432 Class 20

https://thomaselove.github.io/432-2025

2025-03-27

## Today’s Topic

Asbestos Exposure in the US Navy: A new example predicting an ordered multi-categorical outcome

* Proportional Odds Logistic Regression Models
  + with MASS::polr()
  + with rms::lrm()
* Fitting Ordinal Logistic Regressions with rms::orm()

Chapter 27 of the Course Notes describes this material.

## Today’s R Setup

knitr::opts\_chunk$set(comment=NA)  
  
library(janitor); library(broom); library(gt)

Attaching package: 'janitor'

The following objects are masked from 'package:stats':  
  
 chisq.test, fisher.test

library(here); library(conflicted)

here() starts at D:/Teaching/432/2025/432-slides-2025

library(mosaic) ## but just for favstats

Registered S3 method overwritten by 'mosaic':  
 method from   
 fortify.SpatialPolygonsDataFrame ggplot2

The 'mosaic' package masks several functions from core packages in order to add   
additional features. The original behavior of these functions should not be affected by this.

library(nnet)  
library(MASS)  
library(rms)

Loading required package: Hmisc

Attaching package: 'Hmisc'

The following objects are masked from 'package:dplyr':  
  
 src, summarize

The following object is masked from 'package:gt':  
  
 html

The following objects are masked from 'package:base':  
  
 format.pval, units

library(easystats)

# Attaching packages: easystats 0.7.4 (red = needs update)  
✔ bayestestR 0.15.2 ✔ correlation 0.8.7   
✖ datawizard 1.0.1 ✔ effectsize 1.0.0   
✔ insight 1.1.0 ✔ modelbased 0.10.0  
✔ performance 0.13.0 ✔ parameters 0.24.2  
✔ report 0.6.1 ✔ see 0.11.0  
  
Restart the R-Session and update packages with `easystats::easystats\_update()`.

library(tidyverse)

── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
✔ forcats 1.0.0 ✔ stringr 1.5.1  
✔ lubridate 1.9.4 ✔ tibble 3.2.1  
✔ purrr 1.0.4 ✔ tidyr 1.3.1  
✔ readr 2.1.5

conflicts\_prefer(dplyr::filter, dplyr::select,  
 dplyr::summarize, dplyr::count,  
 base::mean, base::max, janitor::clean\_names)

[conflicted] Will prefer dplyr::filter over any other package.

[conflicted] Will prefer dplyr::select over any other package.  
[conflicted] Will prefer dplyr::summarize over any other package.  
[conflicted] Will prefer dplyr::count over any other package.  
[conflicted] Will prefer base::mean over any other package.  
[conflicted] Will prefer base::max over any other package.  
[conflicted] Will prefer janitor::clean\_names over any other package.

theme\_set(theme\_bw())

# POLR and Ordinal Regression Models

## Asbestos exposure in the U.S. Navy

These data describe 83 Navy workers[[1]](#footnote-22), engaged in jobs involving potential asbestos exposure.

* The workers were either removing asbestos tile or asbestos insulation, and we might reasonably expect that those exposures would be different.
* We’d expect more exposure with insulation removal.

## Asbestos exposure in the U.S. Navy

Data describe 83 Navy workers[[2]](#footnote-24) with potential asbestos exposure..

* The workers either worked with general ventilation (like a fan or naturally occurring wind) or negative pressure (where a pump with a High Efficiency Particulate Air filter is used to draw air (and fibers) from the work area.)
* We’d expect more exposure with general ventilation.

## Asbestos exposure in the U.S. Navy

83 Navy workers[[3]](#footnote-26) with potential asbestos exposure…

* The duration of a sampling period (in minutes) was recorded, and their asbestos exposure was classified as:
  + low exposure (< 0.05 fibers per cubic centimeter),
  + action level (between 0.05 and 0.1) and
  + above the legal limit (more than 0.1 fibers per cc).
* Sampling periods ranged from 30 to 300 minutes.

## Ingest and clean asbestos data

asbestos <- read\_csv(here("c20/data/asbestos.csv"),   
 show\_col\_types = FALSE) |>  
 clean\_names() |>  
 mutate(across(where(is\_character), as\_factor),  
 exposure = fct\_relevel(exposure, "1\_Low", "2\_Action", "3\_AboveLimit"),  
 exposure = factor(exposure, ordered = TRUE),  
 worker = as.character(worker))  
  
summary(asbestos |> select(-worker))

exposure task ventilation duration   
 1\_Low :45 Tile :37 NP :49 Min. : 30.0   
 2\_Action : 6 Insulation:46 General:34 1st Qu.: 85.0   
 3\_AboveLimit:32 Median :138.0   
 Mean :147.1   
 3rd Qu.:212.5   
 Max. :300.0

## Our Outcome and Modeling task

* exposure is determined by taking air samples in a circle of diameter 2.5 feet around the worker’s mouth and nose.

Our planned predictors for exposure are:

* task (Tile or Insulation),
* ventilation (Negative Pressure (NP) or General), and
* duration (in minutes).

## Effects of Task and Ventilation

We anticipated greater exposure with Insulation, rather than Tile, and with General ventilation vs. Negative Pressure.

asbestos |> tabyl(task, exposure) |>   
 gt() |> tab\_options(table.font.size = 20)

| task | 1\_Low | 2\_Action | 3\_AboveLimit |
| --- | --- | --- | --- |
| Tile | 32 | 2 | 3 |
| Insulation | 13 | 4 | 29 |

asbestos |> tabyl(ventilation, exposure) |>   
 gt() |> tab\_options(table.font.size = 20)

| ventilation | 1\_Low | 2\_Action | 3\_AboveLimit |
| --- | --- | --- | --- |
| NP | 39 | 2 | 8 |
| General | 6 | 4 | 24 |

