432 Week 3 Slides

github.com/THOMASELOVE/2020-432

2020-01-28 & 01-30

This Week's Agenda

1. Predicting a binary outcome using linear probability models.
2. Predicting a binary outcome with logistic regression

Setup

```
library(here); library(magrittr); library(janitor)
library(broom); library(simputation); library(patchwork)
library(naniar); library(visdat)
library(tidyverse)

theme_set(theme_bw())

smart1 <- readRDS(here("data/smart1.Rds"))
smart1_sh <- readRDS(here("data/smart1_sh.Rds"))
```

smart1_sh Variables, by Type

Variable	Type	Description
landline	Binary (1/0)	survey conducted by landline? (vs. cell)
healthplan	Binary (1/0)	subject has health insurance?
age_imp	Quantitative	age (imputed from groups - see Notes)
fruit_day	Quantitative	mean servings of fruit / day
drinks_wk	Quantitative	mean alcoholic drinks / week
bmi	Quantitative	body-mass index (in kg/m^2)
physhealth	Count (0-30)	of last 30 days, # in poor physical health
dm_status	Categorical	diabetes status (now 2 levels)
activity	Categorical	physical activity level (4 levels)
smoker	Categorical	tobacco use status (now 3 levels)
genhealth	Categorical	self-reported overall health (5 levels)

Today's Questions

Can we predict Prob(BMI < 30) for a subject in the `smart1_sh` data:

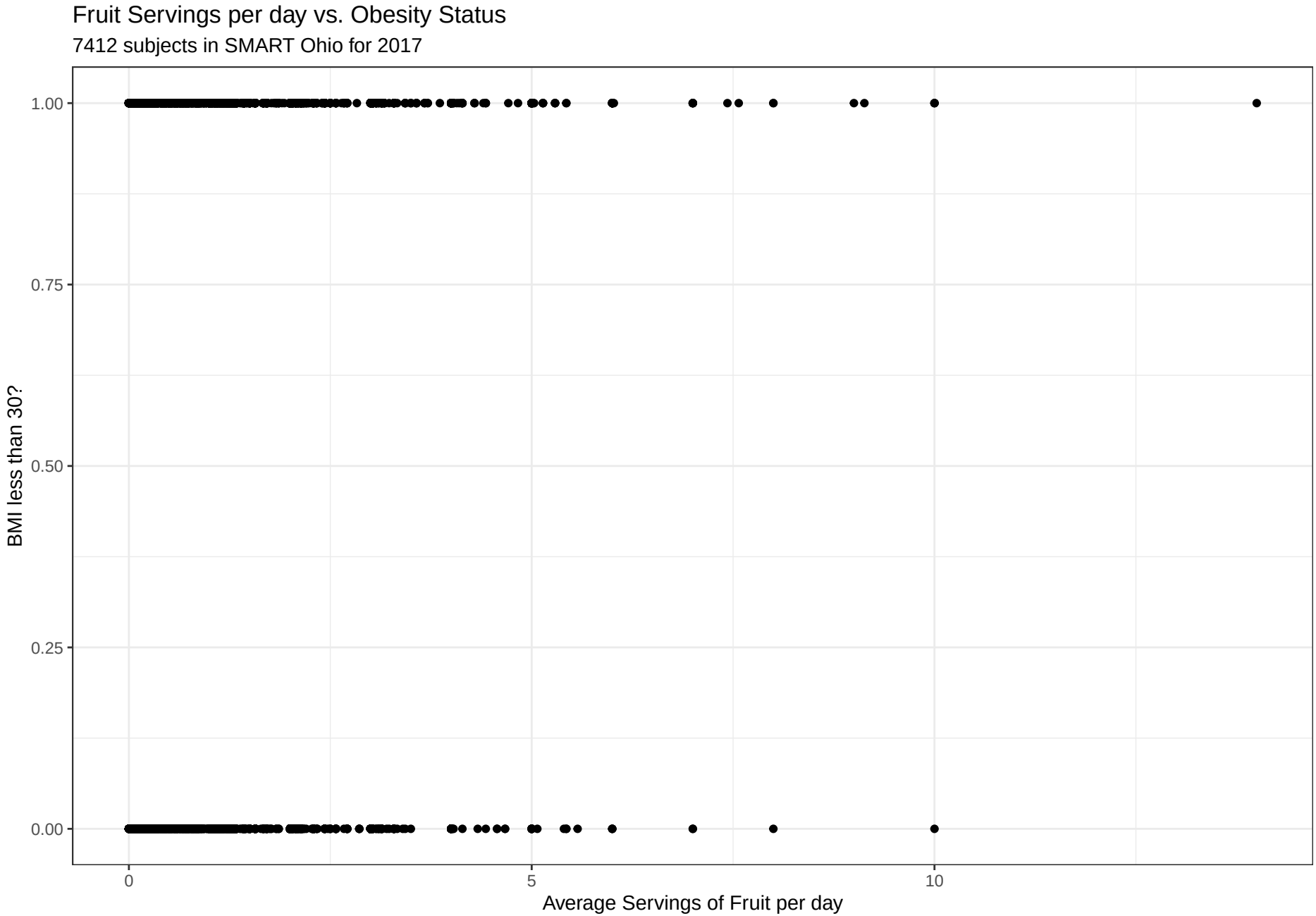
- using the mean number of servings of fruit per day that they consume?
- using their diabetes status?
- using their self-reported general health status?
- using some combination of these predictors?

Let's predict the probability that BMI < 30

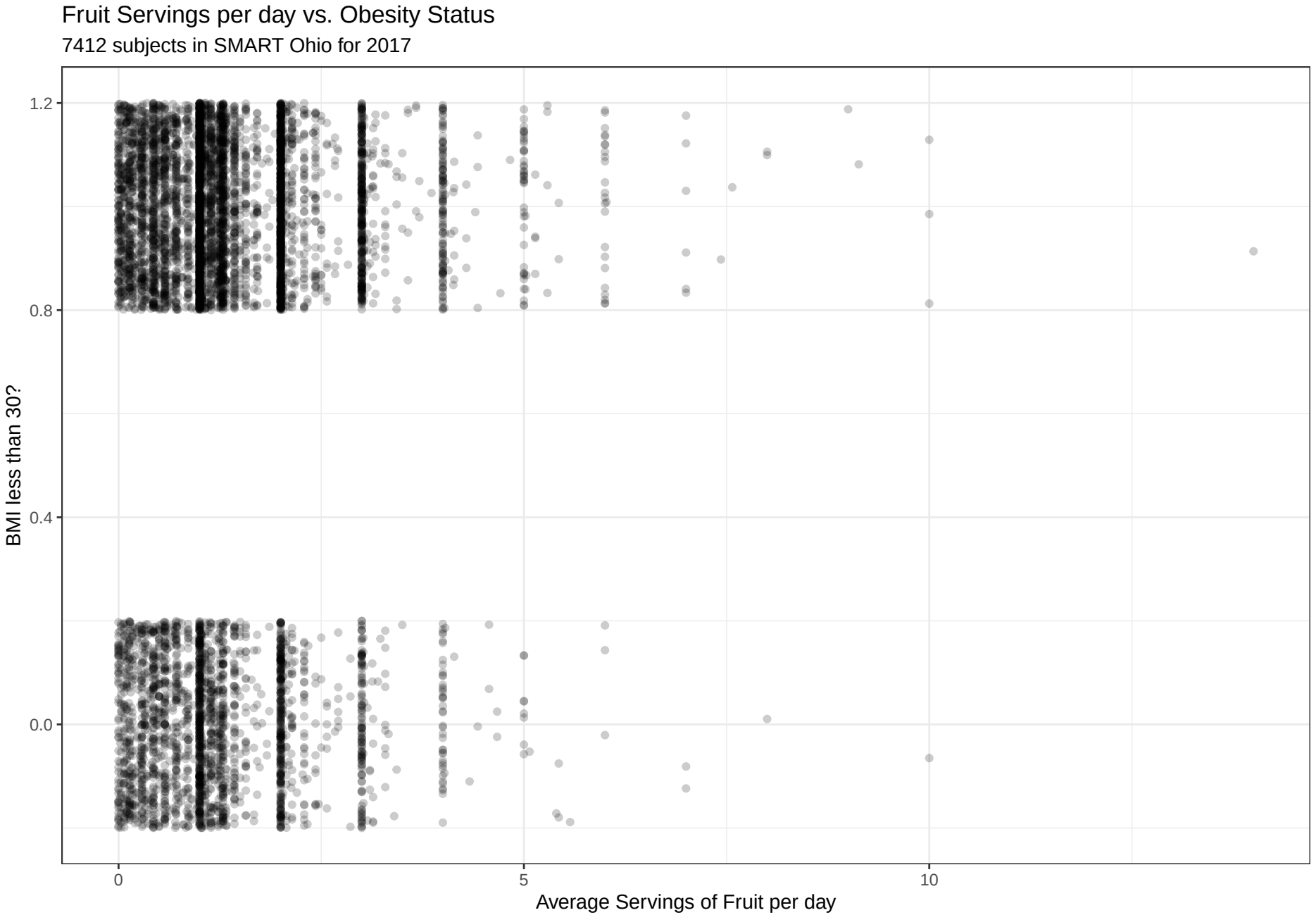
```
smart1_sh <- smart1_sh %>%
  mutate(bmilt30 = as.numeric(bmi < 30),
         dm_status = fct_relevel(dm_status, "No"))

smart1_sh %>% tabyl(bmilt30) %>% adorn_pct_formatting()
```

```
 bmilt30    n percent
       0 2343   31.6%
       1 5069   68.4%
```

