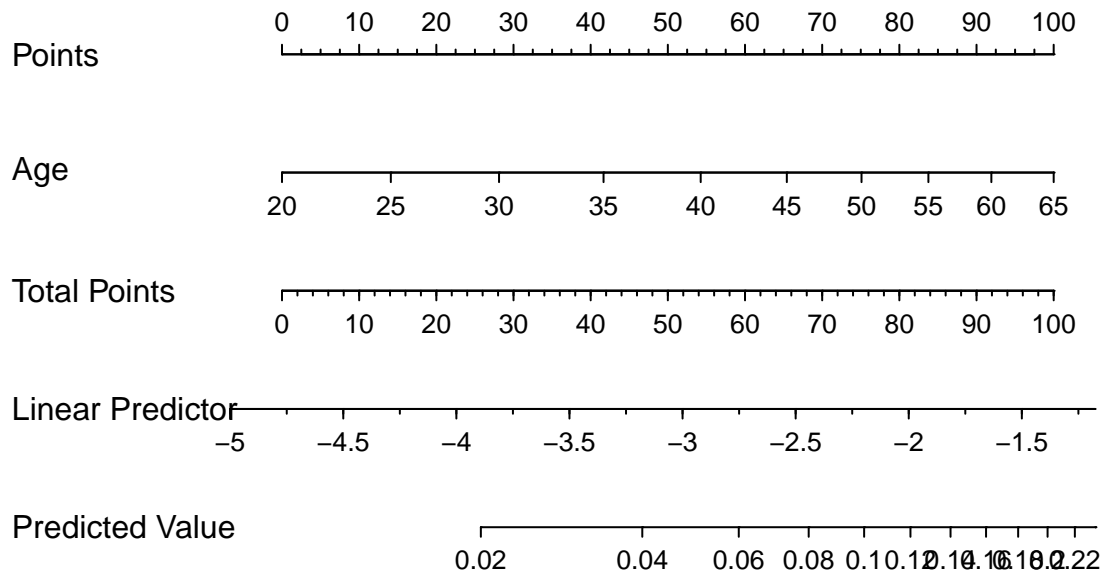
		Model Likelihood		Discrimination		Rank Discrim.	
		Ratio Test		Indexes		Indexes	
Obs	2757	LR chi2	122.76	R2	0.105	C	0.723
FALSE	2550	d.f.	2	g	1.104	Dxy	0.445
TRUE	207	Pr(> chi2)	<0.0001	gr	3.016	gamma	0.455