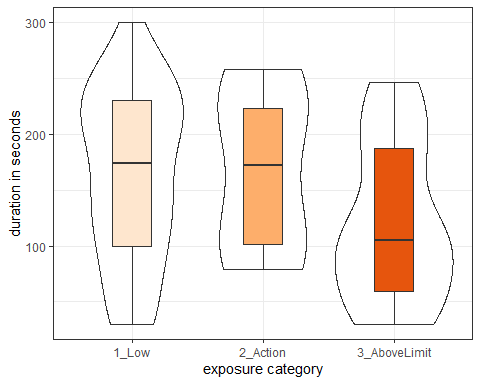
## Exposure and Duration

Is there a strong relationship of exposure and duration?

favstats(duration ~ exposure, data = asbestos) |>  
 gt() |> fmt\_number(columns = mean:sd, decimals = 2) |>  
 tab\_options(table.font.size = 20)

| exposure | min | Q1 | median | Q3 | max | mean | sd | n | missing |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1\_Low | 30 | 100.0 | 174 | 230.00 | 300 | 162.56 | 76.09 | 45 | 0 |
| 2\_Action | 79 | 101.5 | 172 | 223.00 | 258 | 166.50 | 75.77 | 6 | 0 |
| 3\_AboveLimit | 30 | 59.5 | 105 | 187.25 | 246 | 121.66 | 70.72 | 32 | 0 |

ggplot(asbestos, aes(x = exposure, y = duration)) +  
 geom\_violin() +  
 geom\_boxplot(aes(fill = exposure), width = 0.3) +  
 guides(fill = "none") +  
 scale\_fill\_brewer(type = "seq", palette = "Oranges") +  
 labs(y = "duration in seconds", x = "exposure category")



# Fitting polr models with the MASS::polr function

## Proportional-Odds Cumulative Logit

We’ll use the polr function in the **MASS** package.

* Clearly, exposure group (3) Above legal limit, is worst, followed by group (2) Action level, and then group (1) Low exposure.
* We’ll have two binary (1/0) predictors (one for task and one for ventilation) and one quantitative predictor (for duration).

## Equations to be Fit

* The model will have two logit equations: one comparing group (1) to group (2) and one comparing group (2) to group (3), and three slopes, for a total of five free parameters.

and

## Centering Duration

In order to make our result more interpretable, I suggest we center each of our quantitative predictors (in this case, that’s just centering duration.) Recall that mean(duration) = 147.1 minutes in these data.

asbestos <- asbestos |>   
 mutate(dur\_c = duration - mean(duration))

A value of dur\_c = 0 thus means that we have the mean level of duration.

## Model Equations

Note that the intercept term is the only piece that varies across the two equations shown in the previous slide.

* A positive coefficient means that increasing the value of that predictor tends to *raise* the exposure category, and thus *increase* the asbestos exposure.

### Fitting the Model

modelA <- polr(exposure ~ task + ventilation + dur\_c,   
 data=asbestos, Hess = TRUE)

## modelA parameters

model\_parameters(modelA, pretty\_names = FALSE, ci = 0.95)

# alpha  
  
Parameter | Log-Odds | SE | 95% CI | t(78) | p  
-----------------------------------------------------------------------  
1\_Low|2\_Action | 2.45 | 0.57 | [1.32, 3.59] | 4.30 | < .001  
2\_Action|3\_AboveLimit | 3.00 | 0.61 | [1.79, 4.21] | 4.92 | < .001  
  
# beta  
  
Parameter | Log-Odds | SE | 95% CI | t(78) | p  
--------------------------------------------------------------------------  
taskInsulation | 2.25 | 0.64 | [ 1.04, 3.61] | 3.49 | < .001  
ventilationGeneral | 2.16 | 0.57 | [ 1.07, 3.31] | 3.80 | < .001  
dur\_c | -7.08e-04 | 3.80e-03 | [-0.01, 0.01] | -0.19 | 0.853

Uncertainty intervals (profile-likelihood) and p-values (two-tailed)  
 computed using a Wald t-distribution approximation.

The model has a log- or logit-link. Consider using `exponentiate =  
 TRUE` to interpret coefficients as ratios.  
   
Some coefficients seem to be rather large, which may indicate issues  
 with (quasi) complete separation. Consider using bias-corrected or  
 penalized regression models.

## modelA Summary

summary(modelA)

Call:  
polr(formula = exposure ~ task + ventilation + dur\_c, data = asbestos,   
 Hess = TRUE)  
  
Coefficients:  
 Value Std. Error t value  
taskInsulation 2.251344 0.644593 3.4927  
ventilationGeneral 2.156963 0.567535 3.8006  
dur\_c -0.000708 0.003797 -0.1865  
  
Intercepts:  
 Value Std. Error t value  
1\_Low|2\_Action 2.4550 0.5714 4.2963  
2\_Action|3\_AboveLimit 3.0013 0.6094 4.9248  
  
Residual Deviance: 99.87952   
AIC: 109.8795

## Direction of Model Effects

Here are coefficient estimates for the three predictors.

tidy(modelA) |> filter(coef.type == "coefficient") |>  
 gt() |> fmt\_number(decimals = 3) |>  
 tab\_options(table.font.size = 20)

| term | estimate | std.error | statistic | coef.type |
| --- | --- | --- | --- | --- |
| taskInsulation | 2.251 | 0.645 | 3.493 | coefficient |
| ventilationGeneral | 2.157 | 0.568 | 3.801 | coefficient |
| dur\_c | -0.001 | 0.004 | -0.186 | coefficient |

* The estimated slope for task = Insulation is 2.25.
  + Since the slope is positive, task = Insulation produces an *increased* exposure level compared to task = Tile when ventilation and duration are held constant.

## Effect of Task via Odds Ratio + CI

tidy(modelA, exponentiate = TRUE, conf.int = TRUE) |>  
 filter(coef.type == "coefficient") |>  
 gt() |> fmt\_number(decimals = 3) |> tab\_options(table.font.size = 20)

| term | estimate | std.error | statistic | conf.low | conf.high | coef.type |
| --- | --- | --- | --- | --- | --- | --- |
| taskInsulation | 9.500 | 0.645 | 3.493 | 2.826 | 36.787 | coefficient |
| ventilationGeneral | 8.645 | 0.568 | 3.801 | 2.917 | 27.465 | coefficient |
| dur\_c | 0.999 | 0.004 | -0.186 | 0.992 | 1.007 | coefficient |

* Assuming ventilation and duration remain constant, suppose Al has task = Insulation and Bob has task = Tile.
* modelA: Odds of higher asbestos exposure are 9.5 (95% CI 2.8 to 36.8) times as large for Al as they are for Bob.

