

432 Class 8 Slides

github.com/THOMASELOVE/2019-432

2019-02-19

Setup

```
library(broom); library(janitor); library(Epi)
library(skimr); library(simputation); library(ROCR)
library(Hmisc); library(rms) # the key for lrm
library(tidyverse)
```

Today's Materials

- A new logistic regression example
 - Modeling 10-year risk of coronary heart disease
 - based on a sample from the Framingham Heart Study

```
fram <- read_csv("data/fram_new.csv") %>% clean_names()
```

Codebook (4,240 subjects, 17 variables)

Variable	Interpretation (at baseline)	NAs
subj	subject ID code	0
sex	F or M	0
age	in years	0
smoker	current smoker?	0
cigs_day	mean cigarettes smoked per day	29
bp_meds	on at least one BP medication?	53
hx_stroke	history of stroke?	0
hx_htn	history of hypertension?	0
hx_dm	history of diabetes?	0
educ	4 ordered levels (1-4)	105

- variables with ? in Interpretation are 1 = yes, 0 = no
- educ: 1 = some HS, 2 = HS diploma, 3 = some college, 4 = college grad

Codebook (4,240 subjects, 17 variables)

Variable	Interpretation	NAs
tot_chol	baseline total cholesterol, mg/dl	50
sbp	baseline mean systolic BP, mm Hg	0
dbp	baseline mean diastolic BP, mm Hg	0
bmi	baseline body mass index, kg/m ²	19
heart_r	baseline heart rate, beats/min	1
glucose	baseline glucose level, mg/dl	388
chd_10	CHD in 10 years after baseline?	0

- 1 Goal 1. Predict chd_10 using hx_htn
- 2 Goal 2. Predict chd_10 using tot_chol and hx_htn
- 3 Goal 3. Predict chd_10 using kitchen sink
- 4 Goal 4. Fit a smaller model almost as good as the KS.

Skimming the Data, before Cleanup or Imputation

```
> fram %>% select(-subj) %>% skim
```

```
Skim summary statistics
```

```
  n obs: 4240
```

```
  n variables: 16
```

```
-- Variable type:character -----
```

variable	missing	complete	n	min	max	empty	n_unique
sex	0	4240	4240	1	1	0	2

```
-- Variable type:numeric -----
```

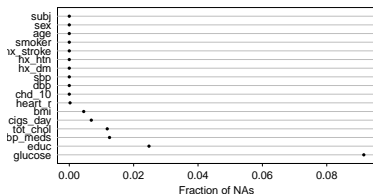
variable	missing	complete	n	mean	sd	p0	p25	p50	p75	p100	hist
age	0	4240	4240	49.58	8.57	32	42	49	56	70	
bmi	19	4221	4240	25.8	4.08	15.54	23.07	25.4	28.04	56.8	
bp_meds	53	4187	4240	0.03	0.17	0	0	0	0	1	
chd_10	0	4240	4240	0.15	0.36	0	0	0	0	1	
cigs_day	29	4211	4240	9.01	11.92	0	0	0	20	70	
dbp	0	4240	4240	82.9	11.91	48	75	82	90	142.5	
educ	105	4135	4240	1.98	1.02	1	1	2	3	4	
glucose	388	3852	4240	81.96	23.95	40	71	78	87	394	
heart_r	1	4239	4240	75.88	12.03	44	68	75	83	143	
hx_dm	0	4240	4240	0.026	0.16	0	0	0	0	1	
hx_htn	0	4240	4240	0.31	0.46	0	0	0	1	1	
hx_stroke	0	4240	4240	0.0059	0.077	0	0	0	0	1	
sbp	0	4240	4240	132.35	22.03	83.5	117	128	144	295	
smoker	0	4240	4240	0.49	0.5	0	0	0	1	1	
tot_chol	50	4190	4240	236.7	44.59	107	206	234	263	696	

Plotting Missingness (with Hmisc, result on next slide)

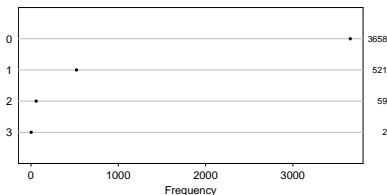
```
par(mfrow = c(2,2))  
naplot(naclus(fram))  
par(mfrow = c(1,1))
```

Plotting Missingness (with Hmisc)

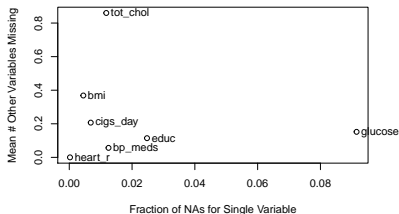
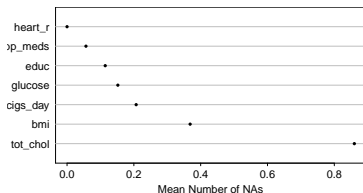
Fraction of NAs in each Variable



Number of Missing Variables Per Observation



Mean Number of Other Variables Missing for Observations where Indicated Variable is NA



Simple Imputation into fram1

```
set.seed(432001)

fram1 <- fram %>%
  impute_pmm(educ + cigs_day + heart_r ~
             age + smoker) %>%
  impute_rlm(bmi + tot_chol ~
            sex + age + sbp + heart_r) %>%
  impute_pmm(bp_meds ~ hx_htn + bmi + tot_chol) %>%
  impute_rlm(glucose ~ hx_dm + bmi + tot_chol + age)
```

Turn educ into ed_f, a factor.

```
fram1 <- fram1 %>%  
  mutate(ed_f = fct_recode(factor(educ),  
    "1_Some_HS" = "1", "2_HS_grad" = "2",  
    "3_Some_Col" = "3", "4_Col_grad" = "4"))  
  
fram1 %>% tabyl(ed_f, educ)
```

ed_f	1	2	3	4
1_Some_HS	1720	0	0	0
2_HS_grad	0	1358	0	0
3_Some_Col	0	0	689	0
4_Col_grad	0	0	0	473

Final Data Set?

```
fram2 <- fram1 %>%  
  select(subj, sex, age, smoker, cigs_day, bp_meds,  
         hx_stroke, hx_htn, hx_dm, ed_f, tot_chol,  
         sbp, dbp, bmi, heart_r, glucose, chd_10)
```

```
fram2 %>% select(-subj) %>% skim
```

```
> fram2 %>% select(-subj) %>% skim
```

```
Skim summary statistics
```

```
  n obs: 4240
```

```
  n variables: 16
```

```
-- Variable type:character -----
```

variable	missing	complete	n	min	max	empty	n_unique
sex	0	4240	4240	1	1	0	2

```
-- Variable type:factor -----
```

variable	missing	complete	n	n_unique	top_counts	ordered
ed_f	0	4240	4240	4	1_S: 1720, 2_H: 1358, 3_S: 689, 4_C: 473	FALSE

```
-- Variable type:numeric -----
```

variable	missing	complete	n	mean	sd	p0	p25	p50	p75	p100	hist
age	0	4240	4240	49.58	8.57	32	42	49	56	70	
bmi	0	4240	4240	25.8	4.07	15.54	23.08	25.38	28.04	56.8	
bp_meds	0	4240	4240	0.029	0.17	0	0	0	0	1	
chd_10	0	4240	4240	0.15	0.36	0	0	0	0	1	
cigs_day	0	4240	4240	9.07	11.91	0	0	0	20	70	
dbp	0	4240	4240	82.9	11.91	48	75	82	90	142.5	
glucose	0	4240	4240	81.7	22.94	40	72	78	85	394	
heart_r	0	4240	4240	75.88	12.02	44	68	75	83	143	
hx_dm	0	4240	4240	0.026	0.16	0	0	0	0	1	
hx_htn	0	4240	4240	0.31	0.46	0	0	0	1	1	
hx_stroke	0	4240	4240	0.0059	0.077	0	0	0	0	1	
sbp	0	4240	4240	132.35	22.03	83.5	117	128	144	295	
smoker	0	4240	4240	0.49	0.5	0	0	0	1	1	
tot_cho1	0	4240	4240	236.73	44.36	107	206	234	263	696	

