432 Class 16

Thomas E. Love, Ph.D.

2024-03-07

## Today’s Agenda

* Can we fit a linear model to a count outcome?
* Selecting non-linear terms in light of Spearman
* Fitting a Poisson regression with the rms package
* Checking Assumptions in Logistic Regression Models

## Setup

knitr::opts\_chunk$set(comment=NA)  
options(width = 80)  
  
library(janitor); library(gt); library(broom)   
library(rsample); library(yardstick)  
library(car)  
library(countreg) ## for rootograms  
library(topmodels) ## for rootograms  
library(rms)  
library(tidyverse)  
  
theme\_set(theme\_bw())

# Could we fit a linear model for a count outcome? Revisiting Class 15

## The medicare data from Class 15

medicare <- read\_csv("c16/data/medicare.csv", show\_col\_types = FALSE) |>   
 mutate(across(where(is\_character), as\_factor),  
 subject = as.character(subject),   
 insurance = fct\_relevel(insurance, "no", "yes"),  
 logvisits = log(visits + 1)) ## needed because some have 0 visits  
  
set.seed(432)  
med\_split <- initial\_split(medicare, prop = 0.75)  
  
med\_train = training(med\_split)  
med\_test = testing(med\_split)

## The medicare data

medicare

# A tibble: 4,406 × 9  
 subject visits hospital health chronic sex school insurance logvisits  
 <chr> <dbl> <dbl> <fct> <dbl> <fct> <dbl> <fct> <dbl>  
 1 1 5 1 average 2 male 6 yes 1.79   
 2 2 1 0 average 2 female 10 yes 0.693  
 3 3 13 3 poor 4 female 10 no 2.64   
 4 4 16 1 poor 2 male 3 yes 2.83   
 5 5 3 0 average 2 female 6 yes 1.39   
 6 6 17 0 poor 5 female 7 no 2.89   
 7 7 9 0 average 0 female 8 yes 2.30   
 8 8 3 0 average 0 female 8 yes 1.39   
 9 9 1 0 average 0 female 8 yes 0.693  
10 10 0 0 average 0 female 8 yes 0   
# ℹ 4,396 more rows

## Reiterating the Goal

Predict visits using these 6 predictors…

| Predictor | Description |
| --- | --- |
| hospital | # of hospital stays |
| health | self-rated health (poor, average, excellent) |
| chronic | # of chronic conditions |
| sex | male or female |
| school | years of education |
| insurance | subject (also) has private insurance? (yes/no) |

## Linear Model for our Count Outcome

Let’s fit a **linear regression** (mod\_0: note *log* transformation) to go along with the Poisson regression (mod\_1) we fit last time.

mod\_0 <- lm(log(visits+1) ~ hospital + health + chronic + sex + school +   
 insurance, data = med\_train)  
  
mod\_1 <- glm(visits ~ hospital + health + chronic + sex + school +   
 insurance, data = med\_train, family = "poisson")

## Linear Model Coefficients?

## linear model  
tidy(mod\_0) |> gt() |> fmt\_number(decimals = 3) |>   
 tab\_options(table.font.size = 20)

| term | estimate | std.error | statistic | p.value |
| --- | --- | --- | --- | --- |
| (Intercept) | 0.678 | 0.054 | 12.661 | 0.000 |
| hospital | 0.174 | 0.020 | 8.678 | 0.000 |
| healthpoor | 0.234 | 0.048 | 4.872 | 0.000 |
| healthexcellent | -0.247 | 0.056 | -4.417 | 0.000 |
| chronic | 0.175 | 0.012 | 14.855 | 0.000 |
| sexfemale | 0.113 | 0.030 | 3.761 | 0.000 |
| school | 0.025 | 0.004 | 6.083 | 0.000 |
| insuranceyes | 0.234 | 0.038 | 6.189 | 0.000 |

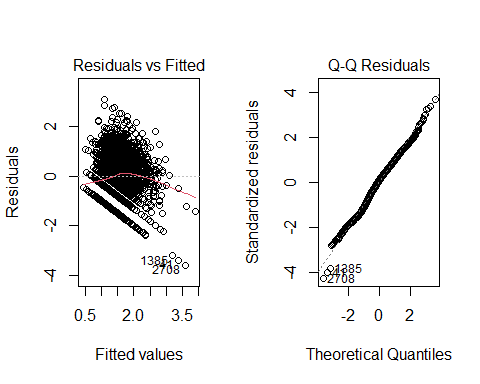
## Poisson Model Coefficients?