## Ventilation Effect

tidy(modelA, exponentiate = TRUE, conf.int = TRUE) |>  
 filter(coef.type == "coefficient") |>  
 gt() |> fmt\_number(decimals = 3) |> tab\_options(table.font.size = 20)

| term | estimate | std.error | statistic | conf.low | conf.high | coef.type |
| --- | --- | --- | --- | --- | --- | --- |
| taskInsulation | 9.500 | 0.645 | 3.493 | 2.826 | 36.787 | coefficient |
| ventilationGeneral | 8.645 | 0.568 | 3.801 | 2.917 | 27.465 | coefficient |
| dur\_c | 0.999 | 0.004 | -0.186 | 0.992 | 1.007 | coefficient |

* Assuming task and duration remain constant, modelA suggests the odds of higher exposure are 8.65 (95% CI 2.9, 27.5) times as large when using General ventilation.
* Impact of duration appears quite small: odds ratio is essentially 1, with 95% CI (0.99, 1.01).

## modelA: Equation 1

tidy(modelA) |>   
 gt() |> fmt\_number(decimals = 3) |> tab\_options(table.font.size = 20)

| term | estimate | std.error | statistic | coef.type |
| --- | --- | --- | --- | --- |
| taskInsulation | 2.251 | 0.645 | 3.493 | coefficient |
| ventilationGeneral | 2.157 | 0.568 | 3.801 | coefficient |
| dur\_c | -0.001 | 0.004 | -0.186 | coefficient |
| 1\_Low|2\_Action | 2.455 | 0.571 | 4.296 | scale |
| 2\_Action|3\_AboveLimit | 3.001 | 0.609 | 4.925 | scale |

* 2.455 is the estimated log odds of falling into category (1) low exposure versus all other categories, when all other predictors (task, ventilation and centered duration) are zero.

## modelA: Equation 2

tidy(modelA) |>   
 gt() |> fmt\_number(decimals = 3) |> tab\_options(table.font.size = 20)

| term | estimate | std.error | statistic | coef.type |
| --- | --- | --- | --- | --- |
| taskInsulation | 2.251 | 0.645 | 3.493 | coefficient |
| ventilationGeneral | 2.157 | 0.568 | 3.801 | coefficient |
| dur\_c | -0.001 | 0.004 | -0.186 | coefficient |
| 1\_Low|2\_Action | 2.455 | 0.571 | 4.296 | scale |
| 2\_Action|3\_AboveLimit | 3.001 | 0.609 | 4.925 | scale |

* 3.001 is the estimated log odds of falling into category (1) or (2) versus category (3), when all other predictors (task, ventilation and centered duration) are zero.

## modelA First Equation

tidy(modelA) |>   
 gt() |> fmt\_number(decimals = 3) |> tab\_options(table.font.size = 20)

| term | estimate | std.error | statistic | coef.type |
| --- | --- | --- | --- | --- |
| taskInsulation | 2.251 | 0.645 | 3.493 | coefficient |
| ventilationGeneral | 2.157 | 0.568 | 3.801 | coefficient |
| dur\_c | -0.001 | 0.004 | -0.186 | coefficient |
| 1\_Low|2\_Action | 2.455 | 0.571 | 4.296 | scale |
| 2\_Action|3\_AboveLimit | 3.001 | 0.609 | 4.925 | scale |

## modelA Second Equation

tidy(modelA) |>   
 gt() |> fmt\_number(decimals = 3) |> tab\_options(table.font.size = 20)

| term | estimate | std.error | statistic | coef.type |
| --- | --- | --- | --- | --- |
| taskInsulation | 2.251 | 0.645 | 3.493 | coefficient |
| ventilationGeneral | 2.157 | 0.568 | 3.801 | coefficient |
| dur\_c | -0.001 | 0.004 | -0.186 | coefficient |
| 1\_Low|2\_Action | 2.455 | 0.571 | 4.296 | scale |
| 2\_Action|3\_AboveLimit | 3.001 | 0.609 | 4.925 | scale |

## model\_performance() and glance()

model\_performance(modelA)

Can't calculate log-loss.

Can't calculate proper scoring rules for ordinal, multinomial or  
 cumulative link models.

# Indices of model performance  
  
AIC | AICc | BIC | Nagelkerke's R2 | RMSE | Sigma  
-------------------------------------------------------------  
109.880 | 110.659 | 121.974 | 0.526 | 1.815 | 1.117

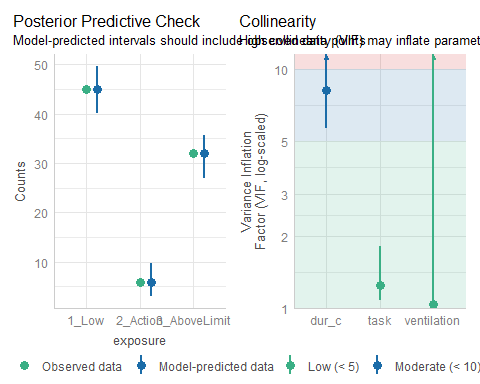
glance(modelA)

# A tibble: 1 × 7  
 edf logLik AIC BIC deviance df.residual nobs  
 <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
1 5 -49.9 110. 122. 99.9 78 83

## check\_model() results

check\_model(modelA)

Cannot simulate residuals for models of class `polr`. Please try  
 `check\_model(..., residual\_type = "normal")` instead.