Goal 1. Predict chd_10 using hx_htn

Predict chd_10 using hx_htn

```
fram2 %>%  
  tabyl(hx_htn, chd_10) %>%  
  adorn_percentages() %>%  
  adorn_pct_formatting() %>%  
  adorn_ns(position = "front") %>%  
  adorn_title()
```

	chd_10	
hx_htn	0	1
0	2604 (89.1%)	319 (10.9%)
1	992 (75.3%)	325 (24.7%)

Convert to Standard Epidemiological Format

```
fram2 <- fram2 %>%  
  mutate(htn_1 = fct_recode(factor(hx_htn),  
                             HTN = "1", NoHTN = "0"),  
         htn_1 = fct_relevel(htn_1, "HTN"),  
         out_1 = fct_recode(factor(chd_10),  
                             CHD = "1", NoCHD = "0"),  
         out_1 = fct_relevel(out_1, "CHD"))
```

A mosaic plot?

```
plot(table(fram2$out_1, fram2$htn_1))
```



Two-by-Two Table Analysis (from the Epi package)

```
twoby2(table(fram2$htn_1, fram2$out_1))
```

2 by 2 table analysis:

Outcome : CHD

Comparing : HTN vs. NoHTN

	CHD	NoCHD	P(CHD)	95% conf. interval	
HTN	325	992	0.2468	0.2242	0.2708
NoHTN	319	2604	0.1091	0.0983	0.1210

	95% conf. interval		
Relative Risk:	2.2612	1.9656	2.6013
Sample Odds Ratio:	2.6744	2.2542	3.1728
Conditional MLE Odds Ratio:	2.6737	2.2454	3.1843
Probability difference:	0.1376	0.1122	0.1640

A Logistic Regression model with glm

```
m_01 <- glm(chd_10 ~ hx_htn, data = fram2,  
            family = binomial)
```

```
m_01
```

Call: `glm(formula = chd_10 ~ hx_htn, family = binomial, data`

Coefficients:

(Intercept)	hx_htn
-2.0996	0.9837

Degrees of Freedom: 4239 Total (i.e. Null); 4238 Residual

Null Deviance: 3612

Residual Deviance: 3487 AIC: 3491

Interpretation of the Model

```
exp(coef(m_01)); exp(confint(m_01))
```

(Intercept)	hx_htn
0.1225038	2.6743730

Waiting for profiling to be done...

	2.5 %	97.5 %
(Intercept)	0.1088612	0.1373705
hx_htn	2.2543282	3.1733406

Compare this to the twoby2 result:

Sample Odds Ratio:	2.6744	2.2542	3.1728
Conditional MLE Odds Ratio:	2.6737	2.2454	3.1843

Using broom

```
tidy(m_01)
```

```
# A tibble: 2 x 5
```

	term	estimate	std.error	statistic	p.value
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	(Intercept)	-2.10	0.0593	-35.4	1.97e-274
2	hx_htn	0.984	0.0872	11.3	1.62e- 29

```
glance(m_01)
```

```
# A tibble: 1 x 7
```

	null.deviance	df.null	logLik	AIC	BIC	deviance
	<dbl>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1	3612.	4239	-1743.	3491.	3504.	3487.

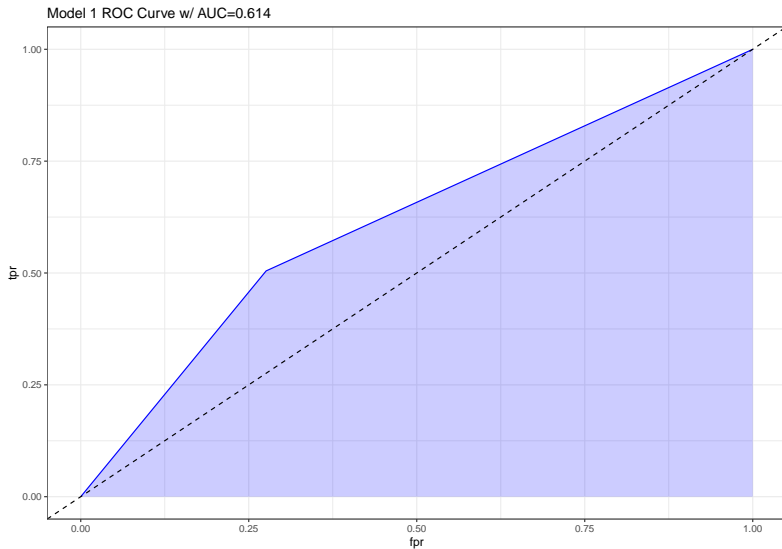
```
# ... with 1 more variable: df.residual <int>
```

Building the ROC Curve with ROCR

```
# requires ROCR package
prob <- predict(m_01, fram2, type = "response")
pred <- prediction(prob, fram2$chd_10)
perf <- performance(pred, measure = "tpr",
                    x.measure = "fpr")

auc <- performance(pred, measure = "auc")
auc <- round(auc@y.values[[1]], 3)
roc.dat <- data.frame(fpr = unlist(perf@x.values),
                     tpr = unlist(perf@y.values),
                     model = "GLM")
```

ROC Curve for our Model



C statistic isn't a “one stop” measure of accuracy

The C statistic tells you about *discrimination* but nothing about *calibration*.

- The poor C statistic indicates that m_{01} has poor discrimination.
 - If m_{01} predicts Harry has a higher $\Pr(\text{CHD})$ than Sally, we cannot really trust that will be an accurate ordering.
- But this isn't any indication of m_{01} 's calibration.
 - Even a large C statistic (near 1) doesn't tell you anything about whether a group of people with $\Pr(\text{CHD}) = 0.20$ would actually have anything close to a 20% chance of CHD.
 - A large C statistic indicates that the model puts subjects in the correct order (low risk of CHD to high risk,) but we can still get the actual risks wrong if the calibration is poor.

Using `lrm` from the `rms` package to fit Logistic Regression Models

A Logistic Regression model with lrm

```
d <- datadist(fram2)
options(datadist = "d")
```

```
m_01_lrm <- lrm(chd_10 ~ hx_htn, data = fram2, x = T, y = T)
```

```
> m_01_lrm
```

Logistic Regression Model

```
lrm(formula = chd_10 ~ hx_htn, data = fram2, x = T, y = T)
```

		Model Likelihood		Discrimination		Rank Discrim.	
		Ratio Test		Indexes		Indexes	
Obs	4240	LR chi2	125.30	R2	0.051	C	0.614
0	3596	d.f.	1	g	0.421	Dxy	0.229
1	644	Pr(> chi2)	<0.0001	gr	1.524	gamma	0.456
max deriv	1e-09			gp	0.059	tau-a	0.059
				Brier	0.125		

	Coef	S.E.	Wald Z	Pr(> Z)
Intercept	-2.0996	0.0593	-35.39	<0.0001
hx_htn	0.9837	0.0872	11.28	<0.0001

lrm output Piece by Piece

		Model Likelihood	
		Ratio Test	
Obs	4240	LR chi2	125.30
0	3596	d.f.	1
1	644	Pr(> chi2)	<0.0001
max deriv 1e-09			

- Likelihood-ratio test = drop in deviance test
 - How much does a goodness-of-fit statistic move as a result of this model?
 - Deviance = $-2 \log(\text{likelihood function})$

1rm output Piece by Piece, 2

Discrimination	Rank Discrim.
Indexes	Indexes
R2	C 0.614
g	Dxy 0.229
gr	gamma 0.456
gp	tau-a 0.059
Brier	0.125

Nagelkerke pseudo- R^2 statistic = 1 if the model predicts the outcome perfectly and the likelihood function is 1.