max deriv	1e-05	gp	0.062	tau-a	0.062
		Brier	0.066		

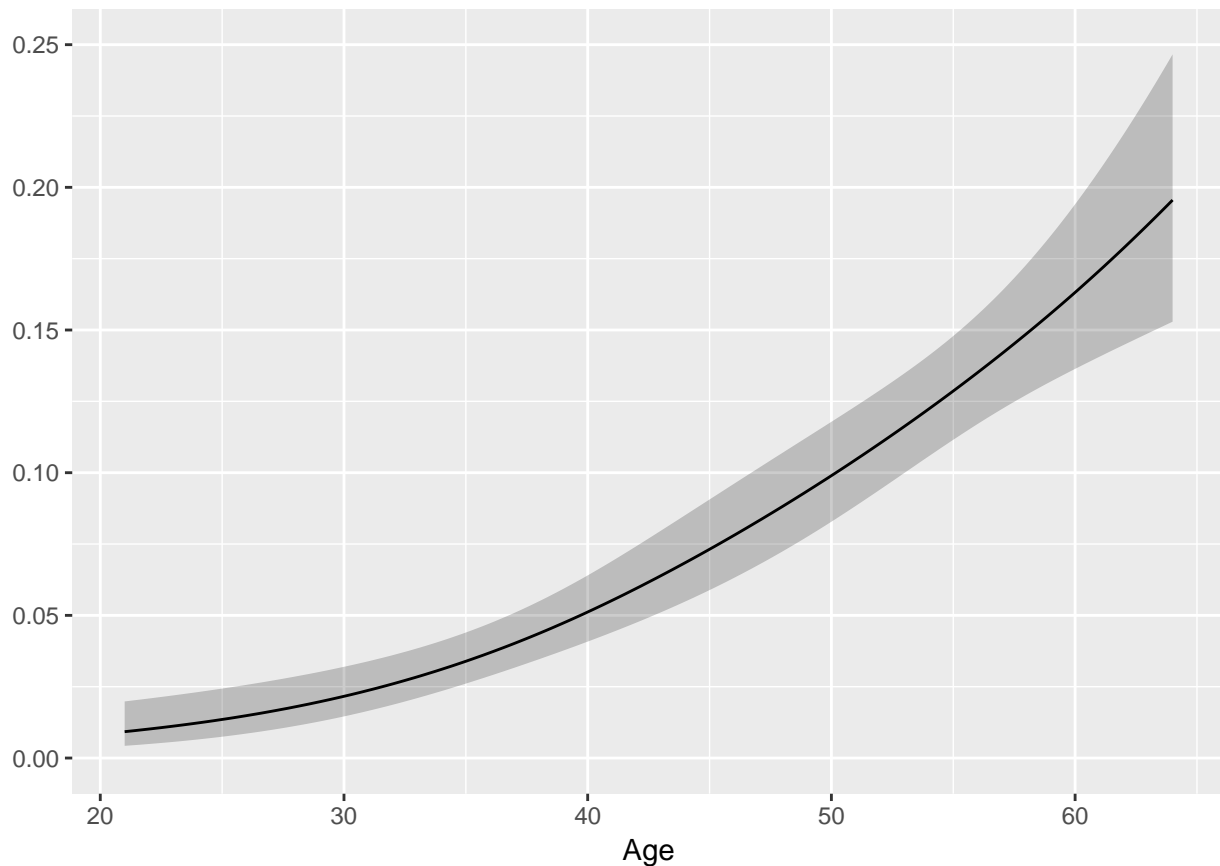
	Coef	S.E.	Wald Z	Pr(> Z)
Intercept	-6.6954	0.8922	-7.50	<0.0001
Age	0.0962	0.0243	3.96	<0.0001
Age'	-0.0267	0.0233	-1.15	0.2520

3.1 Impact of the Non-Linear Term here in Age?

```
plot(nomogram(m2, fun = plogis))
```



```
ggplot(Predict(m2, fun = plogis))
```



3.2 Now what is the effect of Age in m2?

3.2.1 m2: Default summary - move from Age 31 to 53

As we move from the 25th percentile (Age 31) to the 75th percentile (Age 53), we have...

```
summary(m2)
```

	Effects			Response : (Diabetes == "Yes")				
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95	
Age	31	53	22	1.6888	0.2105	1.2762	2.1014	
Odds Ratio	31	53	22	5.4130	NA	3.5831	8.1775	

3.2.2 m2: Effect of moving from Age 31 to 32?

As we move by just one year, from Age 31 to 32, we have...

```
summary(m2, Age = c(31, 32))
```

	Effects			Response : (Diabetes == "Yes")				
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95	
Age	31	32	1	0.09346	0.022001	0.050338	0.13658	
Odds Ratio	31	32	1	1.09800	NA	1.051600	1.14630	

3.2.3 m2: Effect of moving from Age 51 to 52 now isn't the same as 31 to 32?

But now this won't be the same as what we see when we move from Age 51 to 52, because of the non-linear effect (thanks to the restricted cubic spline in Age we included in this model.)

```
summary(m2, Age = c(51, 52))
```

	Effects			Response : (Diabetes == "Yes")			
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95
Age	51	52	1	0.060103	0.011106	0.038336	0.081869
Odds Ratio	51	52	1	1.061900	NA	1.039100	1.085300

3.3 Predicting Alice's probability of diabetes

Suppose Alice is 35 years old. What is her predicted probability of diabetes, according to model m2?

```
predict(m2, newdata = tibble(Age = 35),  
        type = "fitted")
```

```
1  
0.03391639
```

4 Fitting m3 to make things more complex

4.1 m3 includes a spline in Age, and an interaction with obesity...

```
nh1 <- nh %>%  
  mutate(obese = ifelse(BMI >= 30, 1, 0),  
         diabetes = ifelse(Diabetes == "Yes", 1, 0))  
  
d <- datadist(nh1)  
options(datadist = "d")  
  
m3 <- lrm(diabetes ~ rcs(Age, 3) + obese +  
         Age %ia% obese,  
         data = nh1, x = TRUE, y = TRUE)
```

m3

Frequencies of Missing Values Due to Each Variable

diabetes	Age	obese
0	0	20

Logistic Regression Model

```
lrm(formula = diabetes ~ rcs(Age, 3) + obese + Age %ia% obese,  
    data = nh1, x = TRUE, y = TRUE)
```