## Poisson model  
tidy(mod\_1) |> gt() |> fmt\_number(decimals = 3) |>   
 tab\_options(table.font.size = 20)

| term | estimate | std.error | statistic | p.value |
| --- | --- | --- | --- | --- |
| (Intercept) | 0.887 | 0.029 | 30.770 | 0.000 |
| hospital | 0.164 | 0.007 | 24.374 | 0.000 |
| healthpoor | 0.310 | 0.020 | 15.294 | 0.000 |
| healthexcellent | -0.359 | 0.035 | -10.287 | 0.000 |
| chronic | 0.137 | 0.005 | 26.082 | 0.000 |
| sexfemale | 0.098 | 0.015 | 6.641 | 0.000 |
| school | 0.031 | 0.002 | 14.808 | 0.000 |
| insuranceyes | 0.200 | 0.019 | 10.278 | 0.000 |

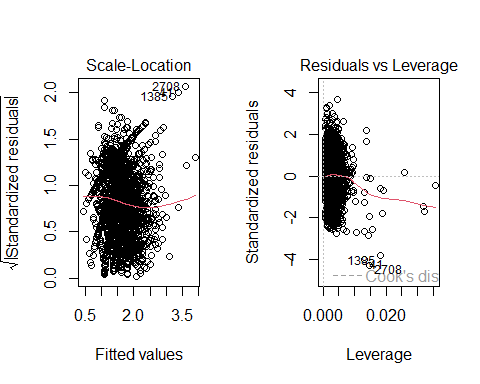
## Linear Regression Assumptions?

par(mfrow = c(1,2)); plot(mod\_0, which = 1:2)



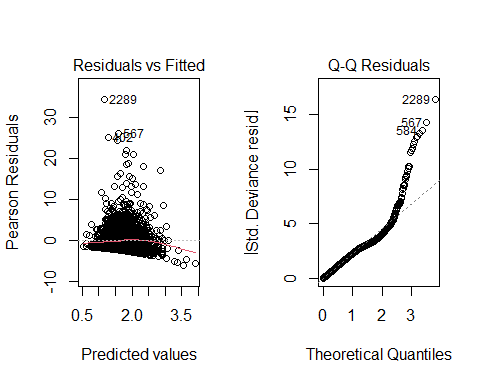
## Linear Regression Assumptions?

par(mfrow = c(1,2)); plot(mod\_0, which = c(3,5))



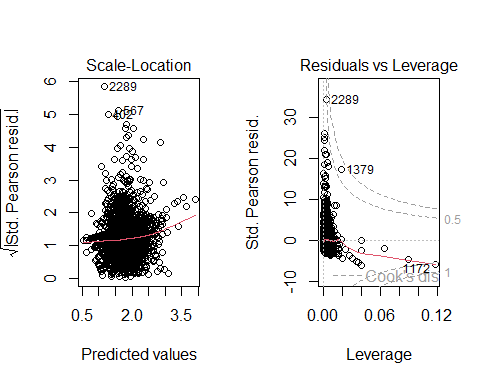
## Poisson Regression Plots?

par(mfrow = c(1,2)); plot(mod\_1, which = 1:2)



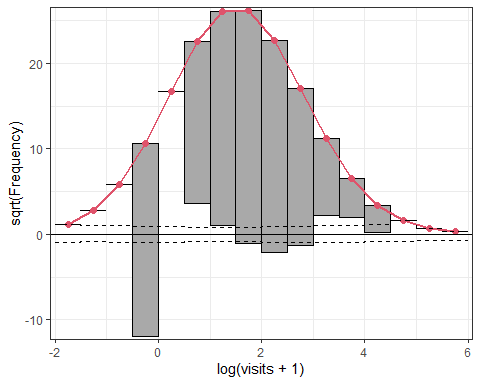
## Poisson Regression Plots

par(mfrow = c(1,2)); plot(mod\_1, which = c(3, 5))



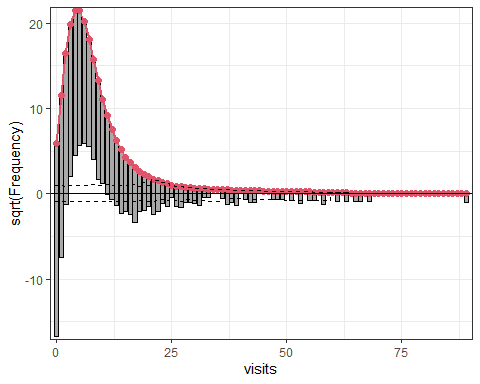
## Rootogram for Linear Model

rootogram(mod\_0)



## Rootogram for Poisson Model

rootogram(mod\_1)



## Test Sample Results (1st 6 subjects)

Actual visits seen in the test sample:

head(med\_test$visits)

[1] 1 16 9 3 0 44

Predicted visits From our linear model (mod\_0):

test\_0 <-   
 exp(predict(mod\_0, newdata = med\_test, type.predict = "response")) - 1  
  
head(test\_0)

1 2 3 4 5 6   
 4.098644 4.730367 2.412072 2.412072 2.412072 10.658053

Predicted visits from our Poisson model (mod\_1):

test\_1 <- predict(mod\_1, newdata = med\_test, type = "response")  
  
head(test\_1)