- an adjusted version (to a 0-1 scale) of the Cox-Snell pseudo- R^2
- compares the log likelihood of our model to the log likelihood for a null model.
- so it's similar to the R^2 for a linear model in terms of improvement from a null model to a fitted model
- neither a percentage of explained variability nor the square of any correlation

1rm output Piece by Piece, 3

Discrimination	Rank Discrim.
Indexes	Indexes
R2	C 0.614
g	Dxy 0.229
gr	gamma 0.456
gp	tau-a 0.059
Brier	0.125

- gp = Gini's index on the probability scale, which we want to be as large as possible
 - Gini's mean difference is the mean absolute difference between any two distinct predictions.
 - This measures the average “purity” in the predictions, essentially.
- R also presents g and gr, which are the same thing on the log odds, and odds scale.
- The **lower** the Brier score, the better the predictions are calibrated.
 - This is a nice measure of the accuracy of probabilistic predictions.

1rm output Piece by Piece, 4

Discrimination	Rank Discrim.
Indexes	Indexes
R2	C
g	Dxy
gr	gamma
gp	tau-a
Brier	

- C = C statistic = area under the ROC curve
- D_{xy} = Somers' d, and $C = 0.5 + D_{xy}/2$
- γ = Goodman and Kruskal's Γ , which is a measure of the rank correlation between the observed and predicted values of CHD = 1.
 - Values range from -1 (perfect negative association) to +1 (perfect agreement.)
- τ -a = Kendall's τ , is another measure of such an association.

Validating our Summary Statistics

```
validate(m_01_lrm)
```

	index.orig	training	test	optimism
Dxy	0.2288	0.2340	0.2288	0.0052
R2	0.0508	0.0532	0.0508	0.0025
Intercept	0.0000	0.0000	-0.0201	0.0201
Slope	1.0000	1.0000	0.9840	0.0160
E _{max}	0.0000	0.0000	0.0072	0.0072
D	0.0293	0.0307	0.0293	0.0014
U	-0.0005	-0.0005	0.0000	-0.0005
Q	0.0298	0.0312	0.0293	0.0019
B	0.1248	0.1240	0.1248	-0.0008
g	0.4214	0.4303	0.4214	0.0089
gp	0.0590	0.0600	0.0590	0.0011

	index.corrected	n
--	-----------------	---

Dxy	0.2236	40
R2	0.0483	40

Coefficients Summary from `m_01_lrm`

	Coef	S.E.	Wald Z	Pr(> Z)
Intercept	-2.0996	0.0593	-35.39	<0.0001
hx_htn	0.9837	0.0872	11.28	<0.0001

Conclusions?

Assessing Effect Sizes

```
summary(m_01_lrm)
```

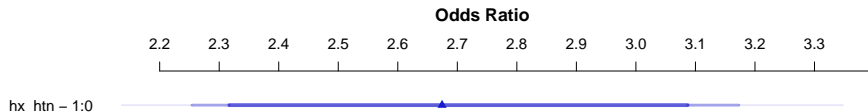
Effects

Response : chd_10

Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95
hx_htn	0	1	1	0.98371	0.0872	0.81281
Odds Ratio	0	1	1	2.67440	NA	2.25420
Upper 0.95						
1.1546						
3.1728						

Plotting the Effect Sizes

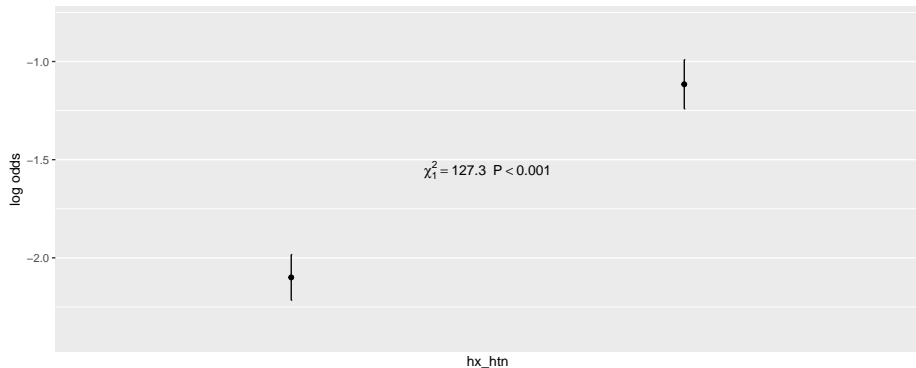
```
plot(summary(m_01_lrm))
```



The plot shows 90%, 95% and 99% confidence intervals.

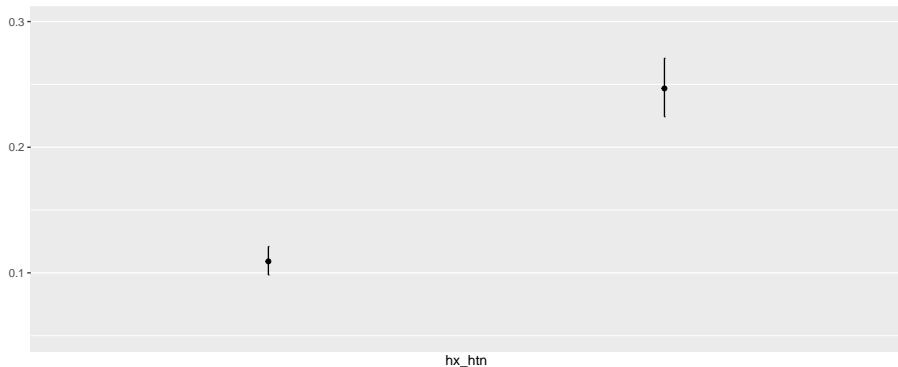
Can we see the prediction results?

```
ggplot(Predict(m_01_lrm),  
       anova = anova(m_01_lrm), pval = TRUE)
```



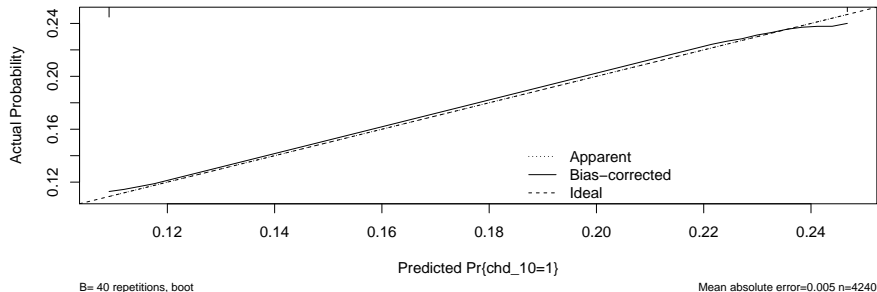
What about on a better scale?

```
ggplot(Predict(m_01_lrm, fun = plogis))
```



Is this m_01_lrm well calibrated?

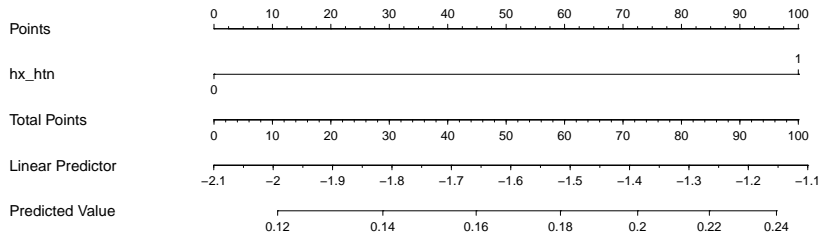
```
plot(calibrate(m_01_lrm))
```



n=4240 Mean absolute error=0.005 Mean squared error=2e-05
0.9 Quantile of absolute error=0.007

Nomogram for m_01_lrm

```
plot(nomogram(m_01_lrm, fun = plogis))
```



Goal 2. Predict chd_10 using hx_htn and tot_chol

glm fit (Don't forget family = binomial!)

```
m_02 <- glm(chd_10 ~ hx_htn + tot_chol,  
            data = fram2, family = binomial)
```

```
m_02
```

```
Call:  glm(formula = chd_10 ~ hx_htn + tot_chol, family = binomial,  
           data = fram2)
```