# Comparing polr models

## modelA vs. “Intercept only” model

model.1 <- polr(exposure ~ 1, data=asbestos)  
anova(model.1, modelA) |>   
 gt() |> tab\_options(table.font.size = 20)

| Model | Resid. df | Resid. Dev | Test | Df | LR stat. | Pr(Chi) |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 81 | 147.61971 |  | NA | NA | NA |
| task + ventilation + dur\_c | 78 | 99.87952 | 1 vs 2 | 3 | 47.74019 | 2.41857e-10 |

### Can we compare AIC and BIC?

AIC(model.1, modelA)

df AIC  
model.1 2 151.6197  
modelA 5 109.8795

BIC(model.1, modelA)

df BIC  
model.1 2 156.4574  
modelA 5 121.9737

## Compare Parameters

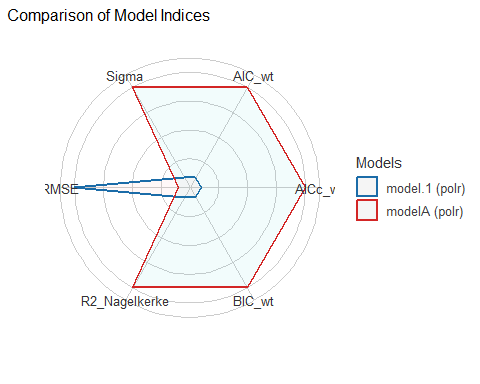
compare\_parameters(model.1, modelA)

Re-fitting to get Hessian

# alpha  
  
Parameter | model.1 | modelA  
---------------------------------------------------------  
1 Low|2 Action | | 2.45 ( 1.32, 3.59)  
2 Action|3 AboveLimit | | 3.00 ( 1.79, 4.21)  
  
# beta  
  
Parameter | model.1 | modelA  
---------------------------------------------------------  
task [Insulation] | | 2.25 ( 0.97, 3.53)  
ventilation [General] | | 2.16 ( 1.03, 3.29)  
dur c | | -7.08e-04 (-0.01, 0.01)  
  
# Fixed Effects  
  
Parameter | model.1 | modelA  
---------------------------------------------------  
1 Low|2 Action | 0.17 (-0.27, 0.61) |   
2 Action|3 AboveLimit | 0.47 ( 0.02, 0.91) |

## Compare Performance

plot(compare\_performance(model.1, modelA))



## Classification Tables

* for modelA and model.1

addmargins(table(predict(modelA), asbestos$exposure,   
 dnn = c("predicted", "actual")))

actual  
predicted 1\_Low 2\_Action 3\_AboveLimit Sum  
 1\_Low 42 3 10 55  
 2\_Action 0 0 0 0  
 3\_AboveLimit 3 3 22 28  
 Sum 45 6 32 83

addmargins(table(predict(model.1), asbestos$exposure,   
 dnn = c("predicted", "actual")))

actual  
predicted 1\_Low 2\_Action 3\_AboveLimit Sum  
 1\_Low 45 6 32 83  
 2\_Action 0 0 0 0  
 3\_AboveLimit 0 0 0 0  
 Sum 45 6 32 83

## modelA vs. “No duration” Model

Compare to a model with just Task and Ventilation

modelTV <- polr(exposure ~ task + ventilation, data=asbestos)  
anova(modelA, modelTV) |>   
 gt() |> tab\_options(table.font.size = 20)

| Model | Resid. df | Resid. Dev | Test | Df | LR stat. | Pr(Chi) |
| --- | --- | --- | --- | --- | --- | --- |
| task + ventilation | 79 | 99.91421 |  | NA | NA | NA |
| task + ventilation + dur\_c | 78 | 99.87952 | 1 vs 2 | 1 | 0.03469476 | 0.8522367 |

AIC(modelA, modelTV)

df AIC  
modelA 5 109.8795  
modelTV 4 107.9142

BIC(modelA, modelTV)

df BIC  
modelA 5 121.9737  
modelTV 4 117.5896

## Classification Tables

* for modelA and modelTV

addmargins(table(predict(modelA), asbestos$exposure,   
 dnn = c("predicted", "actual")))

actual  
predicted 1\_Low 2\_Action 3\_AboveLimit Sum  
 1\_Low 42 3 10 55  
 2\_Action 0 0 0 0  
 3\_AboveLimit 3 3 22 28  
 Sum 45 6 32 83

addmargins(table(predict(modelTV), asbestos$exposure,   
 dnn = c("predicted", "actual")))

actual  
predicted 1\_Low 2\_Action 3\_AboveLimit Sum  
 1\_Low 42 3 10 55  
 2\_Action 0 0 0 0  
 3\_AboveLimit 3 3 22 28  
 Sum 45 6 32 83

## task\*ventilation interaction?

model.TxV <- polr(exposure ~ task \* ventilation, data=asbestos)  
anova(modelTV, model.TxV) |>   
 gt() |> tab\_options(table.font.size = 20)

| Model | Resid. df | Resid. Dev | Test | Df | LR stat. | Pr(Chi) |
| --- | --- | --- | --- | --- | --- | --- |
| task + ventilation | 79 | 99.91421 |  | NA | NA | NA |
| task \* ventilation | 78 | 99.64326 | 1 vs 2 | 1 | 0.2709469 | 0.6026973 |

AIC(modelTV, model.TxV)

df AIC  
modelTV 4 107.9142  
model.TxV 5 109.6433

BIC(modelTV, model.TxV)

df BIC  
modelTV 4 117.5896  
model.TxV 5 121.7375

## Fitting all of the models?

Well, not all of the models, but the interesting ones?

m1 <- polr(exposure ~ 1, data = asbestos)  
m2 <- polr(exposure ~ dur\_c, data = asbestos)  
m3 <- polr(exposure ~ task, data = asbestos)  
m4 <- polr(exposure ~ ventilation, data = asbestos)  
m5 <- polr(exposure ~ task + ventilation, data = asbestos)  
m6 <- polr(exposure ~ task \* ventilation, data = asbestos)  
m7 <- polr(exposure ~ task + ventilation + dur\_c, data = asbestos)  
  
anova(m2, m1)

Likelihood ratio tests of ordinal regression models  
  
Response: exposure  
 Model Resid. df Resid. Dev Test Df LR stat. Pr(Chi)  
1 1 81 147.6197   
2 dur\_c 80 142.2944 1 vs 2 1 5.325273 0.02101831

