Classification

PSTAT100 Winter 2025

Announcements

- HW 4 due tonight.
  - Check eval: true
- Last lab due end of quarter.
- Projects

From last time

We fit this model to the tree cover data:

$$\log(\text{cover}_i) = \beta_0 + \beta_1 \log(\text{income}_i) + \beta_2 \text{low}_i + \beta_3 \text{med}_i + \beta_4 \text{high}_i + \epsilon_i$$

Each level of population density has its own intercept:

population density

$$\begin{aligned}
\text{very low density:} \quad & \mathbb{E}\log(\text{cover}) = \beta_0 + \beta_1 \log(\text{income}) \\
\text{low density:} \quad & \mathbb{E}\log(\text{cover}) = (\beta_0 + \beta_2) + \beta_1 \log(\text{income}) \\
\text{medium density:} \quad & \mathbb{E}\log(\text{cover}) = (\beta_0 + \beta_3) + \beta_1 \log(\text{income}) \\
\text{high density:} \quad & \mathbb{E}\log(\text{cover}) = (\beta_0 + \beta_4) + \beta_1 \log(\text{income})
\end{aligned}$$

Reference.

$\beta_2, \beta_3, \beta_4$ represent the *difference in expected log cover* between very low density and low, medium, high density after accounting for income

lm(log(tree) ~ log(income) + pop_dens)

$$y_i \sim N(X\beta, \sigma^2)$$

$$\Rightarrow E[y|X] = X\beta$$

$$\log(y) \sim N(X\beta, \sigma^2),$$

$z \sim N(\mu, 1)$

$e^z \sim LN(\ )$

$med = e^{\mu}$.

y is log-normal distribution.

$$\text{median}(y) = e^{X\beta}$$

Interpreting estimates

estimate	standard error	type
-3.9945020	0.5857494	(Intercept)
0.5542274	0.0550729	log_income
-0.2859815	0.0670307	pop_densitylow
-0.6214309	0.0757230	pop_densitymedium
-0.6607406	0.0963590	pop_densityhigh

- each doubling of mean income is associated with an estimated 55% increase in median tree cover, after accounting for population density
- census blocks with higher population densities are estimated as having a median tree canopy up to 50% lower than census blocks with very low population densities, after accounting for mean income

On log-transforming the response

The model is $\log(y) \sim N(x\beta, \sigma^2)$; so y is what's known as a *lognormal* random variable.

From the properties of the lognormal distribution:

$$e^{x\beta} = \text{median}(y)$$

So when parameters are back-transformed, they should be interpreted in terms of the *median* response.

Interpretations, again

$$55\% \text{ increase in median tree cover}: e^{\hat{\beta_1} \log(2)} = 1.549$$

Median cover increases by a factor of 1.549, *i.e.*, increases by 54.9%:

- doubling income increments log income by $\log(2)$
- $\hat{\beta_1} \log(2)$ gives the associated change in mean log cover
- exponentiating the change in mean log cover gives the multiplicative change in median cover

Prediction

```r
x_new <- data.frame(log_income = log(115000), pop_density = factor("medium", levels=levels(regdata$pop_densi
pred <- predict(mlr, newdata = x_new, interval = "confidence")
exp(pred)
```