1 2 3 4 5 6   
 5.885106 6.878921 4.200529 4.200529 4.200529 12.235198

## Test Sample Predictions

No negative predictions with either model.

describe(test\_0) ## predictions from Linear fit

test\_0   
 n missing distinct Info Mean Gmd .05 .10   
 1102 0 489 1 3.835 2.037 1.702 1.972   
 .25 .50 .75 .90 .95   
 2.554 3.330 4.365 6.228 8.058   
  
lowest : 0.832359 0.836974 0.969082 1.05604 1.07177   
highest: 15.9109 16.8962 17.5769 24.3302 24.658

describe(test\_1) ## predictions from Poisson fit

test\_1   
 n missing distinct Info Mean Gmd .05 .10   
 1102 0 489 1 5.71 2.225 3.270 3.591   
 .25 .50 .75 .90 .95   
 4.334 5.228 6.385 8.429 9.839   
  
lowest : 1.94482 2.10986 2.32785 2.37599 2.40177  
highest: 17.5383 19.1728 19.3223 26.5918 26.5953

## Validation Results: These Two Models

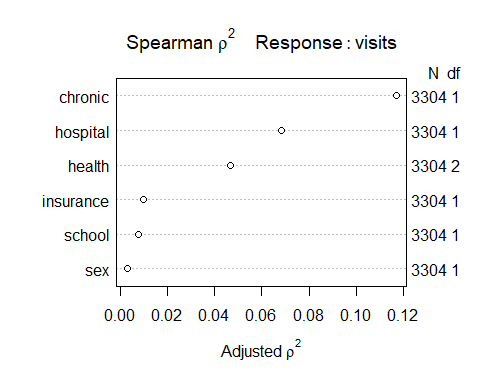
mets <- metric\_set(rsq, rmse, mae)  
  
test\_res <- bind\_cols(med\_test, pre\_m0 = test\_0, pre\_m1 = test\_1)  
  
m0\_sum <- mets(test\_res, truth = visits, estimate = pre\_m0) |>  
 mutate(model = "Linear")  
  
m1\_sum <- mets(test\_res, truth = visits, estimate = pre\_m1) |>  
 mutate(model = "Poisson")   
  
test\_sum <- bind\_rows(m0\_sum, m1\_sum) |>  
 pivot\_wider(names\_from = model, values\_from = .estimate)  
  
test\_sum |> select(-.estimator) |> gt() |> fmt\_number(decimals = 3) |>   
 tab\_options(table.font.size = 20)

| .metric | Linear | Poisson |
| --- | --- | --- |
| rsq | 0.100 | 0.095 |
| rmse | 6.125 | 5.915 |
| mae | 3.793 | 4.021 |

# Selecting non-linear terms after Spearman

## Spearman plot

plot(spearman2(visits ~ hospital + health + chronic + sex + school +   
 insurance, data = med\_train))



## Reiterating the Goal

This is the order of the predictors (chronic highest) on the Spearman plot from the previous slide.

| Predictor | Description |
| --- | --- |
| chronic | # of chronic conditions (all values 0-8) |
| hospital | # of hospital stays (all values 0-8) |
| health | self-rated health (poor, average, excellent) |
| insurance | subject (also) has private insurance? (yes/no) |
| school | years of education |
| sex | male or female |

## What might we do?

* chronic is a count (all values 0-8), then a gap to…
* hospital also quantitative, also a count (0-8)
* health is a 3-category factor

We might:

* include a restricted cubic spline with 4-5 knots in chronic
* include a rcs with fewer knots in hospital
* include an interaction between health and chronic or perhaps health and hospital

## Could we build an ols() fit?

Splines sometimes crash with discrete predictors (like counts.)

* For these data, it turns out that even a 3-knot spline in hospital fails (if we already have the four-knot spline in chronic), but the ols() function will let us add both interactions we’re considering.

d <- datadist(medicare); options(datadist = "d")  
  
mod\_toobig <- ols(log(visits + 1) ~   
 rcs(chronic, 4) + hospital \* health +   
 chronic %ia% health +  
 sex + school + insurance, data = med\_train)

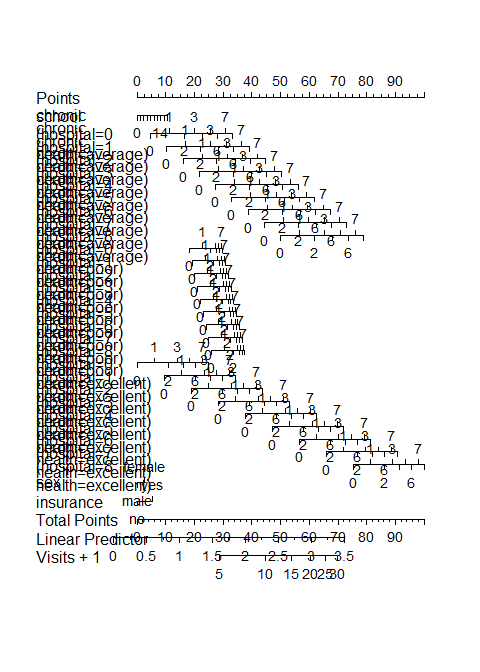
## Why is this model “too big”?

