Multinomial Logistic Regression

Assumptions & Model Selection

Prof. Maria Tackett

04.08.20

Click for PDF of slides

Checking assumptions

Assumptions for multinomial logistic regression

We want to check the following assumptions for the multinomial logistic regression model:

- Linearity: Is there a linear relationship between the log-odds and the predictor variables?

- Randomness: Was the sample randomly selected? Or can we reasonably treat it as random?

- Independence: There is no obvious relationship between observations

Checking linearity

For each category of the response, j:

- Analyze a plot of the binned residuals vs. predicted probabilities
- Analyze a plot of the binned residuals vs. each continuous predictor variable
- Look for any patterns in the residuals plots
- For each categorical predictor variables, examine the average residuals for each category of the predictor

Randomness

Assess randomness based on a description of the data collection

- Was the sample randomly selected?
- If the sample was not randomly selected, is there reason to believe the observations in the sample differ systematically from the population of interest?

Independence

Assess independence based on a description of the data collection

- Is there an obvious relationship between observations?
  - This assumption is most often violated when data was collected over time or there is a spatial relationship between observations?

NHANES Data

- **Question**: Can we use a person's age and whether they do regular physical activity to predict their self-reported health rating?
- Variables:
  - **`HealthGen`:** Self-reported rating of participant's health in general. Excellent, Vgood, Good, Fair, or Poor.
  - **`Age`:** Age at time of screening (in years). Participants 80 or older were recorded as 80.
  - **`PhysActive`:** Participant does moderate to vigorous-intensity sports, fitness or recreational activities
  - **`Education`**: Categorical variable for highest education level attained

Model

y.level	term	estimate	std.error	statistic	p.value	conf.low	conf.high
Vgood	(Intercept)	1.205	0.145	8.325	0.000	0.922	1.489
Vgood	Age	0.001	0.002	0.369	0.712	-0.004	0.006
Vgood	PhysActiveYes	-0.321	0.093	-3.454	0.001	-0.503	-0.139
Good	(Intercept)	1.948	0.141	13.844	0.000	1.672	2.223
Good	Age	-0.002	0.002	-0.977	0.329	-0.007	0.002
Good	PhysActiveYes	-1.001	0.090	-11.120	0.000	-1.178	-0.825
Fair	(Intercept)	0.915	0.164	5.566	0.000	0.592	1.237
Fair	Age	0.003	0.003	1.058	0.290	-0.003	0.009
Fair	PhysActiveYes	-1.645	0.107	-15.319	0.000	-1.856	-1.435
Poor	(Intercept)	-1.521	0.290	-5.238	0.000	-2.090	-0.952
Poor	Age	0.022	0.005	4.522	0.000	0.013	0.032
Poor	PhysActiveYes	-2.656	0.236	-11.275	0.000	-3.117	-2.194

NHANES: Residuals

```
#calculate residuals
residuals <- as_tibble(residuals(health_m)) %>%  #calculate residu
  setNames(paste('resid.', names(.), sep = "")) %>% #update columr
  mutate(obs_num = 1:n()) #add obs number
```

```
residuals %>%
  slice(1:10)
```

```
## # A tibble: 10 x 6
##    resid.Excellent resid.Vgood resid.Good resid.Fair resid.Poor obs_num
##              <dbl>       <dbl>      <dbl>      <dbl>      <dbl>   <int>
##  1         -0.0707      -0.243      0.543    -0.196    -0.0328        1
##  2         -0.0707      -0.243      0.543    -0.196    -0.0328        2
##  3         -0.0707      -0.243      0.543    -0.196    -0.0328        3
##  4         -0.0700      -0.244      0.563    -0.203    -0.0454        4
##  5         -0.155        0.608     -0.360    -0.0859   -0.00648       5
##  6         -0.155        0.608     -0.360    -0.0859   -0.00648       6
##  7         -0.155        0.608     -0.360    -0.0859   -0.00648       7
##  8         -0.156        0.600     -0.343    -0.0916   -0.0103        8
##  9         -0.156        0.603     -0.349    -0.0894   -0.00865       9
## 10         -0.156       -0.396     -0.353     0.912    -0.00791      10
```