Coefficients:

(Intercept)	hx_htn	tot_chol
-2.855553	0.934387	0.003229

Degrees of Freedom: 4239 Total (i.e. Null); 4237 Residual

Null Deviance: 3612

Residual Deviance: 3475 AIC: 3481

Does m_02 improve on m_01 by ANOVA?

```
anova(m_01, m_02)
```

Analysis of Deviance Table

Model 1: chd_10 ~ hx_htn

Model 2: chd_10 ~ hx_htn + tot_chol

	Resid. Df	Resid. Dev	Df	Deviance
1	4238	3486.9		
2	4237	3475.5	1	11.411

```
pchisq(11.41, 1, lower.tail = FALSE)
```

```
[1] 0.0007304983
```


Does m_02 improve on m_01 by AIC/BIC?

```
glance(m_01)
```

```
# A tibble: 1 x 7
  null.deviance df.null logLik   AIC   BIC deviance
      <dbl>     <int>  <dbl> <dbl> <dbl>   <dbl>
1       3612.     4239 -1743. 3491. 3504.   3487.
# ... with 1 more variable: df.residual <int>
```

```
glance(m_02)
```

```
# A tibble: 1 x 7
  null.deviance df.null logLik   AIC   BIC deviance
      <dbl>     <int>  <dbl> <dbl> <dbl>   <dbl>
1       3612.     4239 -1738. 3481. 3501.   3475.
# ... with 1 more variable: df.residual <int>
```

`anova(m_02)`

Analysis of Deviance Table

Model: binomial, link: logit

Response: chd_10

Terms added sequentially (first to last)

	Df	Deviance	Resid. Df	Resid. Dev
NULL			4239	3612.2
hx_htn	1	125.302	4238	3486.9
tot_chol	1	11.411	4237	3475.5

summary(m_02)

```
> summary(m_02)

Call:
glm(formula = chd_10 ~ hx_htn + tot_chol, family = binomial,
    data = fram2)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3188  -0.5493  -0.4814  -0.4494   2.2276

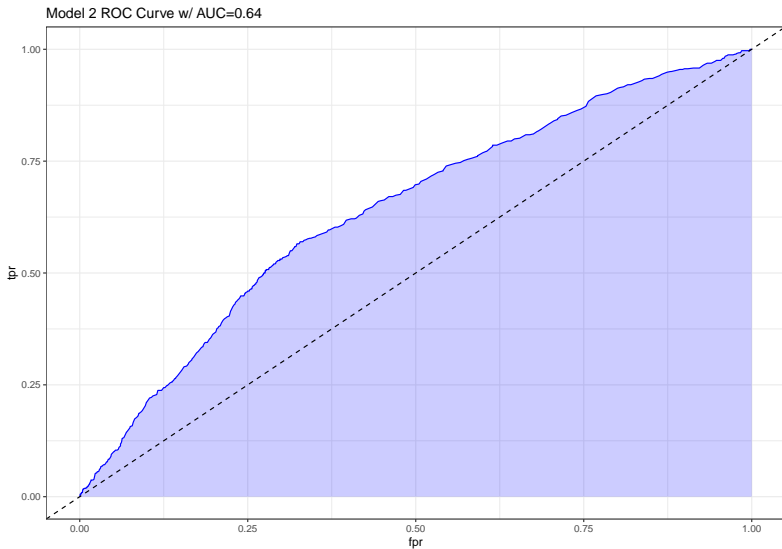
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.855553   0.232476 -12.283  < 2e-16 ***
hx_htn       0.934387   0.088454  10.564  < 2e-16 ***
tot_chol     0.003229   0.000951   3.396 0.000684 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3612.2  on 4239  degrees of freedom
Residual deviance: 3475.5  on 4237  degrees of freedom
AIC: 3481.5

Number of Fisher Scoring iterations: 4
```

ROC plot for m_02



Fitting with lrm

```
d <- datadist(fram2)
options(datadist = "d")
m_02_lrm <- lrm(chd_10 ~ hx_htn + tot_chol, data = fram2,
                x = TRUE, y = TRUE)
```

```
> m_02_lrm
```

Logistic Regression Model

```
lrm(formula = chd_10 ~ hx_htn + tot_chol, data = fram2, x = TRUE,
     y = TRUE)
```

		Model Likelihood Ratio Test		Discrimination Indexes		Rank Discrim. Indexes	
Obs	4240	LR chi2	136.71	R2	0.055	C	0.640
0	3596	d.f.	2	g	0.510	Dxy	0.281
1	644	Pr(> chi2)	<0.0001	gr	1.665	gamma	0.282
max deriv	4e-07			gp	0.069	tau-a	0.072
				Brier	0.125		

	Coef	S.E.	Wald Z	Pr(> Z)
Intercept	-2.8556	0.2325	-12.28	<0.0001
hx_htn	0.9344	0.0885	10.56	<0.0001
tot_chol	0.0032	0.0010	3.40	0.0007

Validating our Summary Statistics

```
validate(m_02_1rm)
```

	index.orig	training	test	optimism
Dxy	0.2805	0.2766	0.2806	-0.0040
R2	0.0553	0.0550	0.0548	0.0001
Intercept	0.0000	0.0000	0.0061	-0.0061
Slope	1.0000	1.0000	1.0085	-0.0085
E _{max}	0.0000	0.0000	0.0029	0.0029
D	0.0320	0.0319	0.0317	0.0002
U	-0.0005	-0.0005	-0.0001	-0.0004
Q	0.0325	0.0324	0.0318	0.0005
B	0.1245	0.1252	0.1246	0.0006
g	0.5098	0.4996	0.5018	-0.0022
gp	0.0690	0.0681	0.0681	0.0001

	index.corrected	n
Dxy	0.2845	40
R2	0.0552	40

ANOVA with lrm

```
anova(m_02_lrm)
```

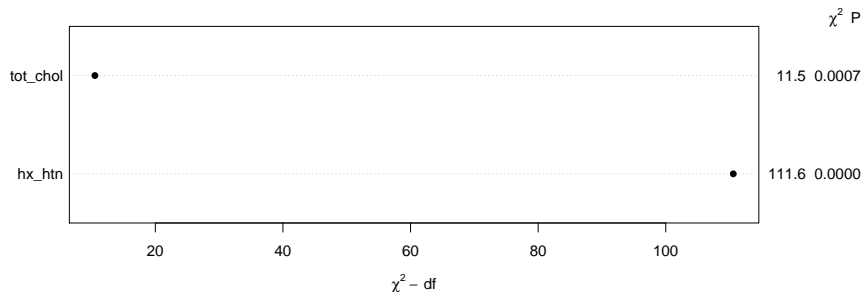
Wald Statistics

Response: chd_10

Factor	Chi-Square	d.f.	P
hx_htn	111.59	1	<.0001
tot_chol	11.53	1	7e-04
TOTAL	137.85	2	<.0001

ANOVA plot in lrm

```
plot(anova(m_02_lrm))
```



Estimated Effect Sizes

```
summary(m_02_lrm)
```

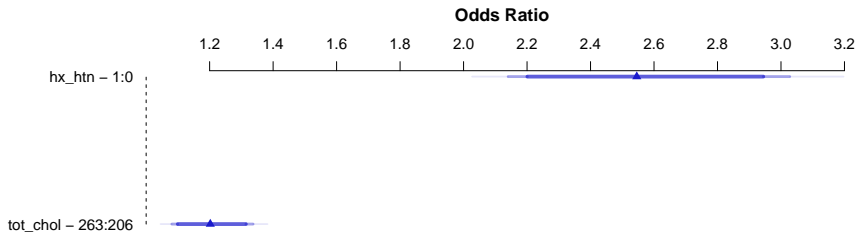
Effects

Response : chd_10

Factor	Low	High	Diff.	Effect	S.E.	Lower	0.95
hx_htn	0	1	1	0.93439	0.088455	0.761020	
Odds Ratio	0	1	1	2.54570		NA	2.140500
tot_chol	206	263	57	0.18407	0.054208	0.077827	
Odds Ratio	206	263	57	1.20210		NA	1.080900
Upper	0.95						
	1.10780						
	3.02760						
	0.29032						
	1.33690						