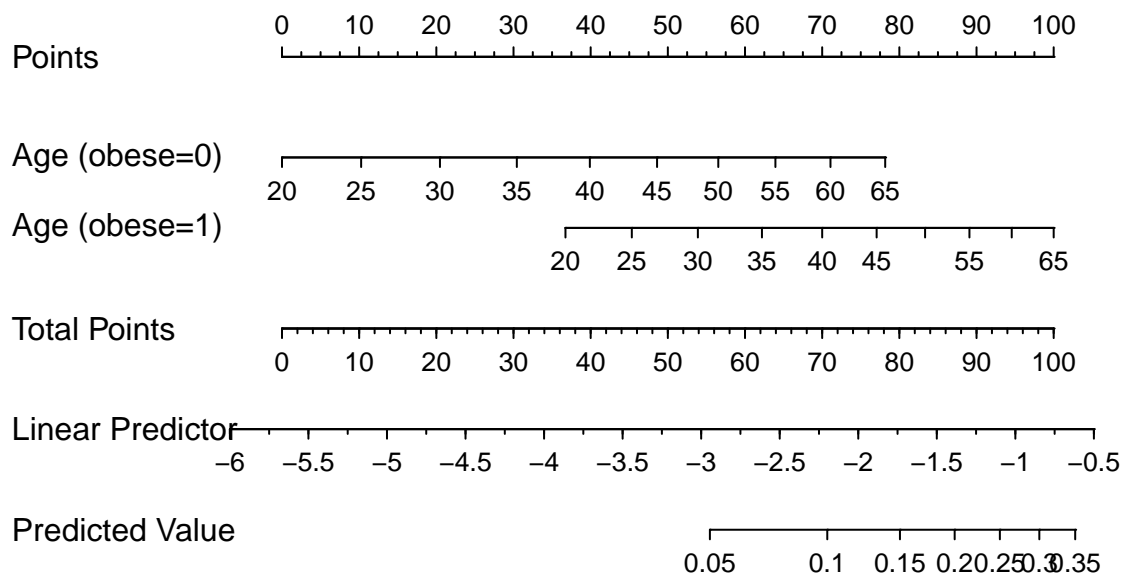
		Model Likelihood		Discrimination		Rank Discrim.		
		Ratio Test		Indexes		Indexes		
Obs	2737	LR	chi2	197.37	R2	0.169	C	0.780

0	2532	d.f.	4	g	1.440	Dxy	0.559
1	205	Pr(> chi2)	<0.0001	gr	4.221	gamma	0.565
max deriv	1e-08			gp	0.077	tau-a	0.078
				Brier	0.064		

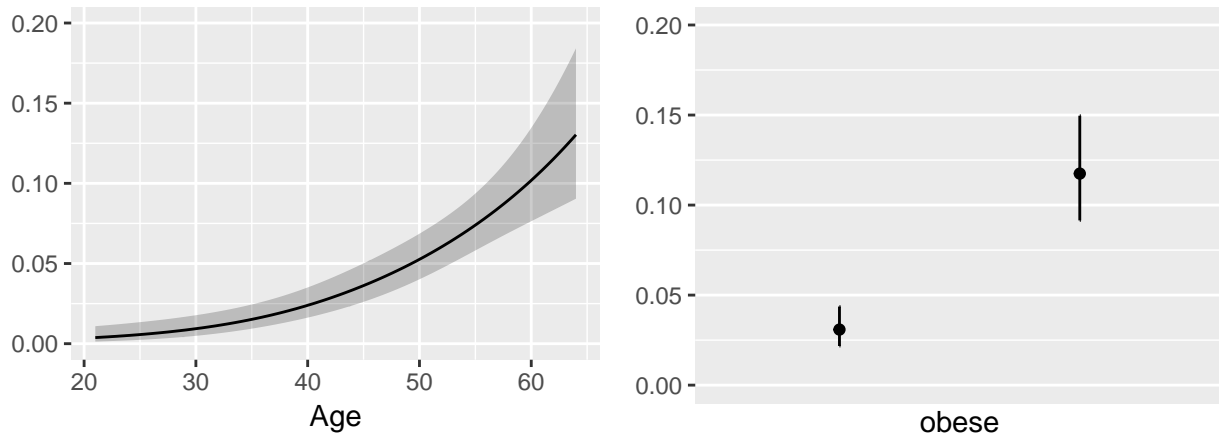
	Coef	S.E.	Wald Z	Pr(> Z)
Intercept	-7.6832	1.0973	-7.00	<0.0001
Age	0.1007	0.0277	3.64	0.0003
Age'	-0.0201	0.0239	-0.84	0.3997
obese	2.1296	0.8212	2.59	0.0095
Age * obese	-0.0163	0.0157	-1.03	0.3010