Make "augmented" dataset

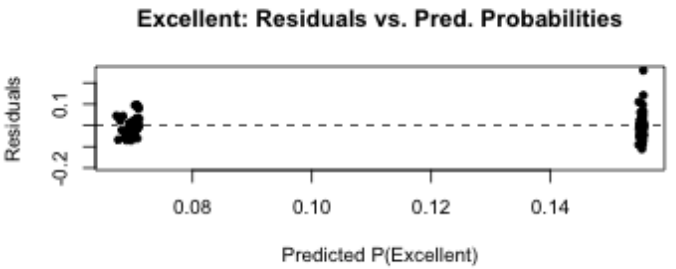
```
#calculate predicted probabilities
pred_probs <- as_tibble(predict(health_m, type = "probs")) %>%
  mutate(obs_num = 1:n())
```

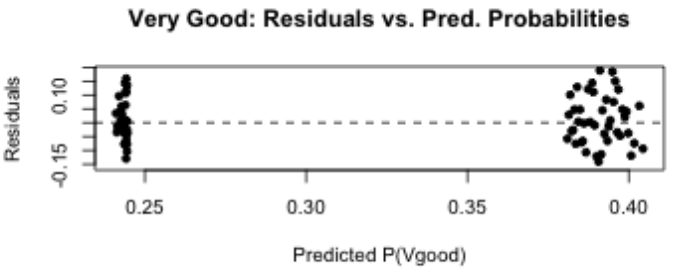
```
health_m_aug <- inner_join(nhanes_adult, pred_probs) #add probs
health_m_aug <- inner_join(health_m_aug, residuals) #add resid
```

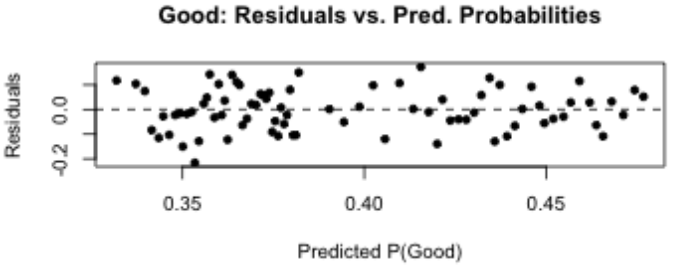
```
health_m_aug %>%
  glimpse()
```

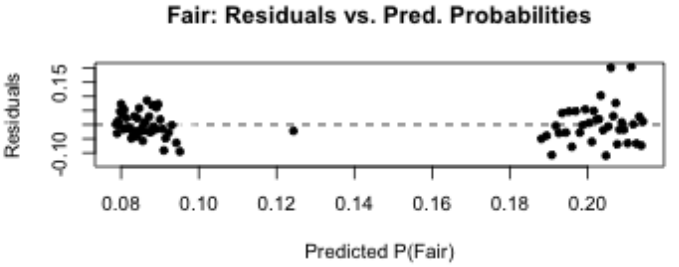
```
## Rows: 6,710
## Columns: 15
## $ HealthGen       <fct> Good, Good, Good, Good, Vgood, Vgood, Vgood, Vgo
## $ Age             <int> 34, 34, 34, 49, 45, 45, 45, 66, 58, 54, 50, 33,
## $ PhysActive      <fct> No, No, No, No, Yes, Yes, Yes, Yes, Yes, Yes, Ye
## $ Education       <fct> High School, High School, High School, Some Coll
## $ obs_num         <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 1
## $ Excellent       <dbl> 0.07069715, 0.07069715, 0.07069715, 0.07003173,
## $ Vgood           <dbl> 0.2433979, 0.2433979, 0.2433979, 0.2444214, 0.39
## $ Good            <dbl> 0.4573727, 0.4573727, 0.4573727, 0.4372533, 0.35
## $ Fair            <dbl> 0.19568909, 0.19568909, 0.19568909, 0.20291032,
```