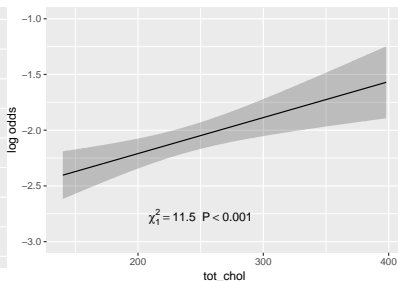
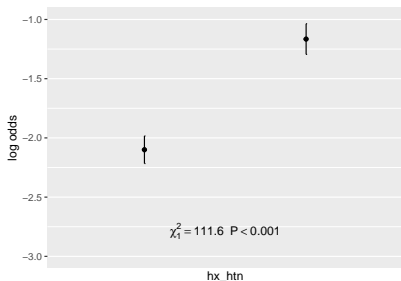
Plotting the Effect Sizes

```
plot(summary(m_02_lrm))
```



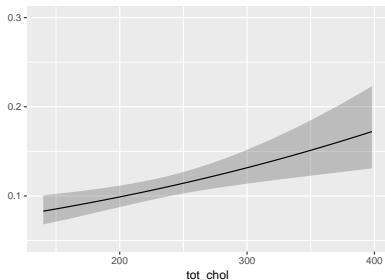
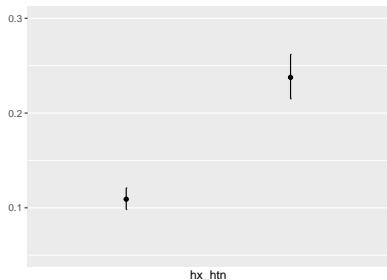
Can we see the prediction results?

```
ggplot(Predict(m_02_lrm),  
       anova = anova(m_02_lrm), pval = TRUE)
```



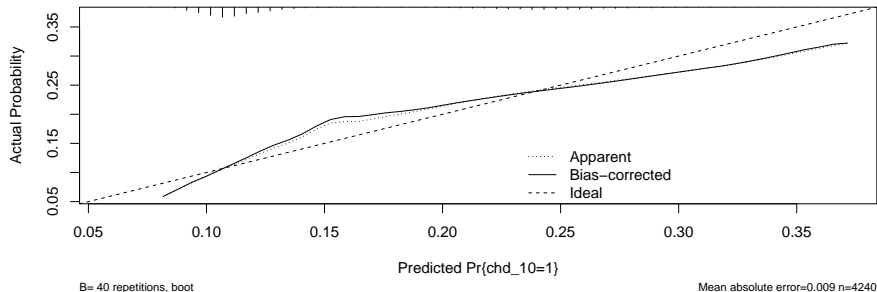
What about on a better scale?

```
ggplot(Predict(m_02_lrm, fun = plogis))
```



Calibration of mod_02_lrm

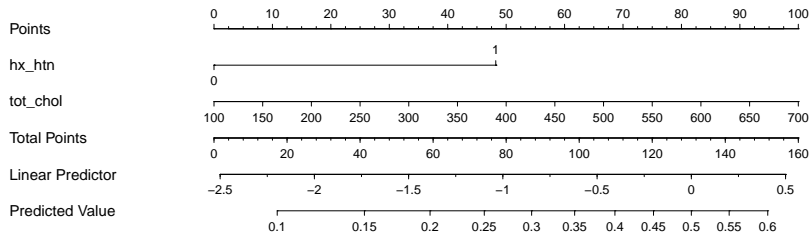
```
plot(calibrate(m_02_lrm))
```



n=4240 Mean absolute error=0.009 Mean squared error=0.0001
0.9 Quantile of absolute error=0.018

Nomogram of mod_02_lrm

```
plot(nomogram(m_02_lrm, fun = plogis))
```



Goal 3. Kitchen Sink Model

Focus on model with lrm first!

```
m_03 <- glm(chd_10 ~ hx_htn + tot_chol + sex + age +  
            smoker + cigs_day + bp_meds +  
            hx_stroke + hx_dm + ed_f + sbp + dbp +  
            bmi + heart_r + glucose,  
            data = fram2, family = binomial)
```

```
d <- datadist(fram2)  
options(datadist = "d")  
m_03_lrm <- lrm(chd_10 ~ hx_htn + tot_chol + sex + age +  
               smoker + cigs_day + bp_meds +  
               hx_stroke + hx_dm + ed_f + sbp + dbp +  
               bmi + heart_r + glucose,  
               data = fram2, x = TRUE, y = TRUE)
```


m_03_lrm (first section of output)

```
> m_03_lrm
```

```
Logistic Regression Model
```

```
lm(formula = chd_10 ~ hx_htn + tot_chol + sex + age + smoker +  
  cigs_day + bp_meds + hx_stroke + hx_dm + ed_f + sbp + dbp +  
  bmi + heart_r + glucose, data = fram2, x = TRUE, y = TRUE)
```

		Model Likelihood		Discrimination		Rank Discrim.	
		Ratio Test		Indexes		Indexes	
Obs	4240	LR chi2	405.40	R2	0.159	C	0.733
0	3596	d.f.	17	g	1.016	Dxy	0.466
1	644	Pr(> chi2)	<0.0001	gr	2.763	gamma	0.466
max deriv	6e-10			gp	0.120	tau-a	0.120
				Brier	0.115		

m_03_lrm (second section of output)

	Coef	S.E.	Wald Z	Pr(> Z)
Intercept	-7.9981	0.6583	-12.15	<0.0001
hx_htn	0.2331	0.1287	1.81	0.0700
tot_chol	0.0018	0.0010	1.73	0.0842
sex=M	0.4886	0.1012	4.83	<0.0001
age	0.0607	0.0063	9.67	<0.0001
smoker	0.0248	0.1451	0.17	0.8642
cigs_day	0.0207	0.0057	3.60	0.0003
bp_meds	0.2534	0.2206	1.15	0.2506
hx_stroke	0.9633	0.4439	2.17	0.0300
hx_dm	0.1353	0.2989	0.45	0.6507
ed_f=2_HS_grad	-0.1906	0.1120	-1.70	0.0889
ed_f=3_Some_Col	-0.1005	0.1397	-0.72	0.4719
ed_f=4_Col_grad	0.0255	0.1533	0.17	0.8679
sbp	0.0141	0.0035	3.98	<0.0001
dbp	-0.0029	0.0060	-0.48	0.6294
bmi	0.0019	0.0118	0.16	0.8712
heart_r	-0.0012	0.0039	-0.32	0.7524
glucose	0.0071	0.0022	3.28	0.0010

Validating our Summary Statistics

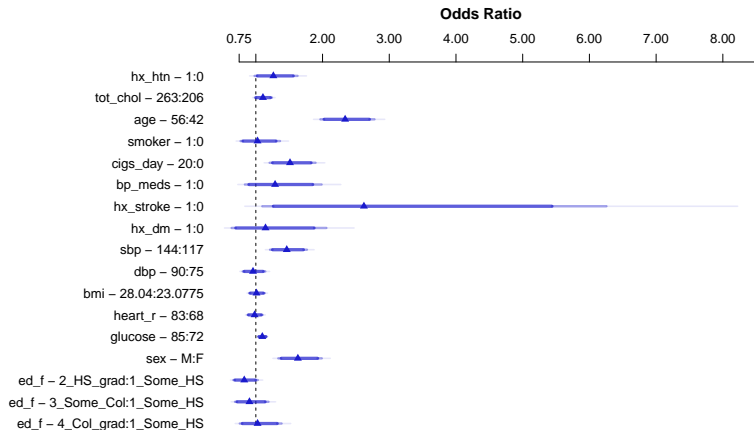
```
validate(m_03_lrm)
```

	index.orig	training	test	optimism
Dxy	0.4659	0.4813	0.4587	0.0226
R2	0.1590	0.1701	0.1532	0.0169
Intercept	0.0000	0.0000	-0.0838	0.0838
Slope	1.0000	1.0000	0.9425	0.0575
E _{max}	0.0000	0.0000	0.0287	0.0287
D	0.0954	0.1025	0.0917	0.0107
U	-0.0005	-0.0005	0.0003	-0.0008
Q	0.0958	0.1029	0.0914	0.0115
B	0.1152	0.1142	0.1159	-0.0017
g	1.0164	1.0550	0.9888	0.0662
gp	0.1203	0.1242	0.1179	0.0063