## asbestos Likelihood Ratio Tests

| Model | Elements | DF | Deviance | Test | *p* |
| --- | --- | --- | --- | --- | --- |
| 1 | Intercept | 81 | 147.62 | – | – |
| 2 | Duration | 80 | 142.29 | vs 1 | 0.021 |
| 3 | Task | 80 | 115.36 | vs 1 | < 0.0001 |
| 4 | Ventilation | 80 | 115.45 | vs 1 | < 0.0001 |
| 5 | T+V | 79 | 99.91 | vs 3 | < 0.0001 |
| 6 | T\*V | 78 | 99.64 | vs 5 | 0.603 |
| 7 | T+V+D | 78 | 99.88 | vs 5 | 0.852 |

## Predictions with our T+V model

modelTV <- polr(exposure ~ task + ventilation, data=asbestos)  
asbestos <- asbestos |> mutate(TV\_preds = predict(modelTV))  
asbestos |> tabyl(TV\_preds, exposure) |> adorn\_title()

exposure   
 TV\_preds 1\_Low 2\_Action 3\_AboveLimit  
 1\_Low 42 3 10  
 2\_Action 0 0 0  
 3\_AboveLimit 3 3 22

* Predicting Low exposure led to 42 right and 13 wrong.
* We never predicted Action Level
* Predicting Above Legal Limit led to 22 right and 6 wrong.

Total: 64 right, 19 wrong. Accuracy = 64/83 = 77.1%

## Proportional odds assumption reasonable?

Alternative: fit a multinomial model?

mult\_TV <- multinom(exposure ~ task + ventilation,   
 data = asbestos, trace = FALSE)  
mult\_TV

Call:  
multinom(formula = exposure ~ task + ventilation, data = asbestos,   
 trace = FALSE)  
  
Coefficients:  
 (Intercept) taskInsulation ventilationGeneral  
2\_Action -3.423661 1.159959 2.316383  
3\_AboveLimit -3.117443 2.699791 2.495969  
  
Residual Deviance: 98.08263   
AIC: 110.0826

## Multinomial T+V model predicts…

asbestos <- asbestos |>   
 mutate(TVmult\_preds = predict(mult\_TV))  
asbestos |> tabyl(TVmult\_preds, exposure) |> adorn\_title()

exposure   
 TVmult\_preds 1\_Low 2\_Action 3\_AboveLimit  
 1\_Low 42 3 10  
 2\_Action 0 0 0  
 3\_AboveLimit 3 3 22

* Exactly the same predictions as our polr model.

asbestos |> count(TVmult\_preds, TV\_preds)

# A tibble: 2 × 3  
 TVmult\_preds TV\_preds n  
 <fct> <fct> <int>  
1 1\_Low 1\_Low 55  
2 3\_AboveLimit 3\_AboveLimit 28

## Compare Models with Likelihood Ratio Test?

(LL\_multTV <- logLik(mult\_TV)) # multinomial model: 6 df

'log Lik.' -49.04131 (df=6)

(LL\_polrTV <- logLik(modelTV)) # polr model: 4 df

'log Lik.' -49.9571 (df=4)

(G = -2 \* (LL\_polrTV[1] - LL\_multTV[1]))

[1] 1.831584

pchisq(G, 2, lower.tail = FALSE)

[1] 0.4001995

*p* = 0.4 testing the difference in goodness of fit between the proportional odds model and the more complex multinomial logistic regression model.

## AIC and BIC for multinomial vs. polr models

AIC(mult\_TV, modelTV)

df AIC  
mult\_TV 6 110.0826  
modelTV 4 107.9142

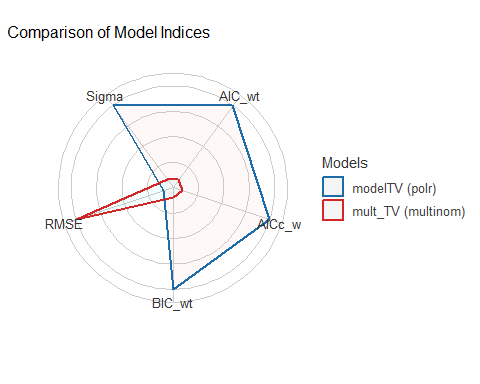
BIC(mult\_TV, modelTV)

df BIC  
mult\_TV 6 124.5957  
modelTV 4 117.5896

* mult\_TV is the multinomial model
* modelTV is the polr model

## Compare Performance

plot(compare\_performance(mult\_TV, modelTV))

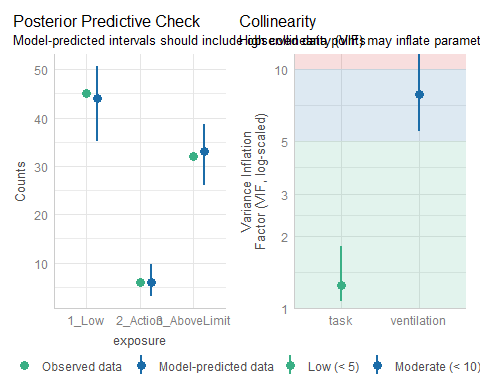


## check\_model() for POLR

check\_model(modelTV)

Re-fitting to get Hessian  
  
  
Re-fitting to get Hessian

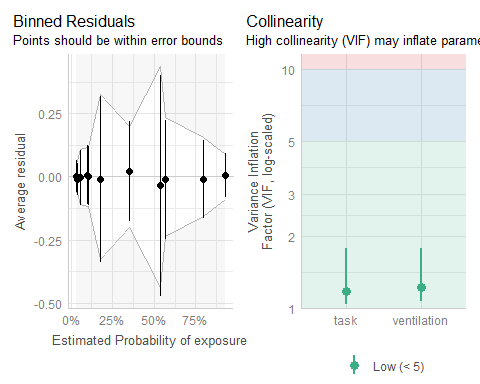
Cannot simulate residuals for models of class `polr`. Please try  
 `check\_model(..., residual\_type = "normal")` instead.