Association of BMI $<$ 30 and Fruit Consumption



Add some vertical jitter and shading to the plot



Linear Probability Model to predict BMI < 30?

```
m1 <- smart1_sh %$% lm(bmilt30 ~ fruit_day)

tidy(m1, conf.int = TRUE, conf.level = 0.95) %>%
  select(term, estimate, std.error, conf.low, conf.high) %>%
  knitr::kable(digits = 3)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	0.645	0.009	0.628	0.662
fruit_day	0.029	0.005	0.019	0.039

Linear Probability Model to predict BMI < 30?

```
tidy(m1, conf.int = TRUE, conf.level = 0.95) %>%
  select(term, estimate, std.error, conf.low, conf.high) %>%
  knitr::kable(digits = 3)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	0.645	0.009	0.628	0.662
fruit_day	0.029	0.005	0.019	0.039

- What's the predicted probability of BMI < 30 if a subject eats 5 servings of fruit per day?

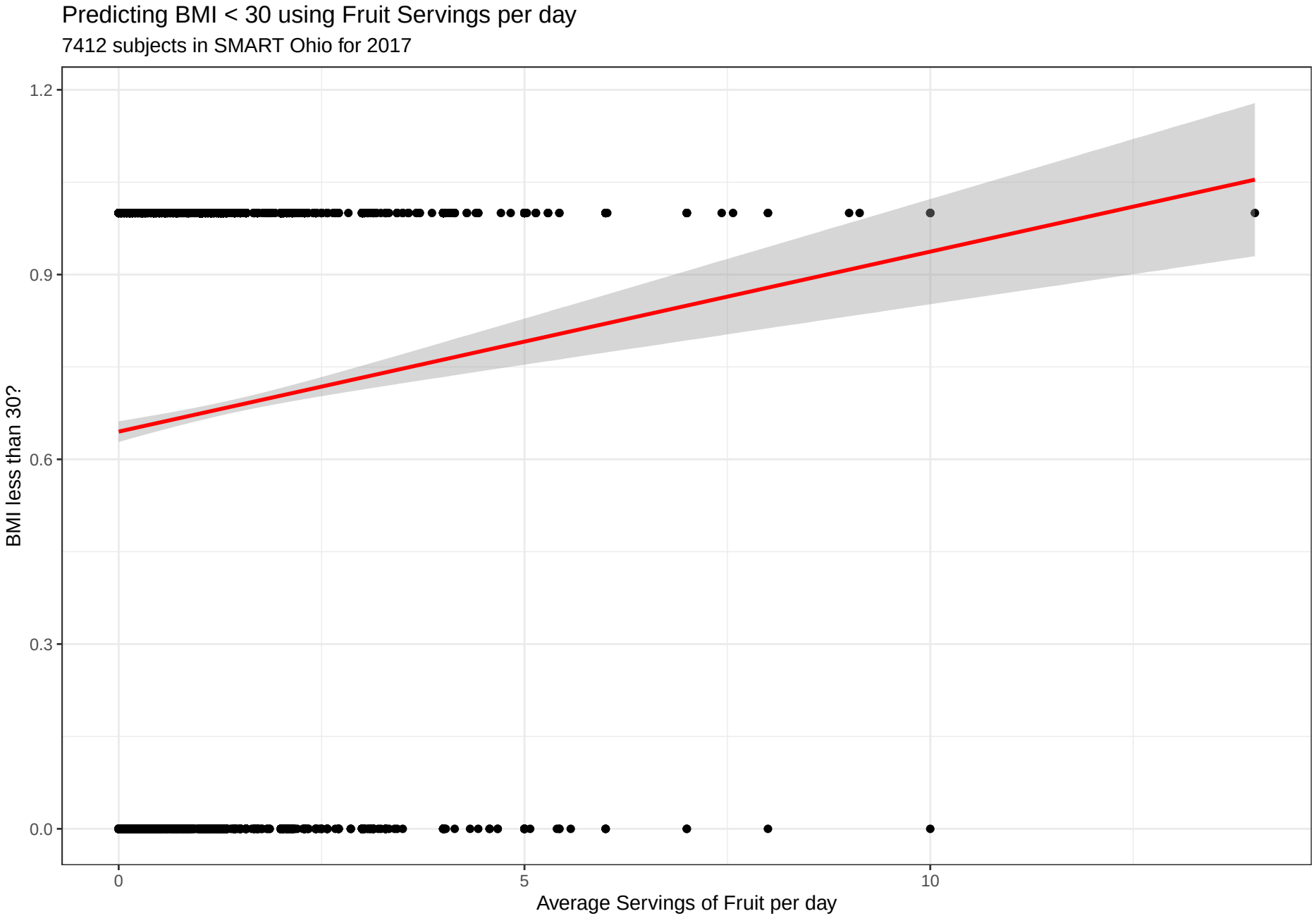
$$Pr(BMI < 30) = 0.645 + 0.029(5) = 0.645 + 0.145 = 0.790$$

- What's the predicted probability of BMI < 30 if a subject eats no fruit?

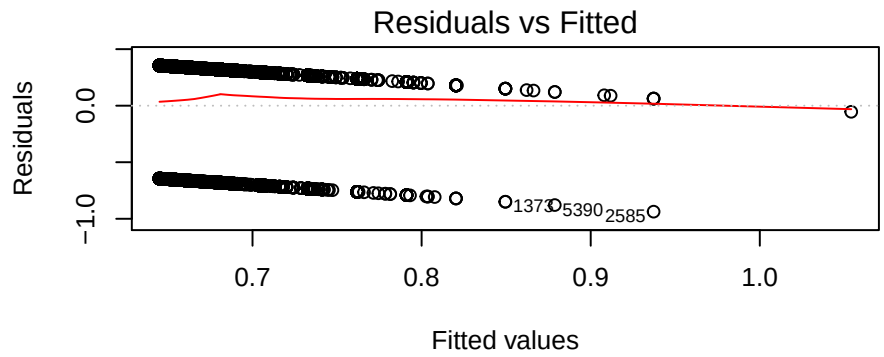
Linear Probability Model `m1` in a plot (code)

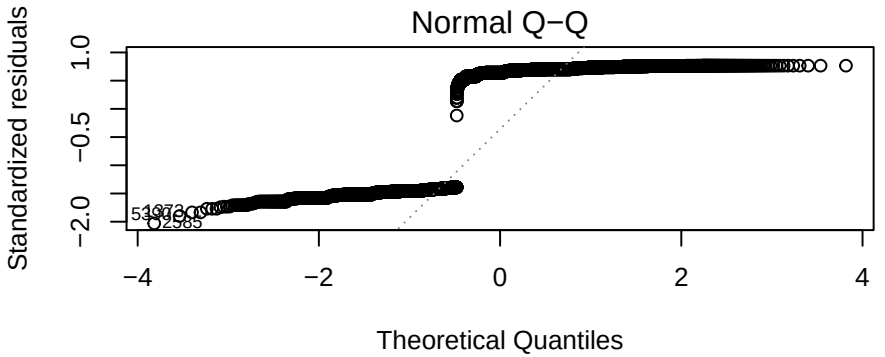
```
ggplot(smart1_sh, aes(x = fruit_day, y = bmilt30)) +
  geom_point() +
  geom_smooth(method = "lm", col = "red") +
  labs(title = "Predicting BMI < 30 using Fruit Servings per d
       subtitle = "7412 subjects in SMART Ohio for 2017",
       y = "BMI less than 30?",
       x = "Average Servings of Fruit per day")
```