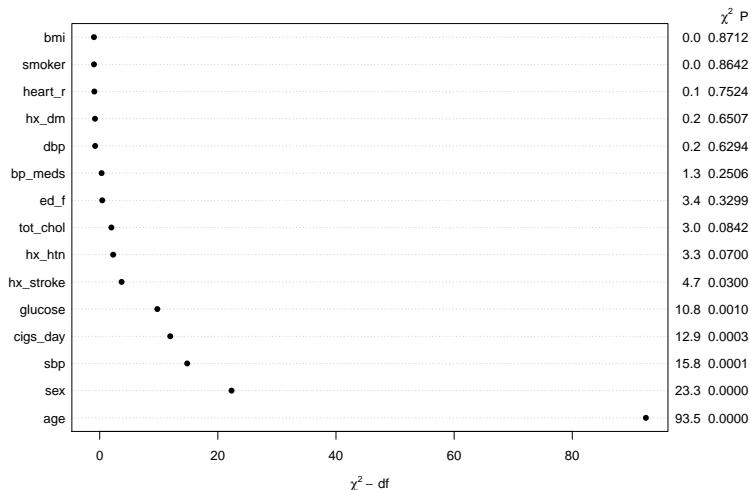
	index.corrected	n
--	-----------------	---

Dxy	0.4433	40
R2	0.1422	40

```
plot(summary(m_03_lrm))
```

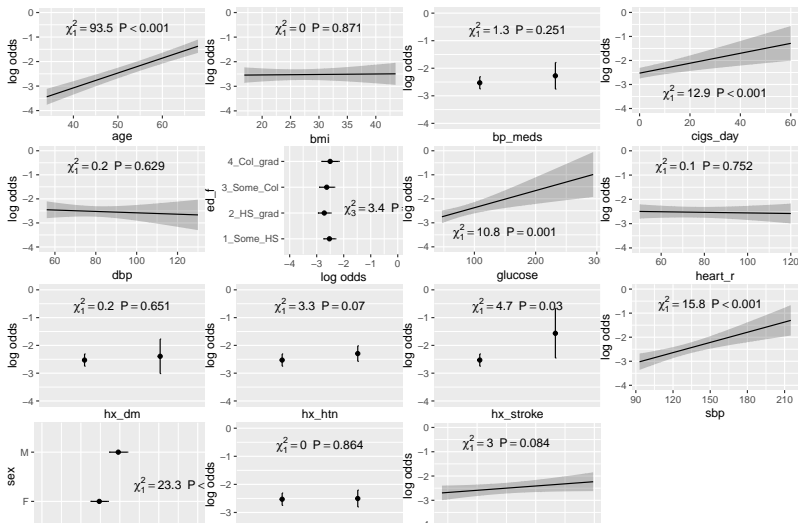


```
plot(anova(m_03_lrm))
```



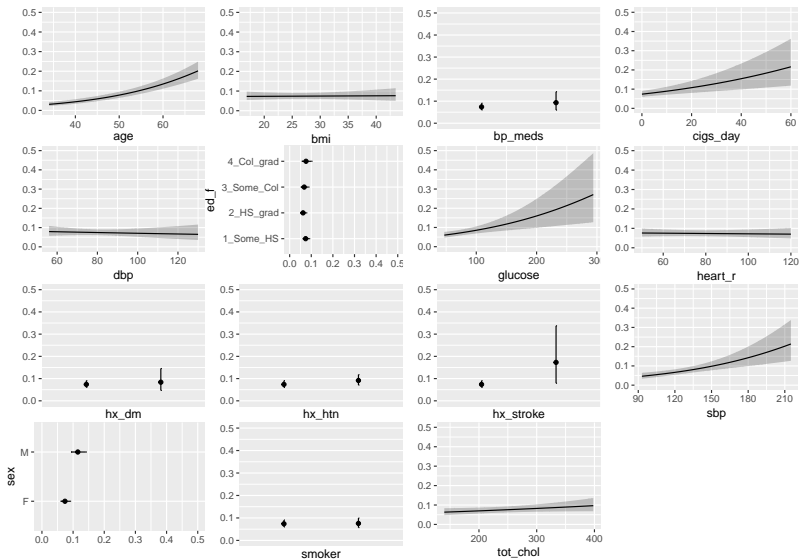
Can we see the prediction results?

```
ggplot(Predict(m_03_lrm),  
  anova = anova(m_03_lrm), pval = TRUE)
```



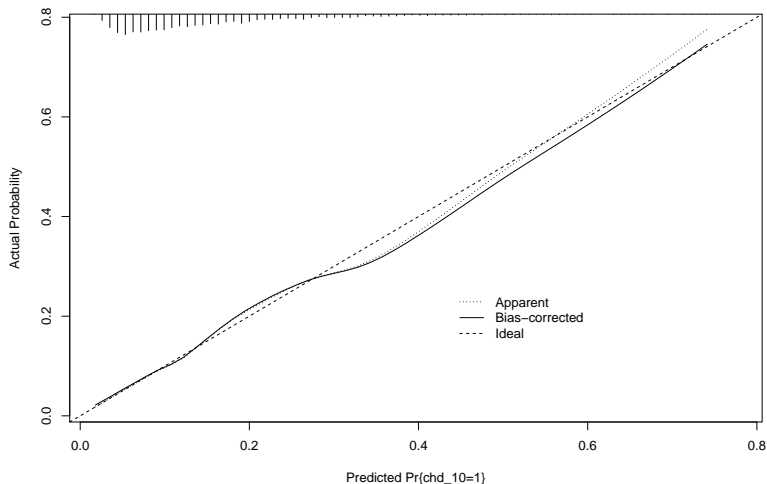
What about on a better scale?

```
ggplot(Predict(m_03_lrm, fun = plogis))
```



Calibration of mod_03_1rm

```
plot(calibrate(m_03_1rm))
```

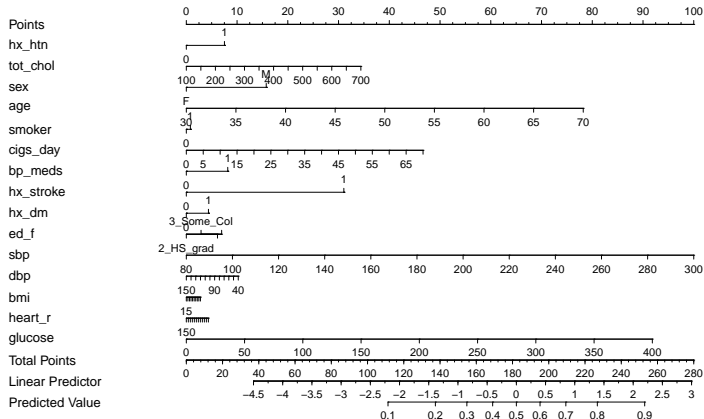


B= 40 repetitions, boot

Mean absolute error=0.008 n=4240

Nomogram of mod_03_lrm

```
plot(nomogram(m_03_lrm, fun = plogis))
```



Comparing our Three Nested Models

```
anova(m_01, m_02, m_03)
```

Analysis of Deviance Table

Model 1: chd_10 ~ hx_htn

Model 2: chd_10 ~ hx_htn + tot_chol

Model 3: chd_10 ~ hx_htn + tot_chol + sex + age + smoker + cig
bp_meds + hx_stroke + hx_dm + ed_f + sbp + dbp + bmi + hea
glucose