4.2 Nomogram and Prediction Plot for Model m3

```
plot(nomogram(m3, fun = plogis))
```



```
ggplot(Predict(m3, fun = plogis))
```



4.3 What is the effect of Age, in model m3?

It depends.

4.3.1 Age 31 to Age 53 in a non-obese subject

```
summary(m3)
```

Effects				Response : diabetes			
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95
Age	31	53	22	1.8930	0.31848	1.2688	2.5172
Odds Ratio	31	53	22	6.6390	NA	3.5564	12.3930
obese	0	1	1	1.4303	0.20384	1.0308	1.8298
Odds Ratio	0	1	1	4.1799	NA	2.8032	6.2327

Adjusted to: Age=43 obese=0

Note the Adjusted to `obese = 0`, which means that this odds ratio for Age is assuming that `obese = 0`.

4.3.2 Age 31 to Age 53 in an obese subject

```
summary(m3, obese = 1)
```

Effects			Response : diabetes			
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95 Upper 0.95
Age	31	53	22	1.5352	0.23942	1.0659 2.0044
Odds Ratio	31	53	22	4.6422	NA	2.9035 7.4219
obese	0	1	1	1.4303	0.20384	1.0308 1.8298
Odds Ratio	0	1	1	4.1799	NA	2.8032 6.2327

Adjusted to: Age=43 obese=1

Now we see a different odds ratio for the effect of moving from Age 31 to 53, when the subject is in fact obese.

4.4 What about a one-year change in Age?

4.4.1 Age 31 to Age 32 in a non-obese subject

```
summary(m3, Age = c(31,32))
```

Effects			Response : diabetes			
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95 Upper 0.95
Age	31	32	1	0.098661	0.025515	0.048653 0.14867
Odds Ratio	31	32	1	1.103700	NA	1.049900 1.16030
obese	0	1	1	1.430300	0.203840	1.030800 1.82980
Odds Ratio	0	1	1	4.179900	NA	2.803200 6.23270

Adjusted to: Age=43 obese=0

Note that the effect shown here (odds ratio = 1.08) is the effect of moving from Age 31 to Age 32, in model m3, assuming the subject is not obese (obese = 0), as indicated.

4.4.2 Effect of moving from age 31 to 32 for an obese subject?

```
summary(m3, Age = c(31,32), obese = 1)
```

Effects			Response : diabetes			
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95 Upper 0.95
Age	31	32	1	0.082398	0.022581	0.03814 0.12666
Odds Ratio	31	32	1	1.085900	NA	1.03890 1.13500
obese	0	1	1	1.430300	0.203840	1.03080 1.82980
Odds Ratio	0	1	1	4.179900	NA	2.80320 6.23270

Adjusted to: Age=43 obese=1

The change we see is due to the fact that an interaction between Age and obese was included in the model m3.

4.4.3 Effect of moving from age 51 to 52 for a non-obese subject?

```
summary(m3, Age = c(51,52))
```

Effects			Response : diabetes					
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95	
Age	51	52	1	0.073453	0.014715	0.044611	0.10229	
Odds Ratio	51	52	1	1.076200	NA	1.045600	1.10770	
obese	0	1	1	1.430300	0.203840	1.030800	1.82980	
Odds Ratio	0	1	1	4.179900	NA	2.803200	6.23270	

Adjusted to: Age=43 obese=0

Note that this odds ratio is different than the one we saw for moving from Age 31 to 32, because of the non-linear (spline) terms in Age included in m3.