mod\_toobig

Linear Regression Model  
  
ols(formula = log(visits + 1) ~ rcs(chronic, 4) + hospital \*   
 health + chronic %ia% health + sex + school + insurance,   
 data = med\_train)  
  
 Model Likelihood Discrimination   
 Ratio Test Indexes   
Obs 3304 LR chi2 664.03 R2 0.182   
sigma0.8363 d.f. 13 R2 adj 0.179   
d.f. 3290 Pr(> chi2) 0.0000 g 0.444   
  
Residuals  
  
 Min 1Q Median 3Q Max   
-2.42109 -0.55490 0.08359 0.56662 3.07394   
  
 Coef S.E. t Pr(>|t|)  
Intercept 0.5590 0.0575 9.73 <0.0001   
chronic 0.3011 0.0546 5.52 <0.0001   
chronic' -0.2051 0.2579 -0.80 0.4264   
chronic'' 0.2159 0.6311 0.34 0.7323   
hospital 0.2475 0.0249 9.95 <0.0001   
health=poor 0.5914 0.0952 6.21 <0.0001   
health=excellent -0.2022 0.0730 -2.77 0.0057   
chronic \* health=poor -0.0931 0.0335 -2.78 0.0054   
chronic \* health=excellent -0.0213 0.0565 -0.38 0.7058   
sex=female 0.1088 0.0297 3.66 0.0003   
school 0.0257 0.0041 6.20 <0.0001   
insurance=yes 0.2353 0.0375 6.28 <0.0001   
hospital \* health=poor -0.2053 0.0421 -4.88 <0.0001   
hospital \* health=excellent 0.1623 0.1493 1.09 0.2769

## Uh, oh.

plot(nomogram(mod\_toobig, fun = exp, funlabel = "Visits + 1"))



## A more reasonable option?

d <- datadist(medicare); options(datadist = "d")  
  
mod\_new <- ols(log(visits + 1) ~   
 rcs(chronic, 4) + hospital + health +   
 chronic %ia% health +  
 sex + school + insurance, data = med\_train)

## What does this mod\_new show?

mod\_new

Linear Regression Model  
  
ols(formula = log(visits + 1) ~ rcs(chronic, 4) + hospital +   
 health + chronic %ia% health + sex + school + insurance,   
 data = med\_train)  
  
 Model Likelihood Discrimination   
 Ratio Test Indexes   
Obs 3304 LR chi2 637.75 R2 0.176   
sigma0.8393 d.f. 11 R2 adj 0.173   
d.f. 3292 Pr(> chi2) 0.0000 g 0.435   
  
Residuals  
  
 Min 1Q Median 3Q Max   
-3.06891 -0.54950 0.08035 0.56946 3.04632   
  
 Coef S.E. t Pr(>|t|)  
Intercept 0.5743 0.0576 9.97 <0.0001   
chronic 0.3036 0.0548 5.55 <0.0001   
chronic' -0.1710 0.2588 -0.66 0.5089   
chronic'' 0.1165 0.6331 0.18 0.8540   
hospital 0.1799 0.0199 9.02 <0.0001   
health=poor 0.5437 0.0951 5.72 <0.0001   
health=excellent -0.1940 0.0724 -2.68 0.0074   
chronic \* health=poor -0.1199 0.0331 -3.62 0.0003   
chronic \* health=excellent -0.0163 0.0563 -0.29 0.7718   
sex=female 0.1051 0.0298 3.53 0.0004   
school 0.0256 0.0042 6.16 <0.0001   
insurance=yes 0.2307 0.0376 6.14 <0.0001

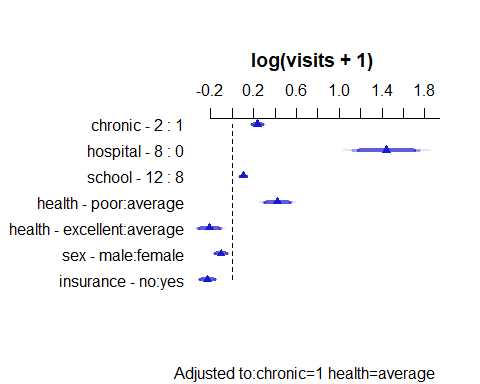
## How many df did we add here?

anova(mod\_new)

Analysis of Variance Response: log(visits + 1)   
  