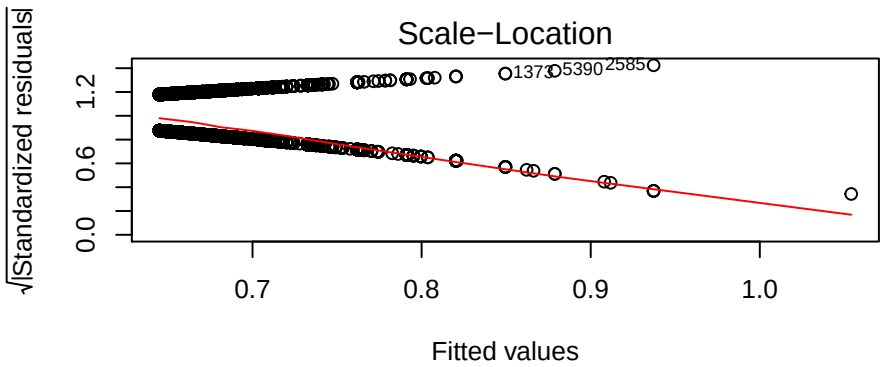
Linear Probability Model `m1` predicting BMI $<$ 30

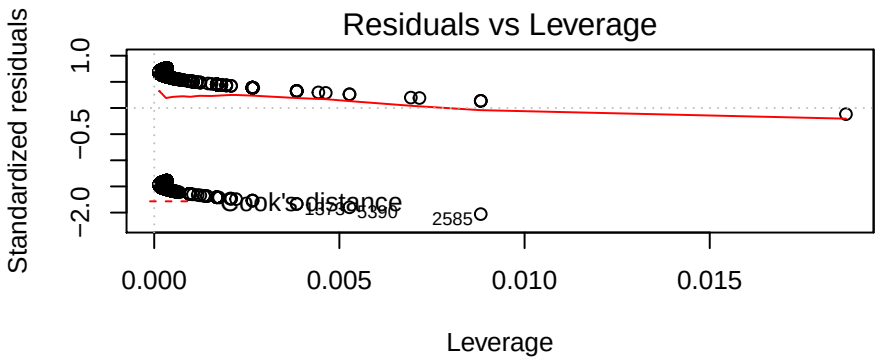


Residual Plots for the Linear Probability Model (m1)









Models to predict a Binary Outcome

Our outcome takes on two values (zero or one) and we then model the probability of a "one" response given a linear function of predictors.

Idea 1: Use a *linear probability model*

- Main problem: predicted probabilities that are less than 0 and/or greater than 1
- Also, how can we assume Normally distributed residuals when outcomes are 1 or 0?

Idea 2: Build a *non-linear* regression approach

- Most common approach: logistic regression, part of the class of *generalized* linear models

The Logit Link and Logistic Function

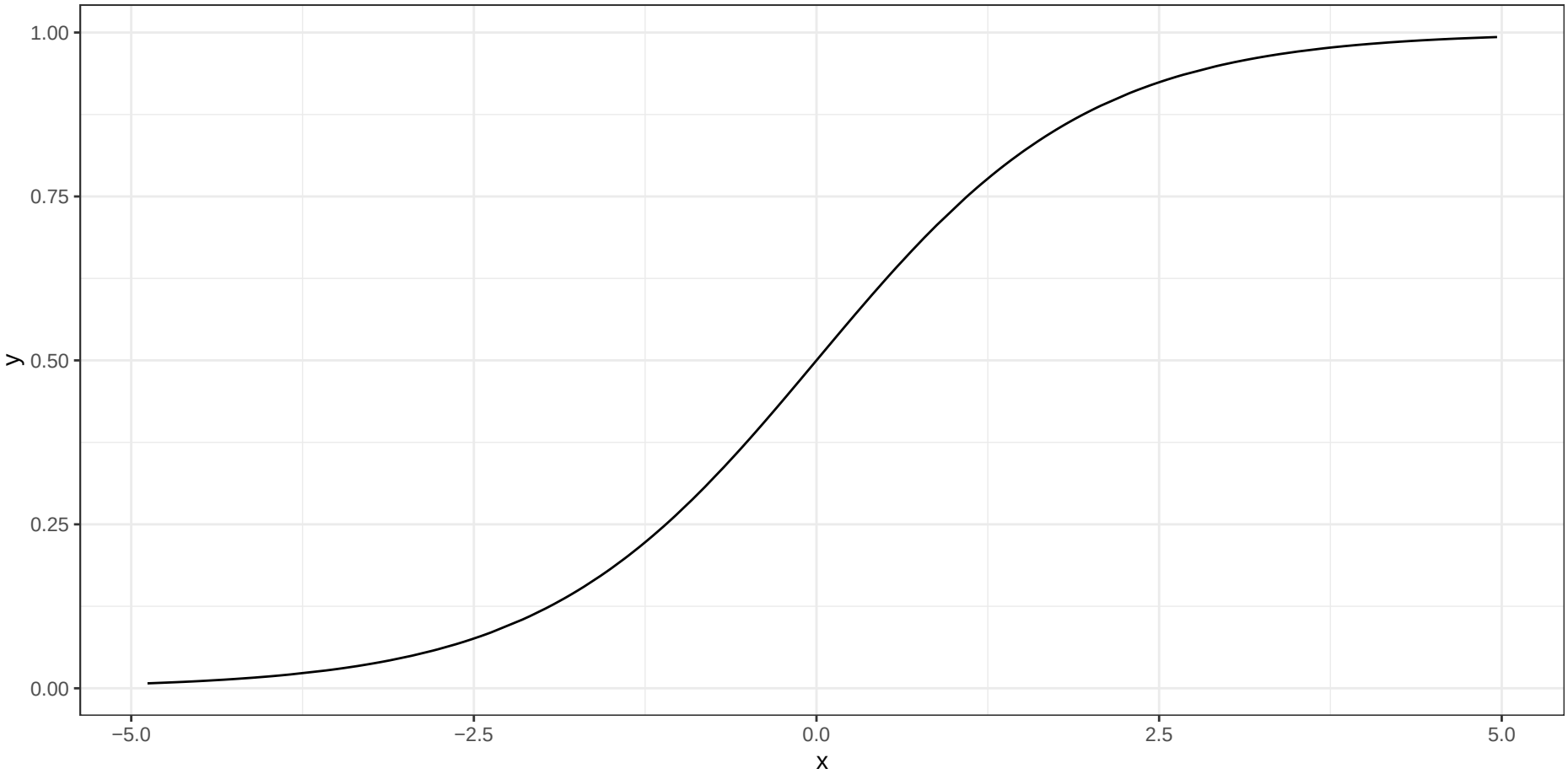
The particular link function we use in logistic regression is called the **logit link**.

$$logit(\pi) = log\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_k X_k$$

The inverse of the logit function is called the **logistic function**. If logit(π) $= \eta$, then $\pi = \frac{exp(\eta)}{1+exp(\eta)}$.

- The logistic function $\frac{e^x}{1+e^x}$ takes any value x in the real numbers and returns a value between 0 and 1.

The Logistic Function $y = \frac{e^x}{1+e^x}$



The logit or log odds

We usually focus on the **logit** in statistical work, which is the inverse of the logistic function.

- If we have a probability $\pi < 0.5$, then $logit(\pi) < 0$.
- If our probability $\pi > 0.5$, then $logit(\pi) > 0$.
- Finally, if $\pi = 0.5$, then $logit(\pi) = 0$.

Why is this helpful?

- log(odds(Y = 1)) or logit(Y = 1) covers all real numbers.
- Prob(Y = 1) is restricted to [0, 1].

Returning to the prediction of Prob(BMI $<$ 30)

We'll use the `glm` function in R, specifying a logistic regression model.