	Resid. Df	Resid. Dev	Df	Deviance
1	4238	3486.9		
2	4237	3475.5	1	11.411
3	4222	3206.8	15	268.682

Model 2 vs. Model 3 at a glance

```
glance(m_02)
```

```
# A tibble: 1 x 7
  null.deviance df.null logLik   AIC   BIC deviance
      <dbl>     <int>  <dbl> <dbl> <dbl>   <dbl>
1      3612.     4239 -1738. 3481. 3501.   3475.
# ... with 1 more variable: df.residual <int>
```

```
glance(m_03)
```

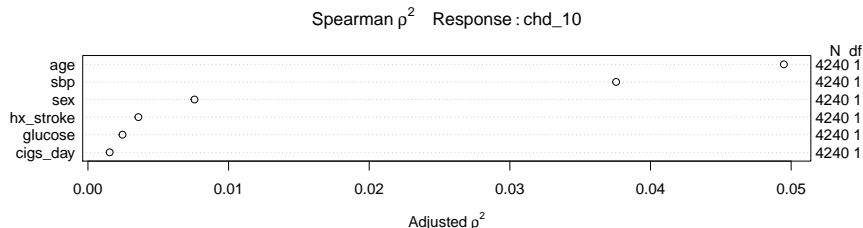
```
# A tibble: 1 x 7
  null.deviance df.null logLik   AIC   BIC deviance
      <dbl>     <int>  <dbl> <dbl> <dbl>   <dbl>
1      3612.     4239 -1603. 3243. 3357.   3207.
# ... with 1 more variable: df.residual <int>
```

Fitting a 6-predictor, but still useful model

What looks useful?

By ANOVA on `m_03_1rm` it looks like `age`, `sex`, `sbp`, `cigs_day`, `glucose`, `hx_stroke` for sure.

```
plot(spearman2(chd_10 ~ age + sex + sbp + cigs_day +  
              glucose + hx_stroke, data = fram2))
```



New Model 4

```
m_04 <- glm(chd_10 ~ rcs(age, 5) + rcs(sbp, 3) + sex +  
            hx_stroke + glucose + cigs_day,  
            data = fram2, family = binomial)  
  
dd <- datadist(fram2)  
options(datadist = "dd")  
  
m_04_lrm <- lrm(chd_10 ~ rcs(age, 5) + rcs(sbp, 3) + sex +  
               hx_stroke + glucose + cigs_day,  
               data = fram2, x = TRUE, y = TRUE)
```

m_04_lrm

```
> m_04_lrm
```

```
Logistic Regression Model
```

```
lrm(formula = chd_10 ~ rcs(age, 5) + rcs(sbp, 3) + sex + hx_stroke +  
      glucose + cigs_day, data = fram2, x = TRUE, y = TRUE)
```

		Model Likelihood Ratio Test	Discrimination Indexes	Rank Discrim. Indexes
Obs	4240	LR chi2 401.44	R2 0.158	C 0.731
0	3596	d.f. 10	g 1.041	Dxy 0.461
1	644	Pr(> chi2) <0.0001	gr 2.833	gamma 0.461
max deriv	2e-06		gp 0.120	tau-a 0.119
			Brier 0.115	

	Coef	S.E.	Wald Z	Pr(> Z)
Intercept	-8.7201	2.4092	-3.62	0.0003
age	0.0732	0.0576	1.27	0.2037
age'	0.2871	0.3539	0.81	0.4172
age''	-1.1057	0.9608	-1.15	0.2498
age'''	1.4723	0.9640	1.53	0.1267
sbp	0.0147	0.0066	2.23	0.0256
sbp'	0.0030	0.0080	0.37	0.7080
sex=M	0.4935	0.0973	5.07	<0.0001
hx_stroke	1.0514	0.4345	2.42	0.0155
glucose	0.0076	0.0016	4.69	<0.0001
cigs_day	0.0208	0.0039	5.39	<0.0001

Validating our Summary Statistics

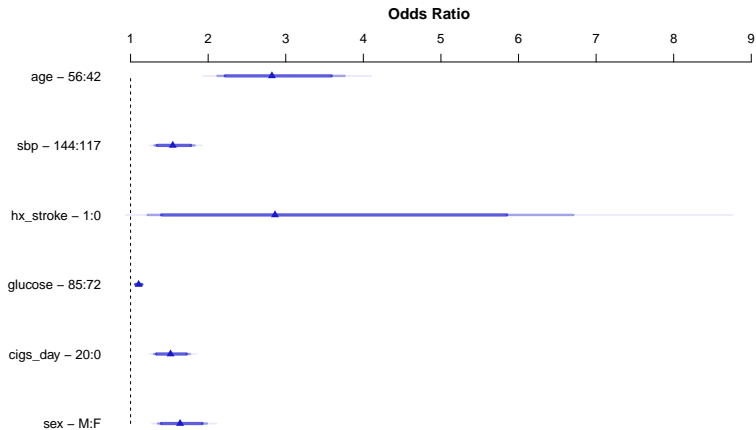
```
validate(m_04_lrm)
```

	index.orig	training	test	optimism
Dxy	0.4611	0.4585	0.4566	0.0019
R2	0.1575	0.1593	0.1540	0.0052
Intercept	0.0000	0.0000	-0.0321	0.0321
Slope	1.0000	1.0000	0.9851	0.0149
E _{max}	0.0000	0.0000	0.0095	0.0095
D	0.0944	0.0959	0.0922	0.0037
U	-0.0005	-0.0005	0.0000	-0.0005
Q	0.0949	0.0964	0.0922	0.0042
B	0.1152	0.1157	0.1156	0.0001
g	1.0414	1.0450	1.0237	0.0212
gp	0.1198	0.1206	0.1181	0.0025

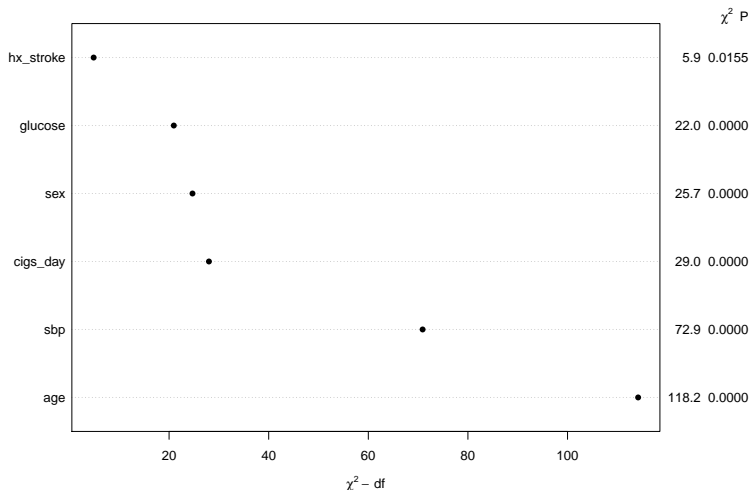
	index.corrected	n
--	-----------------	---