 Factor d.f. Partial SS MS   
 chronic (Factor+Higher Order Factors) 5 189.349521 37.8699042  
 All Interactions 2 9.251314 4.6256570  
 Nonlinear 2 9.483346 4.7416730  
 hospital 1 57.342322 57.3423218  
 health (Factor+Higher Order Factors) 4 39.675076 9.9187689  
 All Interactions 2 9.251314 4.6256570  
 chronic \* health (Factor+Higher Order Factors) 2 9.251314 4.6256570  
 sex 1 8.763370 8.7633703  
 school 1 26.694549 26.6945488  
 insurance 1 26.545793 26.5457929  
 TOTAL NONLINEAR + INTERACTION 4 31.942728 7.9856821  
 REGRESSION 11 493.787139 44.8897399  
 ERROR 3292 2319.211257 0.7044992  
 F P   
 53.75 <.0001  
 6.57 0.0014  
 6.73 0.0012  
 81.39 <.0001  
 14.08 <.0001  
 6.57 0.0014  
 6.57 0.0014  
 12.44 0.0004  
 37.89 <.0001  
 37.68 <.0001  
 11.34 <.0001  
 63.72 <.0001

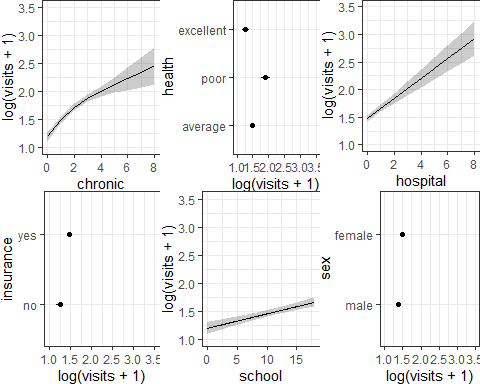
## What does this ols() fit look like?

plot(summary(mod\_new))



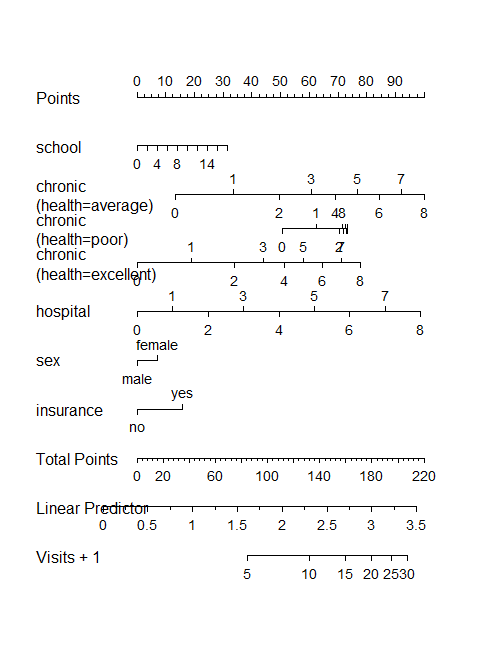
## What does this ols() fit look like?

ggplot(Predict(mod\_new))



## How’s the nomogram?

plot(nomogram(mod\_new, fun = exp, funlabel = "Visits + 1"))



# Can we fit a Poisson model with a function from rms?

## The Glm() function in rms

d <- datadist(medicare); options(datadist = "d")  
  
mod\_1\_Glm <- Glm(visits ~ hospital + health + chronic + sex + school +   
 insurance, data = med\_train, family = poisson())

and we could have used rcs() or polynomials or interactions if we wanted to do so.

Complete and updated documentation for the rms package is found at <https://hbiostat.org/r/rms/>.

### Does a Glm() fit do everything we are used to?

* Nope. No validate() or calibrate() methods exist.

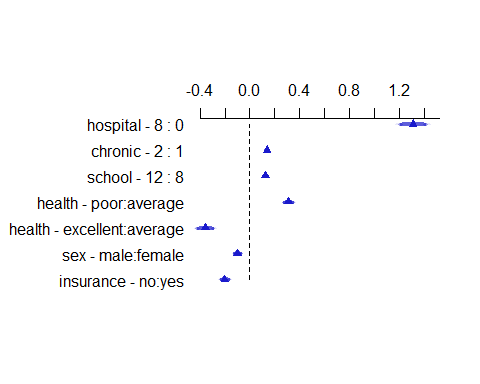
## What’s in mod\_1\_Glm?

mod\_1\_Glm

General Linear Model  
  
Glm(formula = visits ~ hospital + health + chronic + sex + school +   
 insurance, family = poisson(), data = med\_train)  
  
 Model Likelihood   
 Ratio Test   
 Obs3304 LR chi2 3019.53   
Residual d.f.3296 d.f. 7   
 g 0.386 Pr(> chi2) <0.0001   
  
 Coef S.E. Wald Z Pr(>|Z|)  
Intercept 0.8866 0.0288 30.77 <0.0001   
hospital 0.1636 0.0067 24.37 <0.0001   
health=poor 0.3096 0.0202 15.29 <0.0001   
health=excellent -0.3588 0.0349 -10.29 <0.0001   
chronic 0.1373 0.0053 26.08 <0.0001   
sex=female 0.0983 0.0148 6.64 <0.0001   
school 0.0313 0.0021 14.81 <0.0001   
insurance=yes 0.2002 0.0195 10.28 <0.0001