Binned residuals vs. pred. probabilities

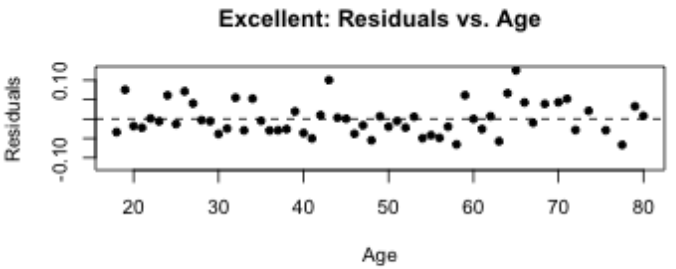


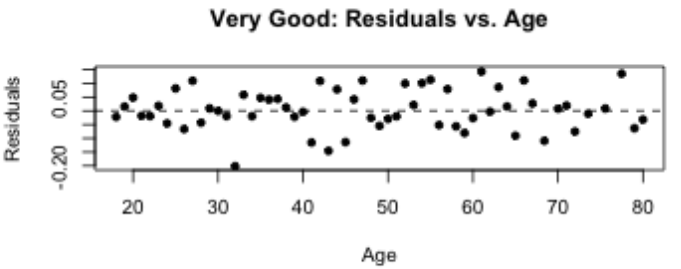


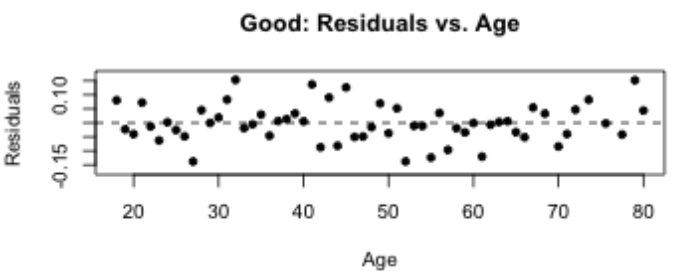


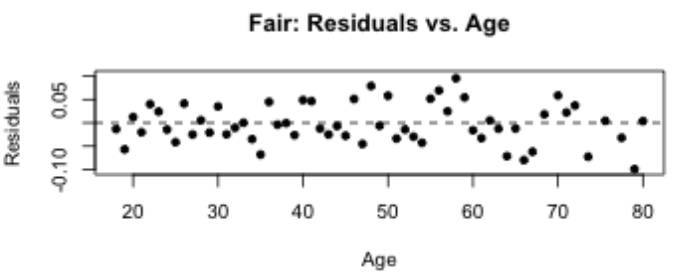


Binned residuals vs. Age









Residuals vs. `PhysActive`

```
health_m_aug %>%
  group_by(PhysActive) %>%
  summarise(mean.Excellent = mean(resid.Excellent),
            mean.Vgood = mean(resid.Vgood),
            mean.Good = mean(resid.Good),
            mean.Fair = mean(resid.Fair),
            mean.Poor = mean(resid.Poor)) %>%
  t()
```

```
##                [,1]            [,2]
## PhysActive     "No"            "Yes"
## mean.Excellent "-1.194022e-07" " 2.106514e-06"
## mean.Vgood     " 1.644794e-06" "-1.871461e-06"
## mean.Good      "-3.227820e-06" " 1.140886e-07"
## mean.Fair      " 1.333924e-06" "-3.860756e-07"
## mean.Poor      "3.685045e-07"  "3.693412e-08"
```

Randomness & Independence

- Randomness:
  - About the R dataset: *"NHANES can be treated, for educational purposes, as if it were a simple random sample from the American population."*
  - What about the actual NHANES data? Type `?NHANES` in the console to read more details about how participants are selected for the actual survey.
- Independence: The participants are randomly selected within subpopulations, so the independence assumption is satisfied.