- Instead of predicting $Pr(BMI < 30)$, we're predicting $log(odds(BMI < 30))$ or $logit(BMI < 30)$.

```
m2 <- smart1_sh %$%
  glm(bmilt30 ~ fruit_day, family = binomial)

tidy(m2, conf.int = TRUE, conf.level = 0.95) %>%
  select(term, estimate, std.error, conf.low, conf.high) %>%
  knitr::kable(digits = 3)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	0.583	0.040	0.505	0.662
fruit_day	0.145	0.025	0.097	0.194

Our model `m2`

$\text{logit}(\text{BMI} < 30) = \log(\text{odds}(\text{BMI} < 30)) = 0.583 + 0.145$ fruit_day

- If Jaime consumes 5 servings of fruit per day, what is the prediction?

$\log(\text{odds}(\text{BMI} < 30)) = 0.583 + 0.145\ (5) = 0.583 + 0.725 = 1.308$

- Exponentiate to get the odds, on our way to estimating the probability.

$\text{odds}(\text{BMI} < 30) = \exp(1.308) = 3.699$

- so, we can estimate his Probability of BMI < 30 as. . .

$$Pr(BMI < 30) = \frac{3.699}{(3.699 + 1)} = 0.787.$$

Another Prediction

What is the predicted probability of BMI < 30 if a subject (Cersei) eats no fruit?

$$log(odds(BMI < 30)) = 0.583 + 0.145(0) = 0.583$$

$$odds(BMI < 30) = exp(0.583) = 1.791$$

$$Pr(BMI < 30) = \frac{1.791}{(1.791 + 1)} = 0.642$$

Can we get R to do this work for us?

Predictions from a Logistic Regression Model

```
new2 <- tibble( fruit_day = c(0, 5) )

predict(m2, newdata = new2, type = "link") # predicted logit
```

```
        1         2
0.5834646 1.3082673
```

```
exp(predict(m2, newdata = new2, type = "link")) # odds
```

```
       1        2
1.792237 3.699758
```

```
predict(m2, newdata = new2, type = "response") # probability
```

```
        1         2
0.6418642 0.7872231
```

Will augment do this, as well?

```
new2 <- tibble( fruit_day = c(0, 5) )

augment(m2, newdata = new2, type.predict = "link")
```

```
# A tibble: 2 x 3
  fruit_day .fitted .se.fit
      <dbl>   <dbl>   <dbl>
1         0   0.583  0.0403
2         5   1.31   0.0964
```

```
augment(m2, newdata = new2, type.predict = "response")
```

```
# A tibble: 2 x 3
  fruit_day .fitted .se.fit
      <dbl>   <dbl>   <dbl>
1         0   0.642 0.00925
2         5   0.787 0.0161
```

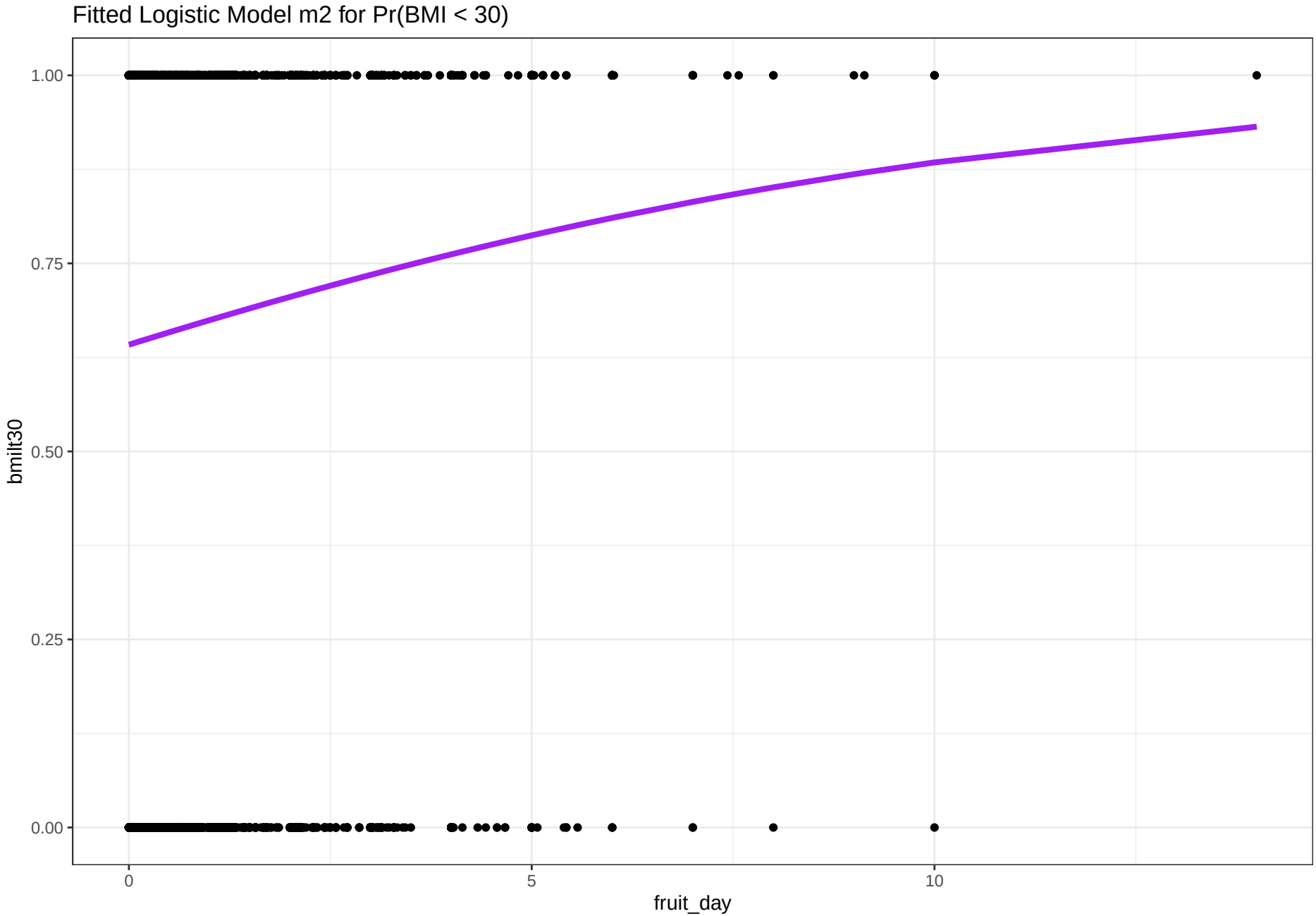
Plotting the Logistic Regression Model

Use the augment function to get the fitted probabilities into the original data, then plot.

```
m2_aug <- augment(m2, type.predict = "response")

ggplot(m2_aug, aes(x = fruit_day, y = bmilt30)) +
  geom_point() +
  geom_line(aes(x = fruit_day, y = .fitted),
            col = "purple", size = 1.5) +
  labs(title = "Fitted Logistic Model m2 for Pr(BMI < 30)")
```

- Results on next slide

Plotting Model m2



Evaluating the Model, again

```
m2
```

```
Call:  glm(formula = bmilt30 ~ fruit_day, family = binomial)

Coefficients:
(Intercept)    fruit_day
     0.5835       0.1450

Degrees of Freedom: 7411 Total (i.e. Null);  7410 Residual
Null Deviance:      9249
Residual Deviance: 9213     AIC: 9217
```

$$logit(BMI < 30) = log(odds(BMI < 30)) = 0.583 + 0.145 fruit$$

How can we interpret the coefficients of the model?

Could we try exponentiating the coefficients?