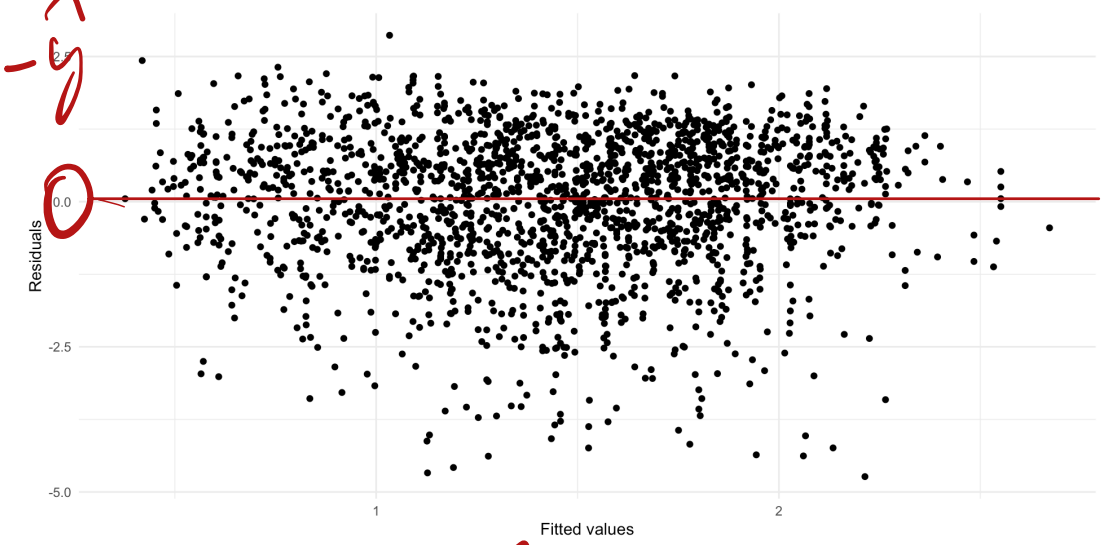
```
       fit      lwr      upr
1 6.311072 5.248709 7.588462
```

Fill in the blanks:

- the median tree cover for a _____ density census block with mean income _____ is estimated to be between _____ and _____ percent
- the tree cover for a _____ density census block with mean income _____ is estimated to be between _____ and _____ percent

Model checking

The linearity and constant variance assumptions can be assessed by plotting residuals against fitted values:



Should see minimal pattern:

- centered at zero
- even spread in either direction

Diabetes data

4831 responses from the 2011-2012 National Health and Nutrition Examination Survey (NHANES):

	Gender	Age	BMI	Diabetes
1	male	14	17.3	No
2	female	43	33.3	No
3	male	80	33.9	No
4	male	80	33.9	No

Binary: "yes" or "no".

Is BMI a risk factor for diabetes after adjusting for age and sex?

possible confounders.

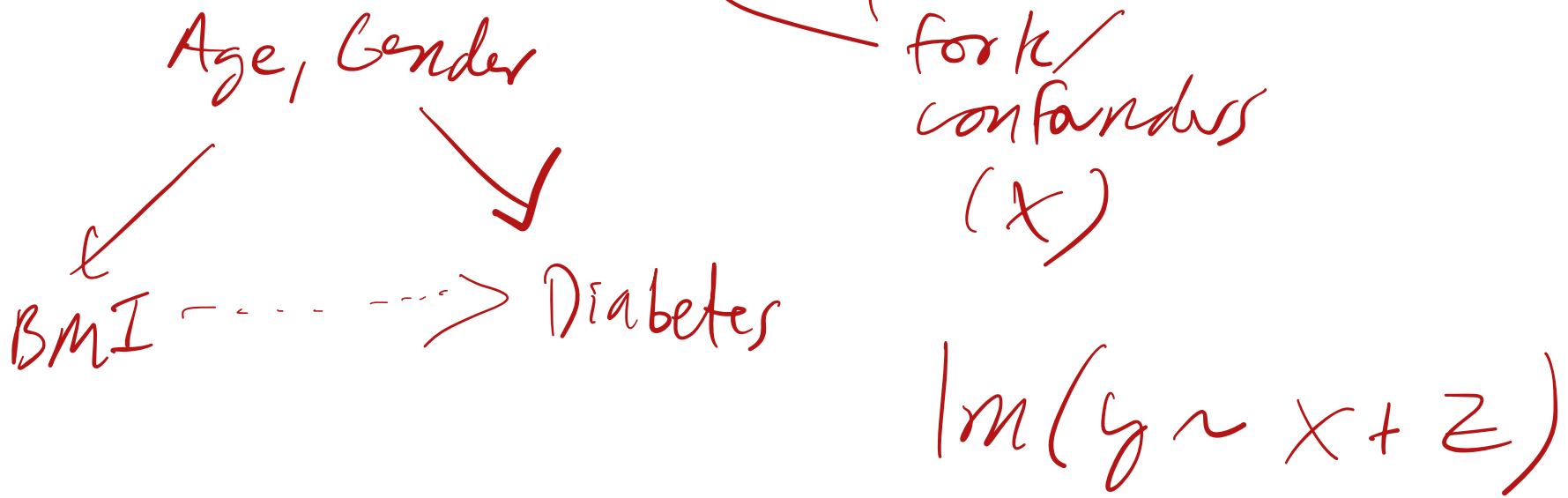
Model sketch

Broadly, we can answer the question by estimating the dependence of diabetes status on age, sex, and BMI.