Model Selection

Comparing Nested Models

- Suppose there are two models:
  - Reduced Model includes predictors $x_1, \ldots, x_q$
  - Full Model includes predictors $x_1, \ldots, x_q, x_{q+1}, \ldots, x_p$
- We want to test the hypotheses

$$H_0 : \beta_{q+1} = \cdots = \beta_p = 0$$
$$H_a : \text{ at least 1 } \beta_j \text{ is not} 0$$

- To do so, we will use the Drop-in-Deviance test (very similar to logistic regression)

Add `Education` to the model?

- We consider adding the participants' `Education` level to the model.
  - Education takes values `8thGrade`, `9-11thGrade`, `HighSchool`, `SomeCollege`, and `CollegeGrad`
- Models we're testing:
  - Reduced Model: `Age`, `PhysActive`
  - Full Model: `Age`, `PhysActive`, `Education`

$$H_0 : \beta_{9-11thGrade} = \beta_{HighSchool} = \beta_{SomeCollege} = \beta_{CollegeGrad} = 0$$
$$H_a : \text{ at least one } \beta_j \text{ is not equal to } 0$$

Add `Education` to the model?

$$H_0 : \beta_{9-11thGrade} = \beta_{HighSchool} = \beta_{SomeCollege} = \beta_{CollegeGrad} = 0$$
$$H_a : \text{ at least one } \beta_j \text{ is not equal to } 0$$

```
model_red <- multinom(HealthGen ~ Age + PhysActive,
                      data = nhanes_adult)
model_full <- multinom(HealthGen ~ Age + PhysActive + Education,
                      data = nhanes_adult)
```

Add `Education` to the model?

```
model_red <- multinom(HealthGen ~ Age + PhysActive,
                      data = nhanes_adult)
model_full <- multinom(HealthGen ~ Age + PhysActive + Education,
                      data = nhanes_adult)
```

```
anova(model_red, model_full, test = "Chisq") %>%
  kable(format = "markdown")
```

Model	Resid. df	Resid. Dev	Test	Df	LR stat.	Pr(Chi)
Age + PhysActive	25848	16994.23		NA	NA	NA
Age + PhysActive + Education	25832	16505.10	1 vs 2	16	489.1319	0

At least one coefficient associated with `Education` is non-zero. Therefore, `Education` is a statistically significant predictor for `HealthGen`.

Model with `Education`

y.level	term	estimate	std.error	statistic	p.value	conf.low	conf.high
Vgood	(Intercept)	0.582	0.301	1.930	0.054	-0.009	1.173
Vgood	Age	0.001	0.003	0.419	0.675	-0.004	0.006
Vgood	PhysActiveYes	-0.264	0.099	-2.681	0.007	-0.457	-0.071
Vgood	Education9 - 11th Grade	0.768	0.308	2.493	0.013	0.164	1.372
Vgood	EducationHigh School	0.701	0.280	2.509	0.012	0.153	1.249
Vgood	EducationSome College	0.788	0.271	2.901	0.004	0.256	1.320
Vgood	EducationCollege Grad	0.408	0.268	1.522	0.128	-0.117	0.933
Good	(Intercept)	2.041	0.272	7.513	0.000	1.508	2.573
Good	Age	-0.002	0.003	-0.651	0.515	-0.007	0.003
Good	PhysActiveYes	-0.758	0.096	-7.884	0.000	-0.946	-0.569
Good	Education9 - 11th Grade	0.360	0.275	1.310	0.190	-0.179	0.899
Good	EducationHigh School	0.085	0.247	0.345	0.730	-0.399	0.569
Good	EducationSome College	-0.011	0.239	-0.047	0.962	-0.480	0.457
Good	EducationCollege Grad	-0.891	0.236	-3.767	0.000	-1.354	-0.427

Compare NHANES models using AIC

```
glance(model_red)
```

```
## # A tibble: 1 x 3
##     edf deviance    AIC
##   <dbl>    <dbl>  <dbl>
## 1    12   16994. 17018.
```

```
glance(model_full)
```

```
## # A tibble: 1 x 3
##     edf deviance    AIC
##   <dbl>    <dbl>  <dbl>
## 1    28   16505. 16561.
```

- Use the `step()` function to do model selection with AIC as the selection criteria

What questions do you have about multinomial logistic regression?