## check\_model() for Multinomial

check\_model(mult\_TV)

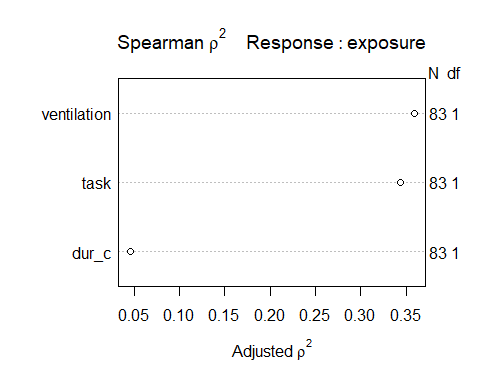
Cannot simulate residuals for models of class `multinom`. Please try  
 `check\_model(..., residual\_type = "normal")` instead.



# Using rms to fit the POLR model via lrm()

## Spearman ?

plot(spearman2(exposure ~ task + ventilation + dur\_c, data=asbestos))



## Proportional Odds Logistic Regression with lrm()

d <- datadist(asbestos)  
options(datadist = "d")  
  
# note that exposure must be an ordered factor  
  
model\_TV\_LRM <- lrm(exposure ~ task + ventilation,  
 data = asbestos, x = TRUE, y = TRUE)

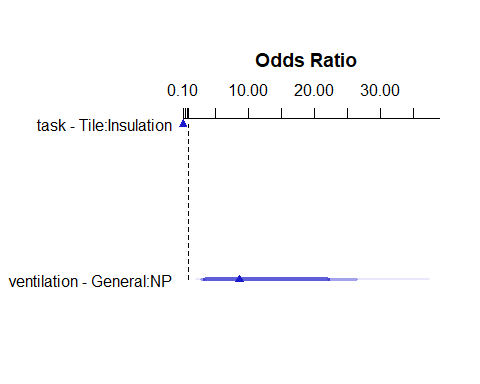
## The lrm() fit

model\_TV\_LRM

Logistic Regression Model  
  
lrm(formula = exposure ~ task + ventilation, data = asbestos,   
 x = TRUE, y = TRUE)  
  
 Model Likelihood Discrimination Rank Discrim.   
 Ratio Test Indexes Indexes   
Obs 83 LR chi2 47.71 R2 0.526 C 0.854   
 1\_Low 45 d.f. 2 R2(2,83) 0.423 Dxy 0.708   
 2\_Action 6 Pr(> chi2) <0.0001 R2(2,65) 0.505 gamma 0.839   
 3\_AboveLimit 32 Brier 0.127 tau-a 0.396   
max |deriv| 3e-10   
  
 Coef S.E. Wald Z Pr(>|Z|)  
y>=2\_Action -2.4751 0.5613 -4.41 <0.0001   
y>=3\_AboveLimit -3.0208 0.6005 -5.03 <0.0001   
task=Insulation 2.2868 0.6173 3.70 0.0002   
ventilation=General 2.1596 0.5675 3.81 0.0001

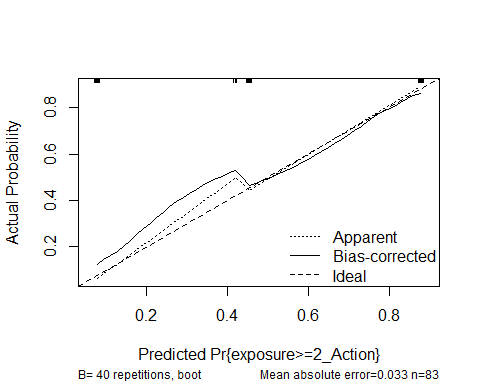
## Effects Plot after lrm()

plot(summary(model\_TV\_LRM))



## Calibrate lrm() fit?

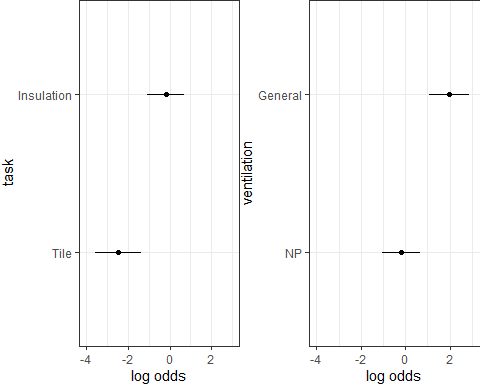
plot(calibrate(model\_TV\_LRM))



n=83 Mean absolute error=0.033 Mean squared error=0.00179  
0.9 Quantile of absolute error=0.048

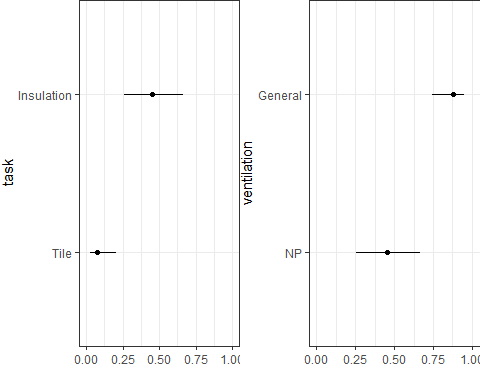
## lrm() fit, plotted on log odds scale

ggplot(Predict(model\_TV\_LRM), layout = c(1,2))



## lrm() fit, on probability scale

ggplot(Predict(model\_TV\_LRM, fun = plogis), layout = c(1,2))



## rms::validate results from lrm()

set.seed(432001)  
validate(model\_TV\_LRM)

index.orig training test optimism index.corrected n  
Dxy 0.7077 0.7194 0.7081 0.0112 0.6964 40  
R2 0.5260 0.5449 0.5210 0.0239 0.5021 40  
Intercept 0.0000 0.0000 0.0093 -0.0093 0.0093 40  
Slope 1.0000 1.0000 0.9414 0.0586 0.9414 40  
Emax 0.0000 0.0000 0.0149 0.0149 0.0149 40  
D 0.5627 0.5946 0.5554 0.0392 0.5235 40  
U -0.0241 -0.0241 -0.3980 0.3739 -0.3980 40  
Q 0.5868 0.6187 0.9534 -0.3347 0.9216 40  
B 0.1270 0.1205 0.1315 -0.0109 0.1379 40  
g 2.0639 2.1935 2.0426 0.1508 1.9131 40  
gp 0.3709 0.3755 0.3708 0.0047 0.3662 40