```
coef(m2)
```

```
(Intercept)   fruit_day
  0.5834646   0.1449605
```

```
exp(coef(m2))
```

```
(Intercept)   fruit_day
   1.792237    1.155994
```

Suppose Charlie ate one more piece of fruit per day than Harry.

- The **odds** of Charlie having BMI < 30 are 1.156 times as large as they are for Harry.
- Odds Ratio comparing two subjects whose `fruit_day` differ by 1 serving is 1.156.

More details on m2 coefficients

```
tidy(m2, exponentiate = TRUE, conf.int = TRUE) %>%
  select(term, estimate, std.error, conf.low, conf.high) %>%
  knitr::kable(digits = 3)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	1.792	0.040	1.656	1.939
fruit_day	1.156	0.025	1.101	1.214

- What would it mean if the Odds Ratio for `fruit_day` was 1?
- If Charlie eats more servings of fruit than Harry, what would an odds ratio for `fruit_day` that was greater than 1 mean?
- How about an odds ratio that was less than 1?
- What is the smallest possible Odds Ratio?

m2: some additional output

```
m2
```

```
Call:  glm(formula = bmilt30 ~ fruit_day, family = binomial)

Coefficients:
(Intercept)    fruit_day
     0.5835       0.1450

Degrees of Freedom: 7411 Total (i.e. Null);  7410 Residual
Null Deviance:      9249
Residual Deviance: 9213     AIC: 9217
```

- Think of the Deviance as a measure of "lack of fit".
- Deviance accounted for by m2 is
  - 9249 - 9213 = 36 points on 7411 - 7410 = 1 df
- Can do a likelihood ratio test via anova.

anova(m2) for our logistic regression model

Analysis of Deviance

```
anova(m2, test = "LRT")
```

```
Analysis of Deviance Table

Model: binomial, link: logit

Response: bmilt30

Terms added sequentially (first to last)

           Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                        7411     9248.7
fruit_day  1   35.737       7410     9213.0 2.259e-09 ***
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

m2: output from glance

```
glance(m2) %>% select(1:2, 6:7, 3)
```

```
# A tibble: 1 x 5
  null.deviance df.null deviance df.residual logLik
          <dbl>   <int>    <dbl>        <int>  <dbl>
1         9249.    7411    9213.         7410 -4606.
```

logLik = log-likelihood = - deviance/2

```
glance(m2) %>% select(4:5)
```

```
# A tibble: 1 x 2
    AIC   BIC
  <dbl> <dbl>
1 9217. 9231.
```

- AIC and BIC still useful for comparing models using the same outcome.

Can we predict BMI < 30 using dm_status?

```
smart1_sh <- smart1_sh %>%
  mutate(bmilt30 = as.numeric(bmi < 30),
         dm_status = fct_relevel(dm_status, "No"))

smart1_sh %>% tabyl(bmilt30) %>% adorn_pct_formatting()
```

```
 bmilt30    n percent
       0 2343   31.6%
       1 5069   68.4%
```

Two-Factor Linear Probability model for `bmilt30`

```
m3 <- smart1_sh %$%
  lm(bmilt30 ~ dm_status * genhealth)

anova(m3) %>% knitr::kable(digits = 3)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
dm_status	1	56.215	56.215	276.320	0.000
genhealth	4	38.003	9.501	46.700	0.000
dm_status:genhealth	4	2.273	0.568	2.793	0.025
Residuals	7402	1505.867	0.203	NA	NA

Equation for model m3

```
tidy(m3) %>%
  select(term, estimate) %>% knitr::kable(digits = 3)
```

term	estimate
(Intercept)	0.847
dm_statusYes	-0.120
genhealth2_VeryGood	-0.090
genhealth3_Good	-0.193
genhealth4_Fair	-0.213
genhealth5_Poor	-0.189
dm_statusYes:genhealth2_VeryGood	-0.101
dm_statusYes:genhealth3_Good	-0.041
dm_statusYes:genhealth4_Fair	-0.047
dm_statusYes:genhealth5_Poor	-0.198

- Prediction for a subject without diabetes who is in Excellent Health?

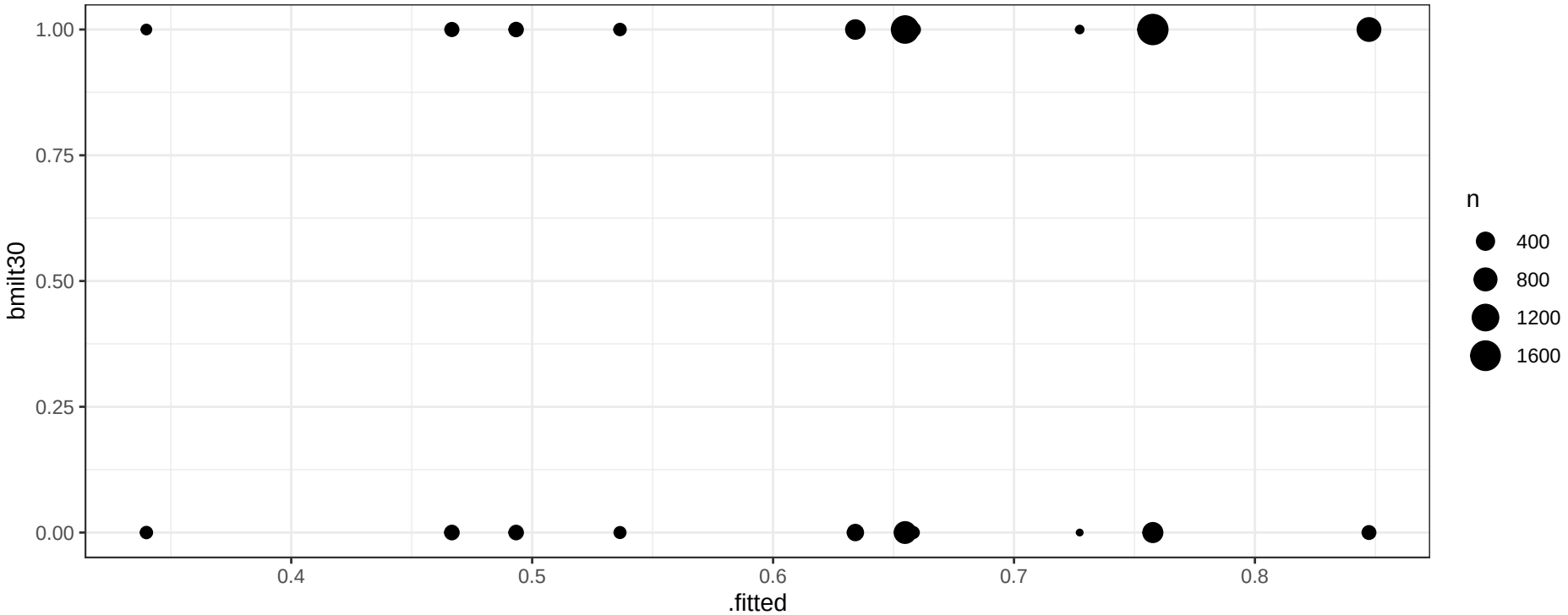
Get predictions for all subjects in our data

```
m3_aug <- augment(m3)

m3_aug %>% count(.fitted, dm_status, genhealth)
```

```
# A tibble: 10 x 4
   .fitted dm_status genhealth       n
     <dbl> <fct>     <fct>        <int>
 1   0.340 Yes       5_Poor         153
 2   0.467 Yes       4_Fair         360
 3   0.493 Yes       3_Good         375
 4   0.536 Yes       2_VeryGood     192
 5   0.634 No        4_Fair         779
 6   0.655 No        3_Good        1993
 7   0.658 No        5_Poor         275
 8   0.727 Yes       1_Excellent     22
 9   0.758 No        2_VeryGood    2228
10   0.847 No        1_Excellent   1035
```

Plot observed vs. predicted values

```
ggplot(m3_aug, aes(x = .fitted, y = bmilt30)) +
  geom_count()
```



How do we fit a simple logistic regression model?

```
m4 <- smart1_sh %$%
  glm(bmilt30 ~ dm_status, family = binomial(link = logit))
```

How do we interpret the coefficients?

```
tidy(m4) %>% select(term, estimate) %>%
  knitr::kable(digits = 3)
```

term	estimate
(Intercept)	0.946
dm_statusYes	-1.044

Equation: `logit(BMI < 30) = 0.946 - 1.044 (dm_status = Yes)`

How can we interpret this result?

Interpreting our Logistic Regression Equation

`logit(BMI < 30) = 0.946 - 1.044 (dm_status = Yes)`

- Harry has diabetes.
  - His predicted `logit(BMI < 30)` is 0.946 - 1.044 (1) = -0.098
- Sally does not have diabetes.
  - Her predicted `logit(BMI < 30)` is 0.946 - 1.044 (0) = 0.946

Now, `logit(BMI < 30)` = `log(odds(BMI < 30))`, so exponentiate to get the odds. . .

- Harry has predicted `odds(BMI < 30)` = exp(-0.098) = 0.9066
- Sally has predicted `odds(BMI < 30)` = exp(0.946) = 2.575

Can we convert these `odds` into something more intuitive?