An additive model might look something like this:

$$\text{diabetes}_i \leftarrow \beta_1 \text{age}_i + \beta_2 \text{male}_i + \beta_3 \text{BMI}_i$$

To answer the question, fit the model and examine β_3.



Binary response

Note that the response variable – whether the respondent has diabetes – is categorical.

```
[1] "No"  "Yes"
```

We can encode this using an indicator variable, which results in a binary response:

```
[1] 0 1
```

Remember, a statistical model is a probability distribution, so we need to choose one that's appropriate for binary outcomes. Ideas?

y: 1 if "yes"

0 if "no"

What not to do

One might think:

$$\text{diabetes}_i = \beta_0 + \beta_1 \text{age}_i + \beta_2 \text{male}_i + \beta_3 \text{BMI}_i + \epsilon_i$$

But $\text{diabetes}_i \sim N(x\beta, \sigma^2)$

- discrete, not continuous
- normal model doesn't make sense for a binary response

$(X^TX)^{-1}X^Ty = \hat{\beta}$

Can't get good confidence intervals / uncertainty.

What not to do

Note that you *can* still fit this model.

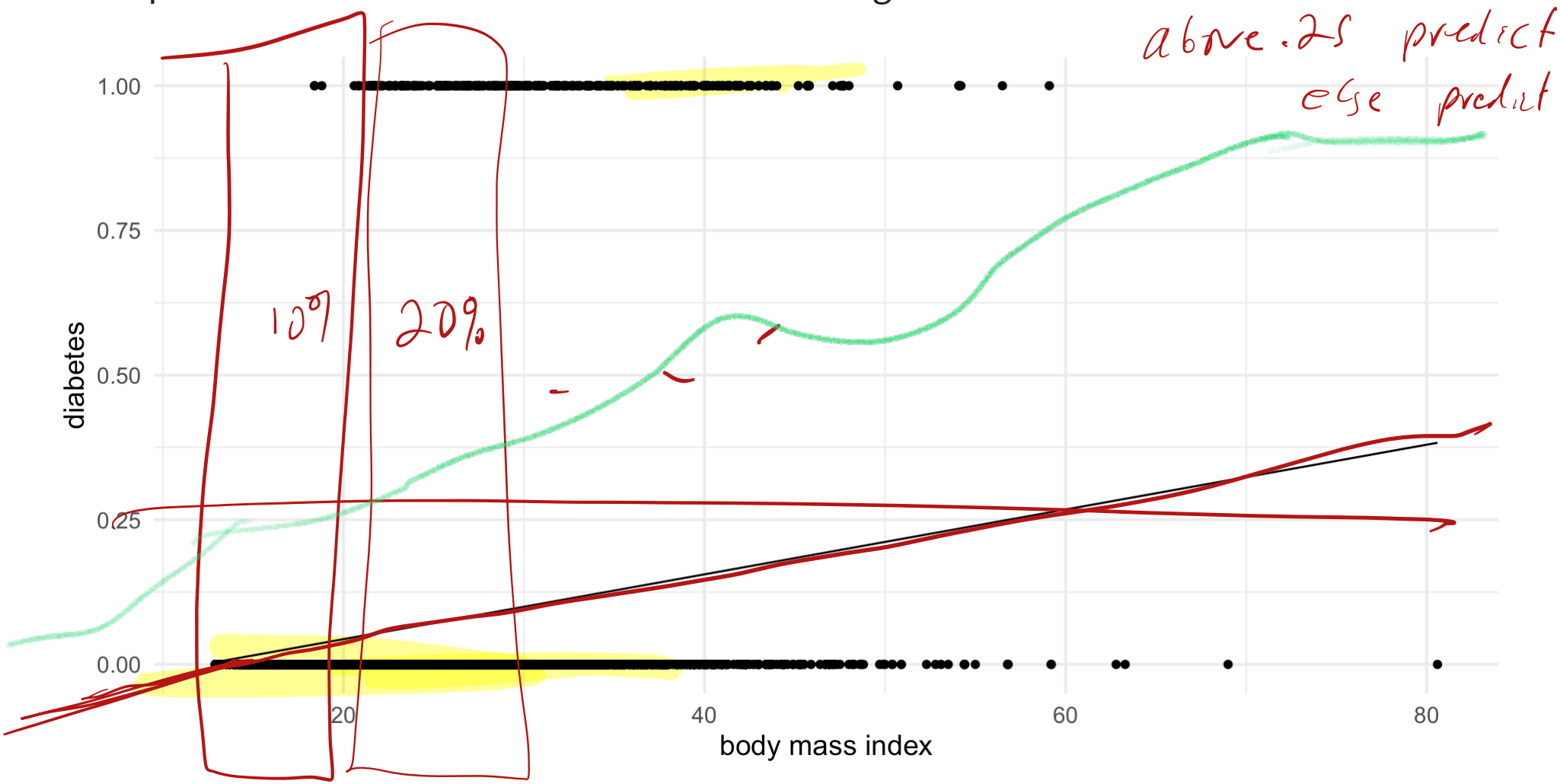
```
 (Intercept)         Male          Age          BMI
-0.169001564  0.010836864  0.002424943  0.005601656
```