Validated = 0.5 + (0.6964/2) = 0.8482

## Model with Task-Ventilation Interaction

d <- datadist(asbestos)  
options(datadist = "d")  
  
# note that exposure must be an ordered factor  
  
model\_TxV\_LRM <- lrm(exposure ~ task \* ventilation,  
 data = asbestos, x = TRUE, y = TRUE)

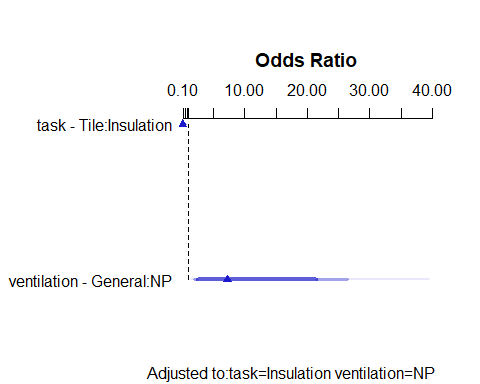
## model\_TxV\_LRM fit

model\_TxV\_LRM

Logistic Regression Model  
  
lrm(formula = exposure ~ task \* ventilation, data = asbestos,   
 x = TRUE, y = TRUE)  
  
 Model Likelihood Discrimination Rank Discrim.   
 Ratio Test Indexes Indexes   
Obs 83 LR chi2 47.98 R2 0.528 C 0.854   
 1\_Low 45 d.f. 3 R2(3,83) 0.418 Dxy 0.709   
 2\_Action 6 Pr(> chi2) <0.0001 R2(3,65) 0.499 gamma 0.840   
 3\_AboveLimit 32 Brier 0.127 tau-a 0.396   
max |deriv| 1e-07   
  
 Coef S.E. Wald Z Pr(>|Z|)  
y>=2\_Action -2.6808 0.7306 -3.67 0.0002   
y>=3\_AboveLimit -3.2245 0.7588 -4.25 <0.0001   
task=Insulation 2.5851 0.8713 2.97 0.0030   
ventilation=General 2.6205 1.0646 2.46 0.0138   
task=Insulation \* ventilation=General -0.6453 1.2471 -0.52 0.6048

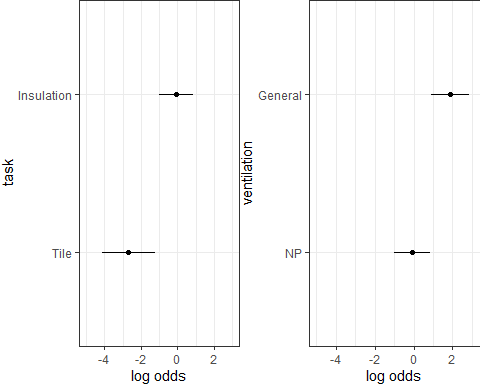
## Effects Plot: model\_TxV\_LRM

plot(summary(model\_TxV\_LRM))



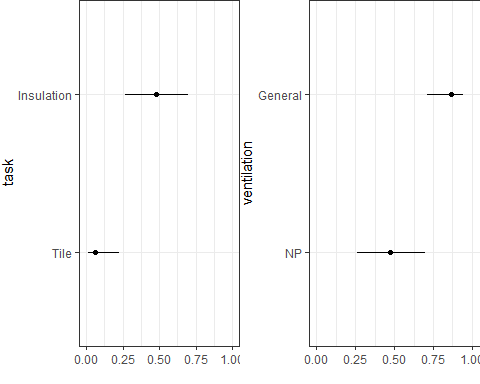
## model\_TxV\_LRM on log odds scale

ggplot(Predict(model\_TxV\_LRM), layout = c(1,2))



## model\_TxV\_LRM on probability scale

ggplot(Predict(model\_TxV\_LRM, fun = plogis), layout = c(1,2))



## Calibrate or Validate?

produces…

Error in predab.resample(fit, method = method, fit = fitit, measure = cal.error, :   
 A training sample has a different number of intercepts (1)  
than the original model fit (2).  
You probably fit an ordinal model with sparse cells and a re-sample  
did not select at least one observation for each value of Y.  
Add the argument group=y where y is the response variable.  
This will force balanced sampling on levels of y.

## All possible combinations of T and V

newdat <- data.frame(  
 worker = c("New1", "New2", "New3", "New4"),  
 task = c("Tile", "Tile", "Insulation", "Insulation"),  
 ventilation = c("NP", "General", "NP", "General")  
) |>  
 mutate(task = factor(task),   
 ventilation = factor(ventilation))  
  
newdat ## note this is NOT a tibble

worker task ventilation  
1 New1 Tile NP  
2 New2 Tile General  
3 New3 Insulation NP  
4 New4 Insulation General

## Add individual predictions

We use predict() with type = "fitted.ind" here.

newdat\_aug <- cbind(newdat,   
 predict(model\_TV\_LRM, newdata = newdat, type = "fitted.ind"))  
  
newdat\_aug |> gt() |> fmt\_number(decimals = 3) |>  
 tab\_options(table.font.size = 20)

| worker | task | ventilation | exposure=1\_Low | exposure=2\_Action | exposure=3\_AboveLimit |
| --- | --- | --- | --- | --- | --- |
| New1 | Tile | NP | 0.922 | 0.031 | 0.046 |
| New2 | Tile | General | 0.578 | 0.125 | 0.297 |
| New3 | Insulation | NP | 0.547 | 0.129 | 0.324 |
| New4 | Insulation | General | 0.122 | 0.072 | 0.806 |

## Instead add fitted predictions?

Using type = "fitted" produces greater than or equal to predictions instead.