Converting Odds to Probabilities

- Harry has predicted `odds(BMI < 30)` = exp(-0.098) = 0.9066
- Sally has predicted `odds(BMI < 30)` = exp(0.946) = 2.575

$$odds(BMI < 30) = \frac{Pr(BMI < 30)}{1 - Pr(BMI < 30)}$$

and

$$Pr(BMI < 30) = \frac{odds(BMI < 30)}{odds(BMI < 30) + 1}$$

- So Harry's predicted `Pr(BMI < 30)` = 0.9066 / 1.9066 = 0.48
- Sally's predicted `Pr(BMI < 30)` = 2.575 / 3.575 = 0.72
- odds range from 0 to ∞, and log(odds) range from $-\infty$ to ∞.
- odds > 1 if probability > 0.5. If odds $= 1$, then probability $= 0.5$.

What about the odds ratio?

`logit(BMI < 30) = 0.946 - 1.044 (dm_status = Yes)`

- Harry, with diabetes, has odds(BMI < 30) $= 0.9066$
- Sally, without diabetes, has odds(BMI < 30) $= 2.575$

Odds Ratio for BMI < 30 associated with having diabetes (vs. not) $=$

$$\frac{0.9066}{2.575} = 0.352$$

- Our model estimates that a subject with diabetes has 35.2% of the odds of a subject without diabetes of having BMI < 30.

Can we calculate the odds ratio from the equation's coefficients?

- Yes, `exp(-1.044)` $= 0.352$.

Tidy with exponentiation

```
tidy(m4, exponentiate = TRUE,
     conf.int = TRUE, conf.level = 0.9) %>%
  select(term, estimate, conf.low, conf.high) %>%
  knitr::kable(digits = 3)
```

term	estimate	conf.low	conf.high
(Intercept)	2.575	2.459	2.697
dm_statusYes	0.352	0.316	0.393

- The odds ratio for BMI $<$ 30 among subjects with diabetes as compared to those without diabetes is 0.352
- The odds of BMI $<$ 30 are 35.2% as large for subjects with diabetes as they are for subjects without diabetes, according to this model.
- A 90% uncertainty interval for the odds ratio estimate includes (0.316, 0.393).

Interpreting these summaries

Connecting the Odds Ratio and Log Odds Ratio to probability statements. . .

- If the probabilities were the same (for diabetes and non-diabetes subjects) of having BMI < 30, then the odds would also be the same, and so the odds ratio would be 1.
- If the probabilities of BMI < 30 were the same and thus the odds were the same, then the log odds ratio would be `log(1)` $= 0$.

`logit(BMI < 30) = 0.946 - 1.044 (dm_status = Yes)`

1. If the log odds of a coefficient (like `diabetes = Yes`) are negative, then what does that imply?
2. What if we flipped the order of the levels for diabetes so our model was about `diabetes = No`?

New model: `logit(BMI < 30) = -0.098 + 1.044 (dm_status = No)`

Two-Factor Logistic Regression (model m5)

First, let's try a model without interaction.

```
m5_without <- smart1_sh %$%
      glm(bmilt30 ~ dm_status + genhealth,
          family = binomial()) # logit is default link

tidy(m5_without) %>% select(term, estimate) %>%
  knitr::kable(digits = 3)
```

term	estimate
(Intercept)	1.716
dm_statusYes	-0.813
genhealth2_VeryGood	-0.595
genhealth3_Good	-1.051
genhealth4_Fair	-1.124
genhealth5_Poor	-1.244

Our model `m5_without`

```
logit(BMI < 30) = log(odds(BMI < 30))
   = 1.72 - 0.81 (dm_status = Yes)
          - 0.60 (genhealth = Very Good)
          - 1.05 (genhealth = Good)
          - 1.12 (genhealth = Fair)
          - 1.24 (genhealth = Poor)
```

1. How do we interpret the meaning of the -0.81 coefficient for `dm_status = Yes` in this model?
2. How do we interpret the meaning of the -1.05 coefficient for `genhealth = Good`?

Our model `m5_without`

```
logit(BMI < 30) =
   = 1.72 - 0.81 (dm = Yes) - 0.60 (Very Good) - 1.05 (Good)
      - 1.12 (Fair) - 1.24 (Poor)
```

1. How do we interpret the meaning of the -0.81 coefficient for `dm_status = Yes` in this model?

If Harry and Sally have the **same `genhealth` status**, but Harry has diabetes and Sally does not, the model predicts that Harry's `log(odds(BMI < 30))` will be 0.81 lower than Sally's.

- Harry: `logit(BMI < 30) = (1.72 - 0.81) - 0.60 (Very Good) - 1.05 (Good) - 1.12 (Fair) - 1.24 (Poor)`
- Sally: `logit(BMI < 30) = 1.72 - 0.60 (VG) - 1.05 (G) - 1.12 (F) - 1.24 (P)`

Suppose that, for example, Harry and Sally each had Excellent `genhealth`...

Question 1 (continued)

```
logit(BMI < 30) =
   = 1.72 - 0.81 (dm = Yes) - 0.60 (Very Good) - 1.05 (Good)
      - 1.12 (Fair) - 1.24 (Poor)
```

1. How do we interpret the meaning of the -0.81 coefficient for `dm_status = Yes` in this model?

Subject	Harry	Sally
genhealth	Excellent	Excellent
dm_status	Yes	No
log(odds(BMI < 30))	1.72 - 0.81 = 0.91	1.72
odds(BMI < 30)	exp(0.91) = 2.484	exp(1.72) = 5.585
Pr(BMI < 30)	2.484/3.484 = 0.71	5.585/6.585 = 0.85

Our model `m5_without`

```
logit(BMI < 30) =
   = 1.72 - 0.81 (dm = Yes) - 0.60 (Very Good) - 1.05 (Good)
       - 1.12 (Fair) - 1.24 (Poor)
```

2. How do we interpret the meaning of the -1.05 coefficient for `genhealth = Good`?

If Harry and Sally have the **same `dm_status`**, but Harry has Good `genhealth` and Sally has Excellent `genhealth`, the model predicts that Harry's `log(odds(BMI < 30))` will be 1.05 lower than Sally's.

- Harry: `logit(BMI < 30) = 1.72 - 0.81 (dm = Yes) - 1.05`
- Sally: `logit(BMI < 30) = 1.72 - 0.81 (dm = Yes)`

Why are we comparing Harry at Good to Sally at Excellent here?

Question 2 (continued)

```
logit(BMI < 30) =
   = 1.72 - 0.81 (dm = Yes) - 0.60 (Very Good) - 1.05 (Good)
      - 1.12 (Fair) - 1.24 (Poor)
```

2. How do we interpret the meaning of the -1.05 coefficient for `genhealth = Good`?

Subject	Harry	Sally
`genhealth`	Good	Excellent
`dm_status`	No	No
`log(odds(BMI < 30))`	1.72 - 1.05 = 0.67	1.72
`odds(BMI < 30)`	exp(0.67) = 1.954	exp(1.72) = 5.585
`Pr(BMI < 30)`	1.954/2.954 = 0.66	5.585/6.585 = 0.85

- What is the odds ratio for BMI < 30 comparing Harry to Sally? $1.954/5.585 = 0.350$
- Now, what if Harry and Sally each had diabetes?

Question 2 (continued)

```
logit(BMI < 30) =
   = 1.72 - 0.81 (dm = Yes) - 0.60 (Very Good) - 1.05 (Good)
      - 1.12 (Fair) - 1.24 (Poor)
```

2. How do we interpret the meaning of the -1.05 coefficient for `genhealth = Good`?

Subject	Harry	Sally
`genhealth`	Good	Excellent
`dm_status`	Yes	Yes
`log(odds(BMI < 30))`	1.72 - 1.05 - 0.81 = -0.14	1.72 - 0.81 = 0.91
`odds(BMI < 30)`	exp(-0.14) = 0.869	exp(0.91) = 2.484
`Pr(BMI < 30)`	0.869/1.869 = 0.46	2.484/3.484 = 0.71

Now what is the odds ratio for BMI $<$ 30 comparing Harry to Sally?
0.869/2.484 = 0.350

Tidying our m5_without model

```
tidy(m5_without, exponentiate = TRUE,
     conf.int = TRUE, conf.level = 0.90) %>%
  select(term, estimate, conf.low, conf.high) %>%
  knitr::kable(digits = 3)
```

term	estimate	conf.low	conf.high
(Intercept)	5.565	4.848	6.416
dm_statusYes	0.444	0.396	0.498
genhealth2_VeryGood	0.551	0.469	0.646
genhealth3_Good	0.350	0.298	0.409
genhealth4_Fair	0.325	0.272	0.387
genhealth5_Poor	0.288	0.232	0.358

How do we interpret the odds ratios here?

m5_with: Adding an interaction term?

```
m5_with <- smart1_sh %$%
  glm(bmilt30 ~ dm_status * genhealth, family = binomial())

tidy(m5_with) %>%
  select(term, estimate, std.error, p.value) %>%
  knitr::kable(digits = 3)
```

Results on next slide. . .