## What can we do: mod\_1\_Glm?

plot(summary(mod\_1\_Glm))



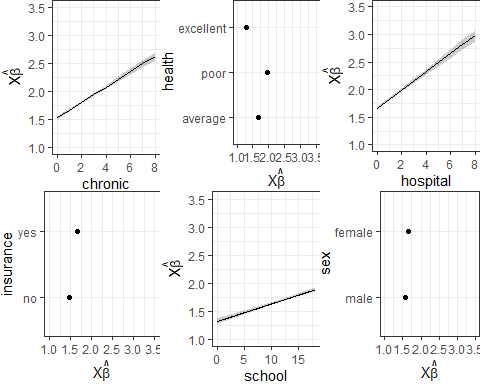
## What can we do: mod\_1\_Glm?

summary(mod\_1\_Glm)

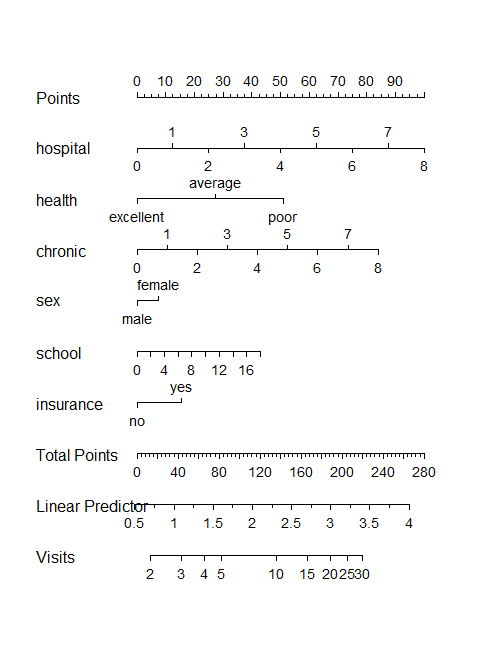
Effects Response : visits   
  
 Factor Low High Diff. Effect S.E. Lower 0.95  
 hospital 0 8 8 1.308400 0.0536810 1.20320   
 chronic 1 2 1 0.137350 0.0052661 0.12702   
 school 8 12 4 0.125030 0.0084433 0.10848   
 health - poor:average 1 2 NA 0.309610 0.0202440 0.26992   
 health - excellent:average 1 3 NA -0.358760 0.0348750 -0.42714   
 sex - male:female 2 1 NA -0.098325 0.0148050 -0.12735   
 insurance - no:yes 2 1 NA -0.200250 0.0194840 -0.23845   
 Upper 0.95  
 1.413700   
 0.147670   
 0.141590   
 0.349300   
 -0.290380   
 -0.069297   
 -0.162050

## What can we do: mod\_1\_Glm?

ggplot(Predict(mod\_1\_Glm))



plot(nomogram(mod\_1\_Glm, fun = exp, funlabel = "Visits",  
 fun.at = c(1, 2, 3, 4, 5, 10, 15, 20, 25, 30)))



# Checking Assumptions in Logistic Regression Models

## Linear Regression vs. Logistic Regression

Adapted from <https://www.statology.org/assumptions-of-logistic-regression/>

In contrast to linear regression, logistic regression does not require:

* A linear relationship between the predictors and the outcome.
* The residuals of the model to be normally distributed.
* The residuals to have constant variance (homoscedasticity/)

## Assumptions of Logistic Regression

Adapted from <https://www.statology.org/assumptions-of-logistic-regression/>

1. The outcome variable is binary.
2. The observations are independent from each other. (They shouldn’t show a pattern in time or space.)
3. There is no severe multicollinearity among the predictors (we use VIF > 5 as an indicator of trouble.)

## Assumptions of Logistic Regression

1. There are no extreme outliers (Cook’s distance > .5 is what R flags as problematic[[1]](#footnote-104).)
2. The sample size is sufficiently large (see next few slides.)
3. There is a linear relationship between predictors and the logit of the outcome (see the final few slides.)

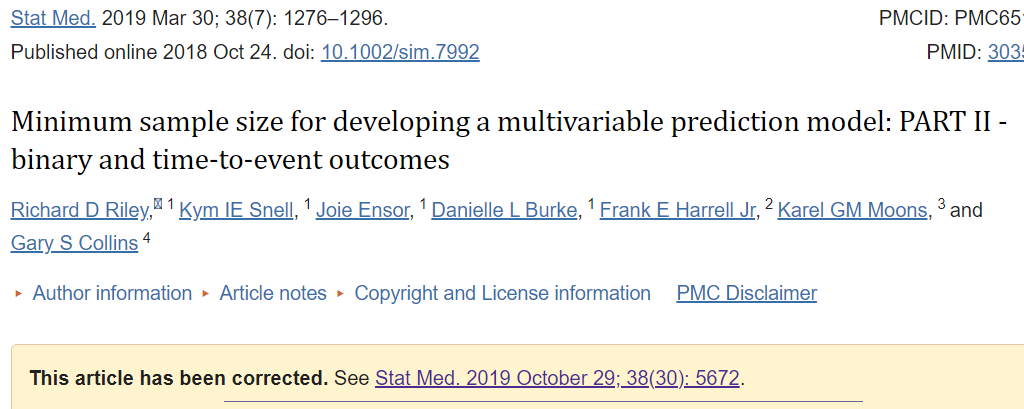
## What does sufficiently large mean?