4.4.4 Effect of moving from age 51 to 52 for an obese subject?

```
summary(m3, Age = c(51,52), obese = 1)
```

Effects			Response : diabetes					
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95	
Age	51	52	1	0.05719	0.013239	0.031241	0.083139	
Odds Ratio	51	52	1	1.05890	NA	1.031700	1.086700	
obese	0	1	1	1.43030	0.203840	1.030800	1.829800	
Odds Ratio	0	1	1	4.17990	NA	2.803200	6.232700	

Adjusted to: Age=43 obese=1

Again, we see the impact of the interaction term.

4.5 Predicting Alice's probability of diabetes

Suppose Alice is 35 years old. To make a prediction for her using model m3, we'd have to specify whether or not she is obese, or at least compare those two predicted probabilities. So what do we get?

```
predict(m3,
  newdata = tibble(names = c("Alice A", "Alice B"),
    Age = c(35,35), obese = c(0,1)),
  type = "fitted")
```

```
1      2
0.01516586 0.06830481
```

So if Alice is obese, her predicted probability of diabetes is much larger than if she is not. That makes sense, given the nomogram, and prediction plot we've seen.

5 Multiple Imputation with a Logistic Regression Model

5.1 Adding Pulse to Model m3

Now consider a model for diabetes that includes the Pulse rate, and leads to more substantial missingness, as a result.

```
m4 <- lrm(diabetes ~ rcs(Age, 3) + obese + Pulse +
          Age %ia% obese,
          data = nh1, x = TRUE, y = TRUE)
```

m4

Frequencies of Missing Values Due to Each Variable

diabetes	Age	obese	Pulse
0	0	20	95

Logistic Regression Model

```
lrm(formula = diabetes ~ rcs(Age, 3) + obese + Pulse + Age %ia%
    obese, data = nh1, x = TRUE, y = TRUE)
```

		Model Likelihood		Discrimination		Rank Discrim.	
		Ratio Test		Indexes		Indexes	
Obs	2646	LR chi2	213.49	R2	0.186	C	0.794
0	2445	d.f.	5	g	1.480	Dxy	0.589
1	201	Pr(> chi2)	<0.0001	gr	4.393	gamma	0.589
max deriv	4e-08			gp	0.081	tau-a	0.083
				Brier	0.064		

	Coef	S.E.	Wald Z	Pr(> Z)
Intercept	-9.8454	1.2448	-7.91	<0.0001
Age	0.1072	0.0283	3.80	0.0001
Age'	-0.0268	0.0245	-1.10	0.2730
obese	1.8427	0.8305	2.22	0.0265
Pulse	0.0266	0.0063	4.19	<0.0001
Age * obese	-0.0109	0.0159	-0.68	0.4934

Suppose we want to use multiple imputation to deal with this missingness.

5.2 nh_imp = The Imputation Model

We'll run an imputation model with 10 imputations, using 0 or 3 knots to represent non-linear terms. I usually take either this or the default (no knots) approach in practical work.

```
set.seed(432)
d <- datadist(nh1)
options(datadist = "d")

nh_imp <- aregImpute(~ diabetes + Age + obese + Pulse,
                    nk = c(0, 3),
                    tlinear = TRUE, data = nh1,
                    n.impute = 10, pr = FALSE)
```

5.3 m5 = The Fitted Model after Multiple Imputation for diabetes

Let's fit the outcome model now, after multiple imputation.

```
d <- datadist(nh1)
options(datadist = "d")

m5 <- fit.mult.impute(diabetes ~ rcs(Age, 3) + obese +
  Pulse + Age %ia% obese,
  fitter = lrm, xtrans = nh_imp,
  data = nh1, x = TRUE, y = TRUE)
```

Variance Inflation Factors Due to Imputation:

Intercept	Age	Age'	obese	Pulse	Age * obese
1.01	1.00	1.00	1.01	1.01	1.01

Rate of Missing Information:

Intercept	Age	Age'	obese	Pulse	Age * obese
0.01	0.00	0.00	0.01	0.01	0.01

d.f. for t-distribution for Tests of Single Coefficients:

Intercept	Age	Age'	obese	Pulse	Age * obese
347364.42	970946.68	14713067.97	79467.24	64967.11	64078.96

The following fit components were averaged over the 10 model fits:

```
stats linear.predictors
m5
```

Logistic Regression Model

```
fit.mult.impute(formula = diabetes ~ rcs(Age, 3) + obese + Pulse +
  Age %ia% obese, fitter = lrm, xtrans = nh_imp, data = nh1,
  x = TRUE, y = TRUE)
```