Coefficients of model `m5_with`

```
m5_with <- smart1_sh %$%
  glm(bmilt30 ~ dm_status * genhealth, family = binomial())
```

term	estimate	std.error	p.value
(Intercept)	1.714	0.086	0.000
dm_statusYes	-0.733	0.486	0.132
genhealth2_VeryGood	-0.574	0.100	0.000
genhealth3_Good	-1.074	0.098	0.000
genhealth4_Fair	-1.164	0.114	0.000
genhealth5_Poor	-1.059	0.154	0.000
dm_statusYes:genhealth2_VeryGood	-0.261	0.510	0.609
dm_statusYes:genhealth3_Good	0.066	0.500	0.894
dm_statusYes:genhealth4_Fair	0.050	0.503	0.922
dm_statusYes:genhealth5_Poor	-0.586	0.531	0.270

Interpreting `m5_with` Coefficients

Equation for log(odds(BMI $<$ 30)) $=$

```
1.71 - 0.73 (dm = Yes)
- 0.57 (Very Good) - 1.07 (Good) - 1.16 (Fair) - 1.06 (Poor)
- 0.26 (dm = Yes)(Very Good) + 0.07 (dm = Yes)(Good)
+ 0.05 (dm = Yes)(Fair) - 0.59 (dm = Yes)(Poor)
```

How do we understand the -0.59 coefficient here?

Suppose Cersei has Excellent and Jaime has Poor `genhealth`. What are their model equations for `log(odds(BMI < 30))`?

- Cersei: 1.71 - 0.73 `dm_status`
- Jaime: (1.71 - 1.06) + ((-0.73) + (-0.59)) `dm_status`,
- so Jaime: 0.65 - 1.32 `dm_status`.

Making Predictions with `m5_with`

Equation for log(odds(BMI $<$ 30)) =

```
1.71 - 0.73 (dm = Yes)
- 0.57 (Very Good) - 1.07 (Good) - 1.16 (Fair) - 1.06 (Poor)
- 0.26 (dm = Yes)(Very Good) + 0.07 (dm = Yes)(Good)
+ 0.05 (dm = Yes)(Fair) - 0.59 (dm = Yes)(Poor)
```

Subject	dm_status	genhealth	log(odds(BMI $<$ 30))
Harry	No	Excellent	1.71
Sally	No	Poor	1.71 - 1.06 = 0.65
Cersei	Yes	Excellent	1.71 - 0.73 = 0.98
Jaime	Yes	Poor	1.71 - 0.73 - 1.06 - 0.59 = -0.67

Getting R to make the predictions

(Reducing rounding errors)

```
new_m5 <- tibble(
  subject = c("Harry", "Sally", "Cersei", "Jaime"),
  dm_status = c("No", "No", "Yes", "Yes"),
  genhealth = c("1_Excellent", "5_Poor",
                "1_Excellent", "5_Poor"))

predict(m5_with, newdata = new_m5, type = "link")
```

```
         1          2          3          4
 1.7139120  0.6552022  0.9808293 -0.6638768
```

Making Predictions with `m5_with` (again)

```
1.71 - 0.73 (dm = Yes)
- 0.57 (Very Good) - 1.07 (Good) - 1.16 (Fair) - 1.06 (Poor)
- 0.26 (dm = Yes)(Very Good) + 0.07 (dm = Yes)(Good)
+ 0.05 (dm = Yes)(Fair) - 0.59 (dm = Yes)(Poor)
```

Subject	dm	genhealth	odds(BMI $<$ 30)
Harry	No	Excellent	exp(1.71) = 5.53
Sally	No	Poor	exp(0.65) = 1.92
Cersei	Yes	Excellent	exp(0.98) = 2.66
Jaime	Yes	Poor	exp(-0.67) = 0.51

Getting R to make the predictions

(Reducing rounding errors)

```
predict(m5_with, newdata = new_m5, type = "link") # logit
```

```
         1          2          3          4
 1.7139120  0.6552022  0.9808293 -0.6638768
```

```
exp(predict(m5_with, newdata = new_m5, type = "link")) # odds
```

```
        1         2         3         4
5.5506329 1.9255319 2.6666667 0.5148515
```

Making Predictions with `m5_with` (one more time)

```
1.71 - 0.73 (dm = Yes)
- 0.57 (Very Good) - 1.07 (Good) - 1.16 (Fair) - 1.06 (Poor)
- 0.26 (dm = Yes)(Very Good) + 0.07 (dm = Yes)(Good)
+ 0.05 (dm = Yes)(Fair) - 0.59 (dm = Yes)(Poor)
```

How do we understand the -0.59 coefficient here?

Subject	dm	genhealth	Pr(BMI $<$ 30)
Harry	No	Excellent	$5.53/6.53 = 0.85$
Sally	No	Poor	$1.92/2.92 = 0.66$
Cersei	Yes	Excellent	$2.66/3.66 = 0.73$
Jaime	Yes	Poor	$0.51/1.51 = 0.34$

Getting R to make the predictions

```
predict(m5_with, newdata = new_m5,
        type = "response") # probs
```

```
        1         2         3         4
0.8473430 0.6581818 0.7272727 0.3398693
```

Model `m5_with` Results (from R's `predict`)

Subject	dm	genhealth	logit	odds	Pr(BMI $<$ 30)
Harry	No	Excellent	1.714	5.551	0.847
Sally	No	Poor	0.655	1.926	0.658
Cersei	Yes	Excellent	0.981	2.667	0.727
Jaime	Yes	Poor	-0.664	0.515	0.340

Calculating Odds Ratios

- Comparing DM to No DM (if GenHealth = Excellent) = 2.667/5.551 = 0.480
- Comparing Poor to Excellent (if no DM) = 1.926 / 5.551 = 0.347
- Comparing DM to No DM (if GenHealth = Poor) = 0.515/1.926 = 0.267
- Comparing Poor to Excellent (if DM) = 0.515 / 2.667 = 0.193

Exponentiating the `m5_with` Coefficients

```
tidy(m5_with, exponentiate = TRUE, conf.int = TRUE,
     conf.level = 0.90) %>%
  select(term, estimate, conf.low, conf.high) %>%
  knitr::kable(digits = 3)
```

Results on the next slide. . .

Exponentiating the `m5_with` Coefficients

term	estimate	conf.low	conf.high
(Intercept)	5.551	4.826	6.414
dm_statusYes	0.480	0.224	1.132
genhealth2_VeryGood	0.563	0.477	0.662
genhealth3_Good	0.342	0.290	0.401
genhealth4_Fair	0.312	0.259	0.376
genhealth5_Poor	0.347	0.270	0.447
dm_statusYes:genhealth2_VeryGood	0.771	0.316	1.726
dm_statusYes:genhealth3_Good	1.068	0.445	2.349
dm_statusYes:genhealth4_Fair	1.051	0.435	2.326
dm_statusYes:genhealth5_Poor	0.557	0.221	1.292

1. Interpret the `dm_statusYes` coefficient (0.480).
2. Interpret the `genhealth5_Poor` coefficient (0.347).

Model `m5_with` Predictions, Again

1. Interpret the `dm_statusYes` coefficient (0.480).
2. Interpret the `genhealth5_Poor` coefficient (0.347).

Subject	dm	genhealth	odds(BMI $<$ 30)
Harry	No	Excellent	5.551
Sally	No	Poor	1.926
Cersei	Yes	Excellent	2.667
Jaime	Yes	Poor	0.515

Odds Ratios we calculated earlier...

1. Comparing DM to No DM (if GenHealth = Excellent) = 2.667/5.551 = 0.480
2. Comparing Poor to Excellent (if no DM) = 1.926 / 5.551 = 0.347

Exponentiating the `m5_with` Coefficients

term	estimate	conf.low	conf.high
(Intercept)	5.551	4.826	6.414
dm_statusYes	0.480	0.224	1.132
genhealth2_VeryGood	0.563	0.477	0.662
genhealth3_Good	0.342	0.290	0.401
genhealth4_Fair	0.312	0.259	0.376
genhealth5_Poor	0.347	0.270	0.447
dm_statusYes:genhealth2_VeryGood	0.771	0.316	1.726
dm_statusYes:genhealth3_Good	1.068	0.445	2.349
dm_statusYes:genhealth4_Fair	1.051	0.435	2.326
dm_statusYes:genhealth5_Poor	0.557	0.221	1.292

3. How do we interpret the interaction coefficients, like 0.557 for (DM = Yes)(GenHealth = Poor)?

Interpreting `m5_with` Interaction Odds Ratios

3. How do we interpret the interaction coefficients, like 0.557 for (DM = Yes)(GenHealth = Poor)?

Odds Ratios we calculated earlier. . .