1. Some people like a simple rule like 500 observations overall and 10 events (where an event is the smaller of your two outcome groups) per predictor parameter. See [Long’s 1997 book (pdf)](https://jslsoc.sitehost.iu.edu/files_research/rm4cldv/sage1997/rm4cldv_toc.pdf).

For *Project A*, we focus on keeping the number of predictors below (4 + (N-100)) / 100) where N is the size of the smaller of your two outcome groups. I wouldn’t use that standard outside of Project A, though.

## What does sufficiently large mean?

1. Riley et al. in [Statistics in Medicine](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6519266/) develop an estimation scheme for the needed sample sizes, and motivate it with several examples. It’s pretty complex but it’s a good option.



## An “Old” Logistic Regression Example

I’ll use the mov23a data that I built back in Class 08. An RDS version of the data is available now on our 432-data page.

mov23a <- read\_rds("c16/data/mov23a.rds")  
  
mov23a

# A tibble: 187 × 9  
 film\_id film bechdel age gross metascore mpa3 comedy drama  
 <chr> <chr> <fct> <dbl> <dbl> <dbl> <fct> <int> <int>  
 1 M001 3 Idiots Fail 15 6.03e+1 67 PG-13 1 1  
 2 M002 8 1/2 Pass 61 1.96e-1 93 Other 0 1  
 3 M003 10 Things I Hate … Pass 25 5.35e+1 70 PG-13 1 1  
 4 M004 2001: A Space Ody… Fail 56 6.64e+1 84 Other 0 0  
 5 M005 About Elly (Darba… Pass 15 8.79e-1 87 Other 0 1  
 6 M006 About Time Pass 11 8.71e+1 55 R 1 1  
 7 M007 Alien Pass 45 1.06e+2 89 R 0 0  
 8 M008 Amadeus Pass 40 5.21e+1 87 Other 0 1  
 9 M009 Avatar Pass 15 2.92e+3 83 PG-13 0 0  
10 M010 Avengers: Infinit… Pass 6 2.80e+3 78 PG-13 0 0  
# ℹ 177 more rows

## Goal of this Example

* We’re trying to predict bechdel result (Pass or Fail) using three predictors: age, mpa3 and metascore, as we did in slides 40-52 in the Class 08 Slides.
* When we did this back in Class 08, we got C = 0.620 and Nagelkerke = 0.062, with a likelihood ratio *p*-value of 0.0666

Now, we want to see if our model passes these six assumption checks.

## Fitting the Model

We’ll use both glm() and lrm() to fit the model.

mod2\_glm <- glm((bechdel == "Pass") ~ age + metascore + mpa3,   
 data = mov23a, family = binomial(link = logit))  
  
ddd <- datadist(mov23a); options(datadist = "ddd")  
mod2\_lrm <- lrm((bechdel == "Pass") ~ age + metascore + mpa3,   
 data = mov23a, x = TRUE, y = TRUE)

We did all of this in Class 08. We have no missing data here.

## Tidied Coefficients from our Model

tidy(mod2\_glm, exponentiate = TRUE, conf.int = TRUE, conf.level = 0.9) |>  
 gt() |> fmt\_number(decimals = 3) |> tab\_options(table.font.size = 20)

| term | estimate | std.error | statistic | p.value | conf.low | conf.high |
| --- | --- | --- | --- | --- | --- | --- |
| (Intercept) | 4.277 | 0.743 | 1.957 | 0.050 | 1.285 | 14.949 |
| age | 0.973 | 0.012 | -2.347 | 0.019 | 0.953 | 0.991 |
| metascore | 0.991 | 0.010 | -0.920 | 0.358 | 0.974 | 1.007 |
| mpa3R | 0.850 | 0.375 | -0.434 | 0.664 | 0.458 | 1.575 |
| mpa3Other | 1.620 | 0.402 | 1.200 | 0.230 | 0.842 | 3.171 |

## First Two Assumptions

1. The outcome variable is binary.
   * OK. bechdel is either Pass or Fail.

mov23a |> count(bechdel)

# A tibble: 2 × 2  
 bechdel n  
 <fct> <int>  
1 Fail 80  
2 Pass 107

1. The observations are independent from each other. (They shouldn’t show a pattern in time or space.)
   * The data are cross-sectional. No one film’s results should affect another film’s results, so we’re OK.