		Model Likelihood		Discrimination		Rank Discrim.	
		Ratio Test		Indexes		Indexes	
Obs	2757	LR chi2	217.63	R2	0.184	C	0.793
0	2550	d.f.	5	g	1.492	Dxy	0.586
1	207	Pr(> chi2)	<0.0001	gr	4.446	gamma	0.587
max deriv	7e-08			gp	0.080	tau-a	0.081
				Brier	0.063		

	Coef	S.E.	Wald Z	Pr(> Z)
Intercept	-10.0165	1.2339	-8.12	<0.0001
Age	0.1101	0.0281	3.92	<0.0001
Age'	-0.0276	0.0242	-1.14	0.2535
obese	1.9383	0.8248	2.35	0.0188
Pulse	0.0273	0.0062	4.38	<0.0001
Age * obese	-0.0136	0.0158	-0.86	0.3898

```
summary(m5)
```

Effects				Response : diabetes			
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95
Age	31	53	22	1.97810	0.320790	1.34940	2.60680
Odds Ratio	31	53	22	7.22910	NA	3.85500	13.55600
obese	0	1	1	1.35360	0.204650	0.95254	1.75470
Odds Ratio	0	1	1	3.87150	NA	2.59230	5.78200
Pulse	64	80	16	0.43608	0.099573	0.24092	0.63124
Odds Ratio	64	80	16	1.54660	NA	1.27240	1.87990

Adjusted to: Age=43 obese=0

Note that the only predictors included in the **Adjusted to:** section are those included as part of interactions.

If we want to see the results of adjusting the Age from 31 to 32 among non-obese subjects, or adjusting Pulse by just one beat per minute, we can do that...

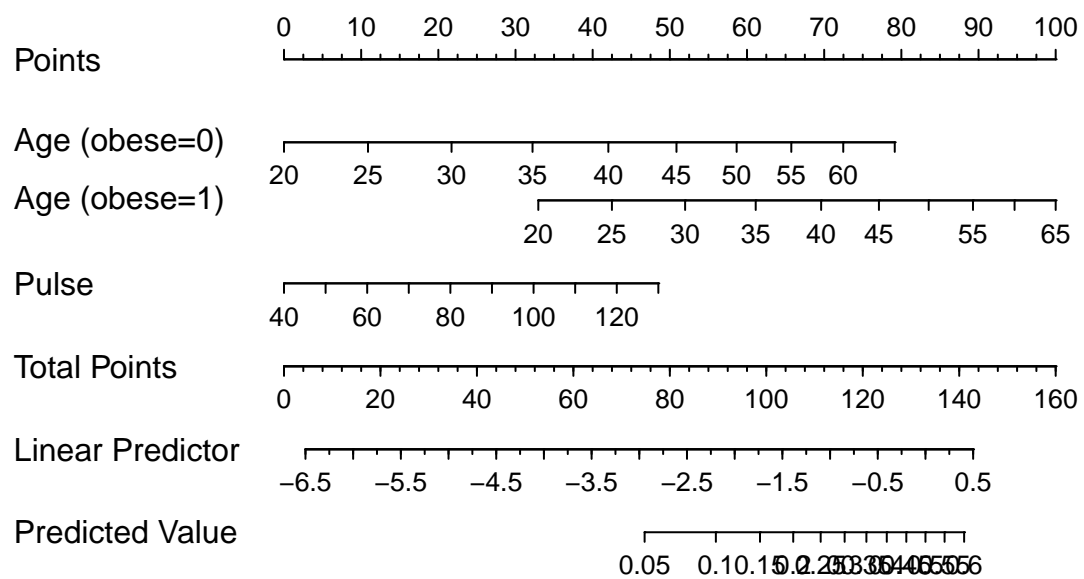
```
summary(m5, Age = c(31,32), obese = 0, Pulse = c(64,65))
```

Effects				Response : diabetes			
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95
Age	31	32	1	0.107200	0.0258460	0.056547	0.157860
Odds Ratio	31	32	1	1.113200	NA	1.058200	1.171000
obese	0	1	1	1.353600	0.2046500	0.952540	1.754700
Odds Ratio	0	1	1	3.871500	NA	2.592300	5.782000
Pulse	64	65	1	0.027255	0.0062233	0.015057	0.039452
Odds Ratio	64	65	1	1.027600	NA	1.015200	1.040200