- Comparing DM to No DM (if GenHealth = Poor) $\approx$ 0.267
- Comparing DM to No DM (if GenHealth = Excellent) $\approx$ 0.480
- Comparing Poor to Excellent (if DM) $\approx$ 0.193
- Comparing Poor to Excellent (if no DM) $\approx$ 0.347

Within rounding error,

$$\frac{0.267}{0.480} \approx \frac{0.193}{0.347} \approx 0.557$$

Using glance on these models

```
bind_rows(glance(m5_with), glance(m5_without)) %>%
  mutate(model = c("With Interaction", "No Interaction"),
         deviance_diff = null.deviance - deviance,
         df_diff = df.null - df.residual) %>%
  select(model, AIC, BIC, deviance_diff, df_diff) %>%
  knitr::kable(digits = 1)
```

model	AIC	BIC	deviance_diff	df_diff
With Interaction	8821.6	8890.7	447.1	9
No Interaction	8823.5	8864.9	437.2	5

Logistic Regression Comparisons via `anova`

Based on Likelihood Ratio Test

```
anova(m5_without, m5_with, test = "LRT")
```

```
Analysis of Deviance Table

Model 1: bmilt30 ~ dm_status + genhealth
Model 2: bmilt30 ~ dm_status * genhealth
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1      7406     8811.5
2      7402     8801.6  4   9.8769  0.04255 *
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Other options include Rao's efficient score test (`test = "Rao"`) and Pearson's chi-square test (`test = "Chisq"`)

Logistic Regression Comparisons via `anova`

Another potentially attractive option compares the models based on Mallows' C_p statistic, which is closely related to the AIC, in general, and identical to what `glance` provides for AIC in this case.

```
anova(m5_without, m5_with, test = "Cp")
```

```
Analysis of Deviance Table

Model 1: bmilt30 ~ dm_status + genhealth
Model 2: bmilt30 ~ dm_status * genhealth
  Resid. Df Resid. Dev Df Deviance     Cp
1      7406     8811.5                8823.5
2      7402     8801.6  4   9.8769 8821.6
```

m6: Logistic Regression (Interaction & Covariate)

```
m6 <- smart1_sh %$%
  glm(bmilt30 ~ fruit_day + dm_status * genhealth,
      family = binomial)

tidy(m6) %>%
  select(term, estimate, std.error, p.value) %>%
  knitr::kable(digits = 3)
```

Results on next slide. . .

m6 model coefficients

term	estimate	std.error	p.value
(Intercept)	1.548	0.094	0.000
fruit_day	0.114	0.025	0.000
dm_statusYes	-0.741	0.487	0.128
genhealth2_VeryGood	-0.563	0.100	0.000
genhealth3_Good	-1.052	0.099	0.000
genhealth4_Fair	-1.129	0.114	0.000
genhealth5_Poor	-1.016	0.154	0.000
dm_statusYes:genhealth2_VeryGood	-0.258	0.511	0.614
dm_statusYes:genhealth3_Good	0.071	0.500	0.887
dm_statusYes:genhealth4_Fair	0.051	0.504	0.920
dm_statusYes:genhealth5_Poor	-0.601	0.532	0.259

The m6 model

```
log(odds(BMI < 30)) =
  1.548 +
  + 0.114 fruit_day
  - 0.741 dm_status = Yes
  - 0.563 genhealth = Very Good
  - 1.052 genhealth = Good
  - 1.129 genhealth = Fair
  - 1.016 genhealth = Poor
  - 0.258 (dm_status = Yes)(genhealth = Very Good)
  + 0.071 (dm_status = Yes)(genhealth = Good)
  + 0.051 (dm_status = Yes)(genhealth = Fair)
  - 0.601 (dm_status = Yes)(genhealth = Poor)
```

Does the impact of fruit_day change depending on dm_status and genhealth?

The m7 model with factor-covariate interactions

```
m7 <- smart1_sh %$%
  glm(bmilt30 ~
        fruit_day*dm_status +
        fruit_day*genhealth +
        dm_status*genhealth,
      family = binomial)

tidy(m7) %>%
  select(term, estimate, std.error, p.value) %>%
  knitr::kable(digits = 3)
```

The m7 model

term	estimate	std.error	p.value
(Intercept)	1.463	0.149	0.000
fruit_day	0.175	0.088	0.048
dm_statusYes	-0.788	0.498	0.113
genhealth2_VeryGood	-0.481	0.168	0.004
genhealth3_Good	-0.957	0.166	0.000
genhealth4_Fair	-1.039	0.180	0.000
genhealth5_Poor	-0.798	0.224	0.000
fruit_day:dm_statusYes	0.029	0.065	0.656
fruit_day:genhealth2_VeryGood	-0.059	0.099	0.555
fruit_day:genhealth3_Good	-0.069	0.099	0.486
fruit_day:genhealth4_Fair	-0.066	0.107	0.542
fruit_day:genhealth5_Poor	-0.184	0.132	0.163
dm_statusYes:genhealth2_VeryGood	-0.251	0.512	0.624
dm_statusYes:genhealth3_Good	0.081	0.502	0.872
dm_statusYes:genhealth4_Fair	0.063	0.506	0.901
dm_statusYes:genhealth5_Poor	-0.567	0.534	0.288

Could we do a three-way interaction?

```
m8 <- smart1_sh %$%
  glm(bmilt30 ~ fruit_day*dm_status*genhealth,
      family = binomial)
```

term	estimate
(Intercept)	1.504
fruit_day	0.145
dm_statusYes	-2.526
genhealth2_VeryGood	-0.527
dm_statusYes:genhealth5_Poor	0.734
fruit_day:dm_statusYes:genhealth2_VeryGood	-1.599
fruit_day:dm_statusYes:genhealth3_Good	-1.676
fruit_day:dm_statusYes:genhealth4_Fair	-1.609
fruit_day:dm_statusYes:genhealth5_Poor	-1.225

These are just 9 of the 20 coefficients fit in total.

Comparison of Models with Deviance Tests

```
anova(m5_without, m5_with, m6, m7, m8, test = "LRT")
```

```
Analysis of Deviance Table

Model 1: bmilt30 ~ dm_status + genhealth
Model 2: bmilt30 ~ dm_status * genhealth
Model 3: bmilt30 ~ fruit_day + dm_status * genhealth
Model 4: bmilt30 ~ fruit_day * dm_status + fruit_day * genheal
    genhealth
Model 5: bmilt30 ~ fruit_day * dm_status * genhealth
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1      7406     8811.5
2      7402     8801.6  4   9.8769   0.04255 *
3      7401     8780.3  1  21.3157 3.895e-06 ***
4      7396     8778.2  5   2.0927   0.83617
5      7392     8770.2  4   7.9755   0.09248 .
---
Signif. codes:
```

Comparison of Models with AIC/BIC

```
bind_rows(glance(m5_without), glance(m5_with), glance(m6),
          glance(m7), glance(m8)) %>%
  mutate(model = c("m5_without", "m5_with",
                   "m6", "m7", "m8")) %>%
  select(model, AIC, BIC)
```

```
# A tibble: 5 x 3
  model         AIC   BIC
  <chr>       <dbl> <dbl>
1 m5_without 8823. 8865.
2 m5_with    8822. 8891.
3 m6         8802. 8878.
4 m7         8810. 8921.
5 m8         8810. 8948.
```

What's Still To Come?

Building on what we know about Linear & Logistic Regression

- Model Selection and Cross-Validation Strategies
- Incorporating the Survey Weights into our Analyses
- Checking Assumptions (in Logistic Regression)
- Multiple Imputation (rather than Simple Imputation) to deal with missing data

At which point, we'll move on to . . .

- Other methods for predicting 1/0 and quantitative outcomes
- Using regression-style approaches to predict other kinds of outcomes (counts, multiple categories, times to event with censoring)