So you have to discern that it isn't appropriate. A few ways to tell:

- parameter interpretations won't make sense – *e.g.* age is associated with a 0.0024 increase in diabetes presence
- model may yield predictions that are negative or greater than one
- plots will look odd

What not to do

Attempts at model visualization will look something like this:



Regression with a binary response

For a binary response $Y \in \{0, 1\}$, we model $P(Y = 1)$ as a function of the explanatory variable(s) x:

$$P(Y = 1) = f(x)$$

Of course, we don't directly observe $P(Y = 1)$ – but there are various ways around this.

$P(Y=1|x) = f(x)$,

$f(x)$ must be between 0 and 1

for all x.

Logistic regression model

The most common approach to modeling binary responses is *logistic regression*:

$$\log\left(\frac{p_x}{1-p_x}\right) = x'\beta$$

any number.

must be positive

"the log odds are linear in x"

- $p_x = P(Y = 1 | x) \in [0, 1]$
- x is a vector of explanatory variable(s)
- β is a vector of parameters

> This model holds that **the log odds of the outcome of interest is a linear function of the explanatory variable(s)**

Odds: $p = 2/3$. $\frac{p}{1-p} = \frac{2/3}{1/3} = 2$.

Logistic regression model

What does the model imply about the probability (rather than log-odds) of the outcome of interest?