Dxy	0.4592	40
R2	0.1523	40


```
plot(summary(m_04_lrm))
```

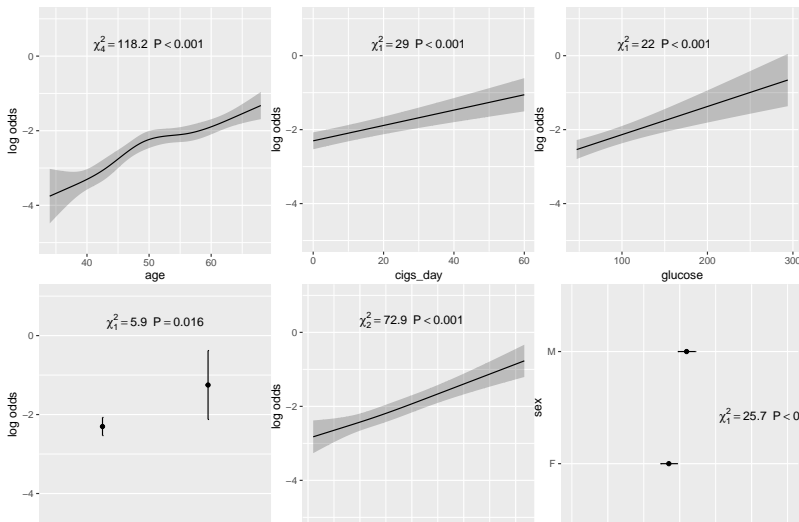


```
plot(anova(m_04_lrm))
```



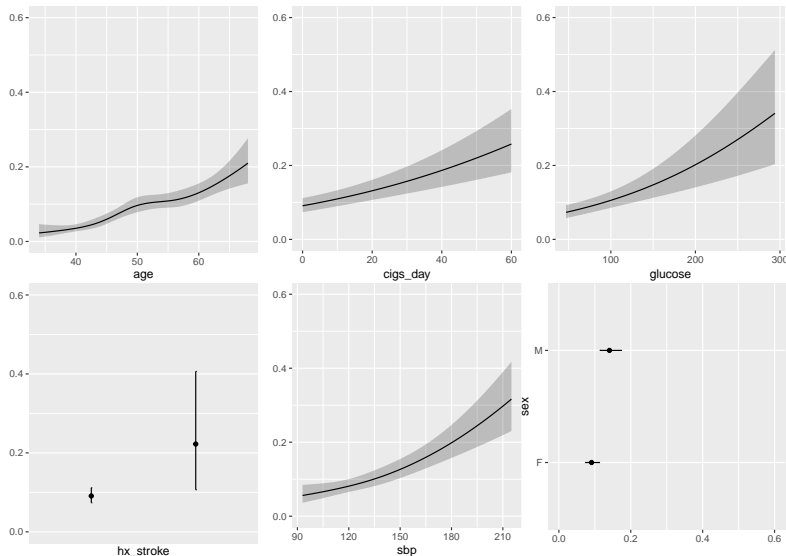
Can we see the prediction results?

```
ggplot(Predict(m_04_lrm),  
       anova = anova(m_04_lrm), pval = TRUE)
```



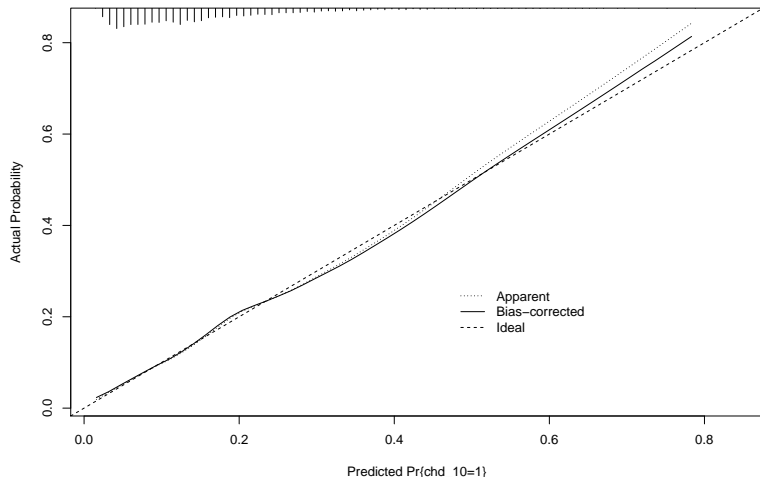
What about on a better scale?

```
ggplot(Predict(m_04_lrm, fun = plogis))
```



Calibration of mod_04_lrm

```
plot(calibrate(m_04_lrm))
```

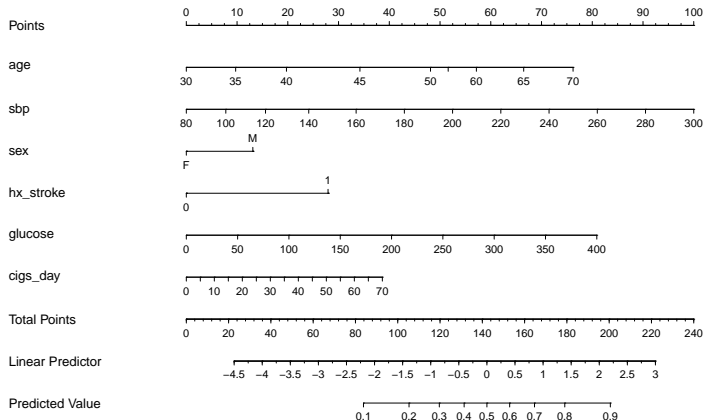


B= 40 repetitions, boot

Mean absolute error=0.005 n=4240

Nomogram of mod_04_lrm

```
plot(nomogram(m_04_lrm, fun = plogis))
```



Comparing Models 3 and 4 (which aren't nested)

```
glance(m_03) # kitchen sink but no non-linear terms
```

```
# A tibble: 1 x 7
  null.deviance df.null logLik   AIC   BIC deviance
      <dbl>     <int>  <dbl> <dbl> <dbl>     <dbl>
1       3612.     4239 -1603. 3243. 3357.     3207.
# ... with 1 more variable: df.residual <int>
```

```
glance(m_04) # six predictors but with non-linear terms
```

```
# A tibble: 1 x 7
  null.deviance df.null logLik   AIC   BIC deviance
      <dbl>     <int>  <dbl> <dbl> <dbl>     <dbl>
1       3612.     4239 -1605. 3233. 3303.     3211.
# ... with 1 more variable: df.residual <int>
```

Checking Residuals?

- Yes/No outcomes contain less information than quantitative outcomes
- Residuals cannot be observed - predicted
 - There are several different types of residuals defined
- Assumptions of logistic regression are different
 - Model is deliberately non-linear
 - Error variance is a function of the mean, so it isn't constant
 - Errors aren't assumed to follow a Normal distribution
 - Only thing that's the same: leverage and influence

So, plot 5 (residuals/leverage/influence) can be a little useful, but that's it.

- We'll need better diagnostic tools for generalized linear models.

Any observations particularly influential on Model 4?

```
which.influence(m_04_lrm, cutoff = 0.3)
```

```
$Intercept
```

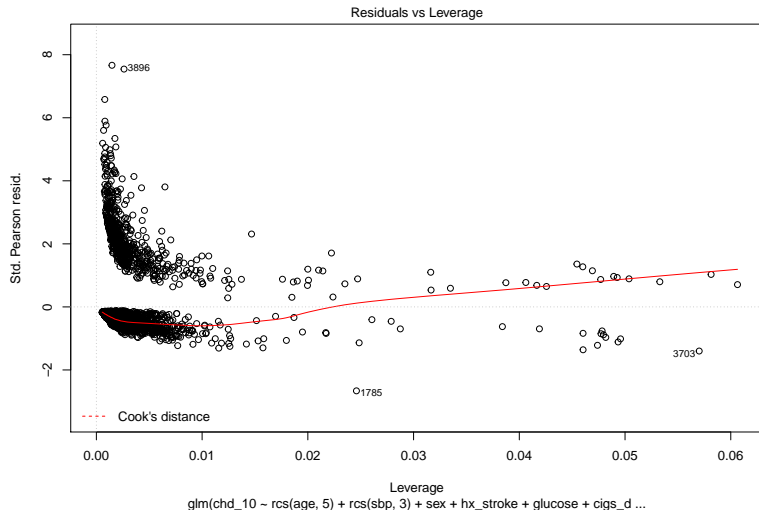
```
[1] "3896"
```

```
$glucose
```

```
[1] "1785" "3703"
```

Influence and Model 4?

```
plot(m_04, which = 5)
```



Next Time

Project 1 Discussions in Small Groups