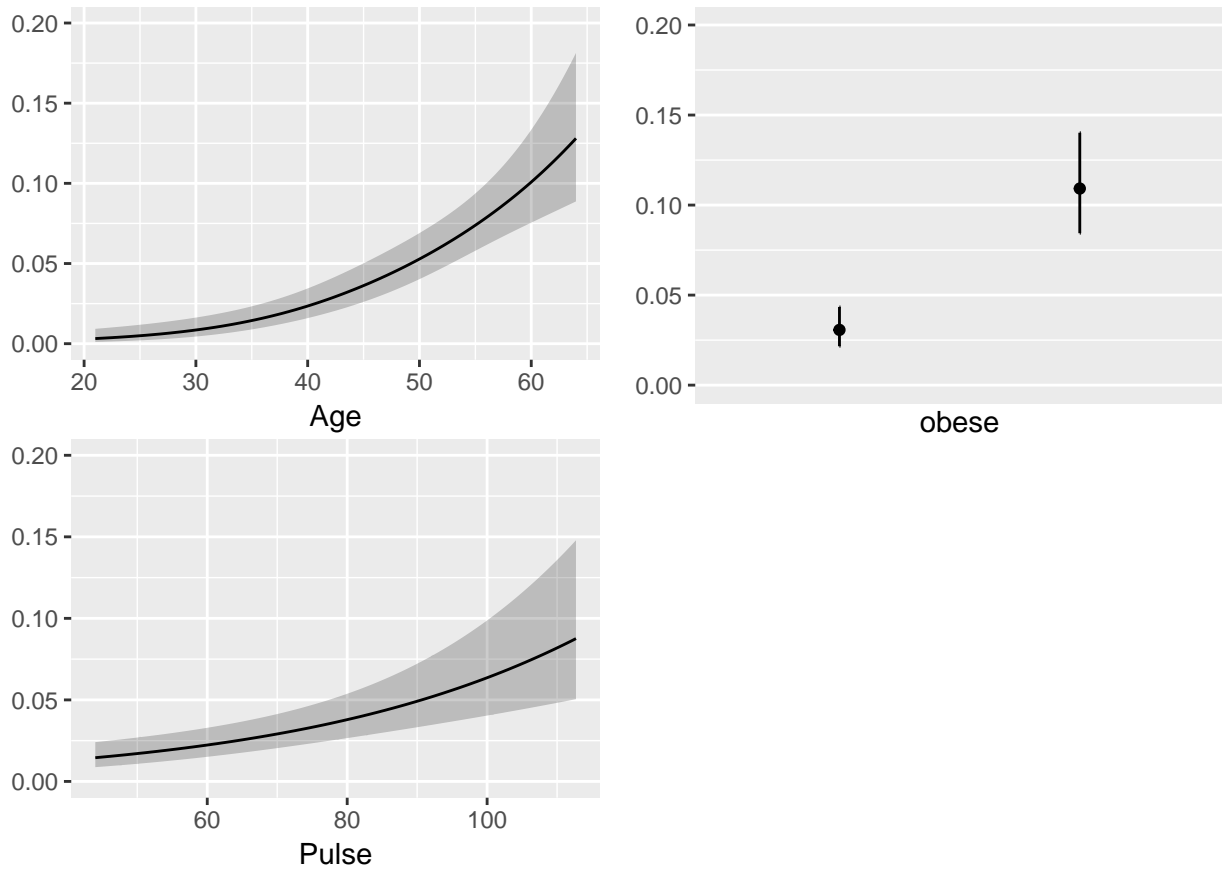
Adjusted to: Age=43 obese=0

5.4 Prediction Plot and Nomogram for Model m5

```
plot(nomogram(m5, fun = plogis))
```

```
ggplot(Predict(m5, fun = plogis))
```



It's hard to read the details of that nomogram. We better be sure we can make predictions using code directly...

5.5 Predicting Alice's probability of diabetes

Suppose Alice is 35 years old and has a Pulse of 100 beats per minute. To make a prediction for her using model `m5`, we'd again have to specify whether or not she is obese, or at least compare those two predicted probabilities. So what do we get?

```
predict(m5,
  newdata = tibble(names = c("Alice A", "Alice B"),
    Age = c(35, 35), obese = c(0, 1),
    Pulse = c(100, 100)),
  type = "fitted")
```

```
      1      2
0.03043615 0.11932894
```

I hope this is helpful.