newdat\_aug2 <- cbind(newdat,   
 predict(model\_TV\_LRM, newdata = newdat, type = "fitted"))  
  
newdat\_aug2 |> gt() |> fmt\_number(decimals = 3) |>  
 tab\_options(table.font.size = 20)

| worker | task | ventilation | y>=2\_Action | y>=3\_AboveLimit |
| --- | --- | --- | --- | --- |
| New1 | Tile | NP | 0.078 | 0.046 |
| New2 | Tile | General | 0.422 | 0.297 |
| New3 | Insulation | NP | 0.453 | 0.324 |
| New4 | Insulation | General | 0.878 | 0.806 |

# Ordinal Logistic Regression with orm() from rms

## orm() vs. lrm() differences?

* In fitting an orm() vs. lrm(), just using the letter “o” instead of “l”.
* The orm() model: appropriate when we are interested in studying the rank correlation between the predictions and the outcomes - in essence, we are interested in “penalizing” more for being two categories away from correct than being one category away from correct.
* The lrm() or polr() model: appropriate when we are interested in “penalizing” all incorrect predictions the same way.

## Ordinal Logistic Regression for T+V with orm

d <- datadist(asbestos)  
options(datadist = "d")  
  
model\_TV\_ORM <- orm(exposure ~ task + ventilation,  
 data = asbestos, x = TRUE, y = TRUE)  
  
# note that exposure must be an ordered factor

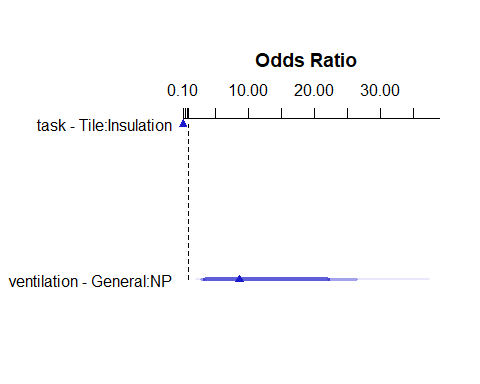
## model\_TV\_ORM fit with orm

model\_TV\_ORM

Logistic (Proportional Odds) Ordinal Regression Model  
  
orm(formula = exposure ~ task + ventilation, data = asbestos,   
 x = TRUE, y = TRUE)  
  
 Model Likelihood Discrimination Rank Discrim.   
 Ratio Test Indexes Indexes   
Obs 83 LR chi2 47.71 R2 0.526 rho 0.697   
 1\_Low 45 d.f. 2 R2(2,83) 0.423   
 2\_Action 6 Pr(> chi2) <0.0001 R2(2,65) 0.505   
 3\_AboveLimit 32 Score chi2 42.42 |Pr(Y>=median)-0.5| 0.301   
Distinct Y 3 Pr(> chi2) <0.0001   
Median Y 1   
max |deriv| 3e-10   
  
 Coef S.E. Wald Z Pr(>|Z|)  
y>=2\_Action -2.4751 0.5613 -4.41 <0.0001   
y>=3\_AboveLimit -3.0208 0.6005 -5.03 <0.0001   
task=Insulation 2.2868 0.6173 3.70 0.0002   
ventilation=General 2.1596 0.5675 3.81 0.0001

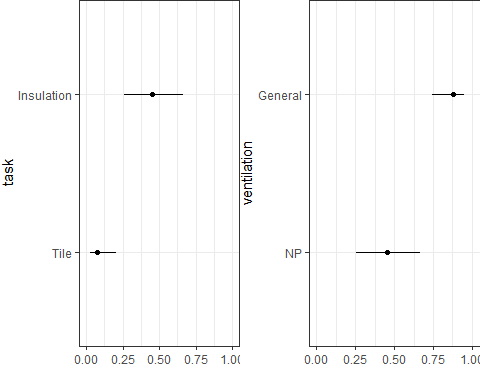
## Effects Plot from orm

plot(summary(model\_TV\_ORM))



## orm model fit, plotted

ggplot(Predict(model\_TV\_ORM, fun = plogis), layout = c(1,2))



## rms::validate results from orm

set.seed(432002)  
validate(model\_TV\_ORM)

index.orig training test optimism index.corrected n  
rho 0.6970 0.6969 0.6975 -0.0006 0.6976 40  
R2 0.5260 0.5308 0.5157 0.0151 0.5109 40  
Slope 1.0000 1.0000 0.9526 0.0474 0.9526 40  
g 2.0639 2.2598 2.0132 0.2467 1.8172 40  
pdm 0.3010 0.3103 0.3052 0.0051 0.2959 40

* rho = Spearman’s rank correlation between linear predictor and outcome
* R2 = Nagelkerke R-square

## Predicting with orm()

We can from the information below, estimate the model probability of obtaining each of the three possible results.

newdat\_aug3 <- cbind(newdat,   
 predict(model\_TV\_ORM, newdata = newdat, type = "fitted"))  
  
newdat\_aug2 |> gt() |> fmt\_number(decimals = 3) |>  
 tab\_options(table.font.size = 20)

| worker | task | ventilation | y>=2\_Action | y>=3\_AboveLimit |
| --- | --- | --- | --- | --- |
| New1 | Tile | NP | 0.078 | 0.046 |
| New2 | Tile | General | 0.422 | 0.297 |
| New3 | Insulation | NP | 0.453 | 0.324 |
| New4 | Insulation | General | 0.878 | 0.806 |

## Conclusions?

We can fit both POLR models and ordinal regression models with rms approaches, and we can also fit POLR with MASS::polr().

* All are designed for *ordinal* multi-categorical outcomes.
* Can compare results to what we would get with multinomial models, designed for *nominal* multi-categorical outcomes.

We’ll focus on regression for *nominal* multi-categorical outcomes next time.

1. Simonoff JS (2003) *Analyzing Categorical Data*. Chapter 10. [↑](#footnote-ref-22)
2. Simonoff JS (2003) *Analyzing Categorical Data*. Chapter 10. [↑](#footnote-ref-24)
3. Simonoff JS (2003) *Analyzing Categorical Data*. Chapter 10. [↑](#footnote-ref-26)