## Assumption Three

1. There is no severe multicollinearity among the predictors (we use VIF > 5 as an indicator of trouble.)
   * Let’s look at the VIF results. Are we OK?

car::vif(mod2\_glm)

GVIF Df GVIF^(1/(2\*Df))  
age 1.162272 1 1.078087  
metascore 1.066514 1 1.032722  
mpa3 1.223542 2 1.051731

rms::vif(mod2\_lrm)

age metascore mpa3=R mpa3=Other   
 1.162273 1.066514 1.372551 1.490063

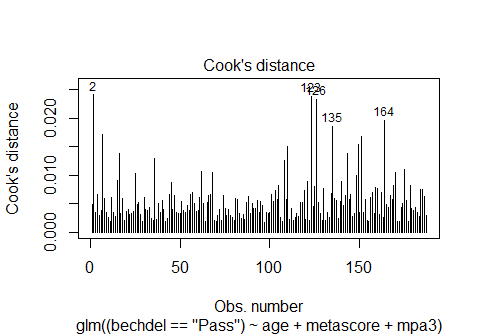
## Assumption Four

1. There are no extreme outliers (no Cook’s distance > 0.5)

max(cooks.distance(mod2\_glm))

[1] 0.02409287

plot(mod2\_glm, which = 4, id.n = 5)



## Assumption Five

1. The sample size is sufficiently large.

* Recall that we have 107 Pass and 80 Fail subjects in mov23a.

glance(mod2\_glm) |> select(nobs) ## could use mod2\_lrm$stats["Obs"]

# A tibble: 1 × 1  
 nobs  
 <int>  
1 187

Does this seem like enough observations to fit a logistic regression model with 3 predictors (and 4 predictor coefficients) under consideration?

## Assumption Six

1. There is a linear relationship between predictors and the logit of the outcome.

A **Box-Tidwell test** is a common strategy to test this assumptions, but it doesn’t work for logistic models [according to John Fox](https://stackoverflow.com/questions/56350546/how-to-use-the-box-tidwell-function-with-a-logistic-regression-in-r), inventor of the car package[[2]](#footnote-125).

He instead recommends what he calls Component + Residual plots and some people call Partial Residual plots, which can be used for both linear and generalized linear models.

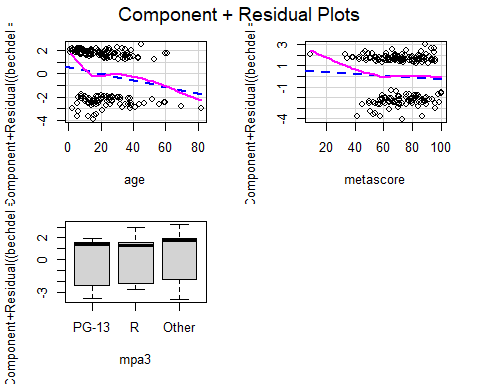
## Interpreting the Partial Residual Plots

* The blue dashed line shows the expected residuals if the relationship between the predictor and response variable (here the log odds of our outcome) was linear.
* The solid pink curve shows a loess smooth of the actual residuals.

If the two lines are meaningfully different, then this is evidence of a nonlinear relationship. One way to fix this issue is to build a transformation on the predictor variables, or consider incorporating some non-linear terms.

## Running Partial Residual Plots

crPlots(mod2\_glm) ## crPlots comes from the car package



## What Dr. Love does

I have often used an alternative called CERES Plots (invented by Dennis Cook) when I fit logistic regression models.

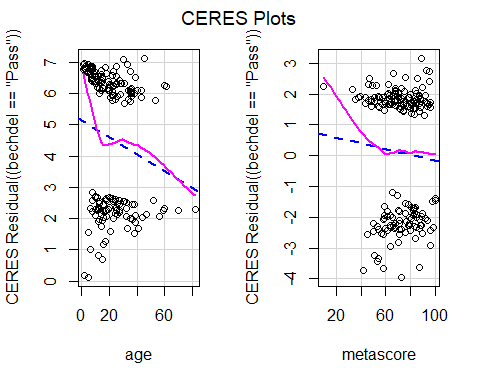
Again, the blue line shows the expected residuals if the relationship between the predictor and response variable (here the log odds of our outcome) was linear, while the pink line shows a loess smooth of the actual residuals.

This looks only at the quantitative predictors.

## An Alternative: CERES Plots

ceresPlots(mod2\_glm) ## ceresPlots also comes from car package

Warning in ceresPlots.default(mod2\_glm): Factors skipped in drawing CERES  
plots.



I see nothing especially problematic here.

# In closing, have a nice break, and good luck with project A!

1. Some argue for a standard of 0.25, or 1, or even 4/n, where n is your sample size. [↑](#footnote-ref-104)
2. See Fox, J. (2016) Applied Regression Analysis and Generalized Linear Models, 3rd Ed., Sage. [↑](#footnote-ref-125)