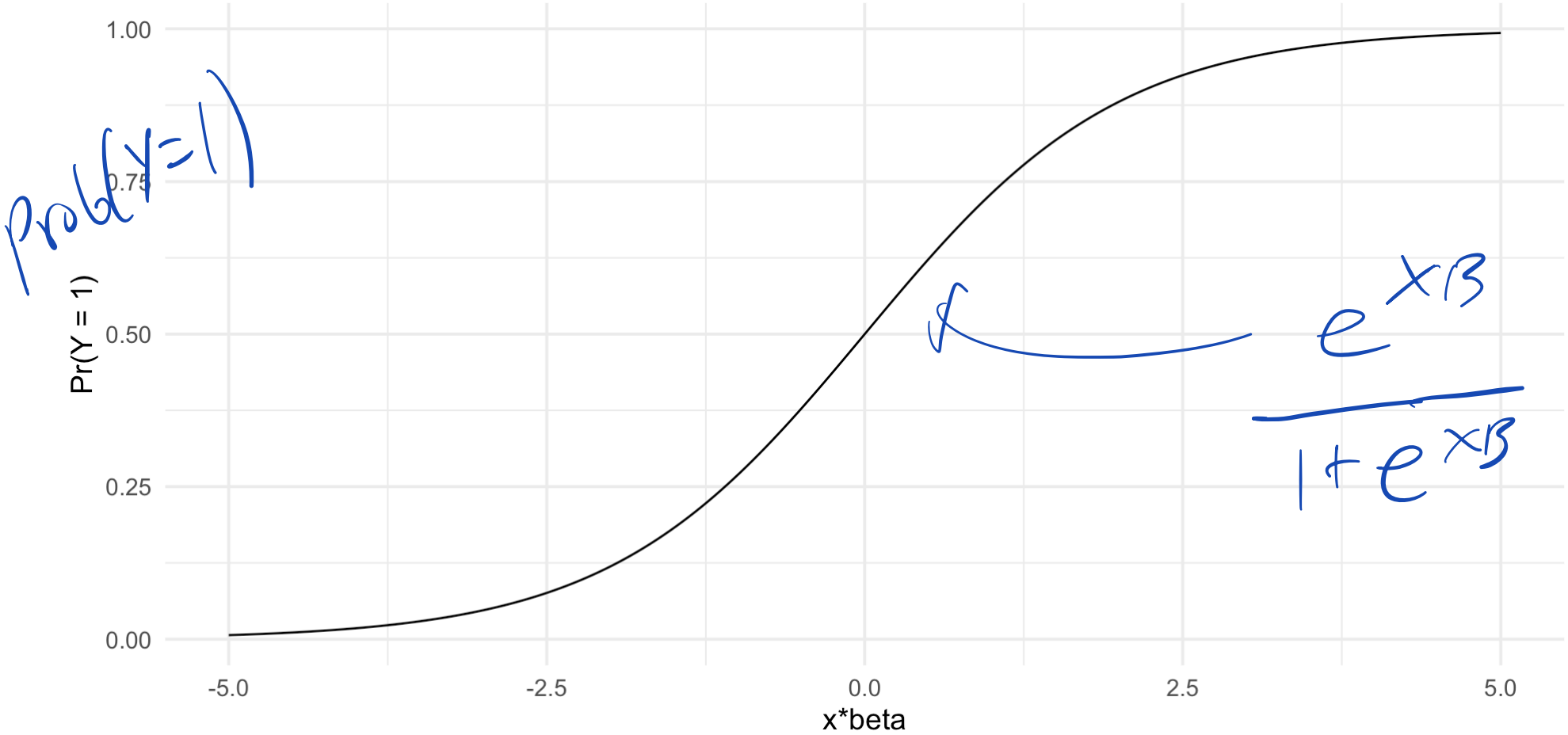
$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right) = x'\beta \quad \Longleftrightarrow \quad P(Y=1) = ??$$

$$p/1-p = e^{x\beta} \Rightarrow p = \frac{e^{x\beta}}{1+e^{x\beta}}$$

$$P(y|x) = \text{inv-logit}(x'\beta) = \frac{e^{x\beta}}{1+e^{x\beta}}$$

Logistic regression model

The logistic function looks like this:



Assumptions

The model makes two key assumptions:

1. the probability of the outcome changes monotonically with each explanatory variable
2. observations are independent (used to obtain a joint distribution)

↑ same as in in lm.

3. Pr(Y|X) follows the inv-logit form.

Estimation

The model is fit by maximum likelihood: find the parameters for which the observed data are most likely. The likelihood (joint distribution) is constructed from the model and the Bernoulli distribution.

```
# fit model
fit <- glm(y ~ Male + Age + BMI, family = binomial(link = "logit"), data = diabetes)

# parameter estimates
coef_tbl <- tibble(
  estimate = coef(fit),
  `standard error` = sqrt(diag(vcov(fit)))
)
rownames(coef_tbl) <- names(coef(fit))
kable(coef_tbl)
```

specifies logistic regression specifically.

	estimate	standard error
	-8.1992313	0.3575649
	0.2703777	0.1177993
	0.0532001	0.0035124
	0.1006066	0.0078616

Gender.

"If you're male, the log odds are .27 higher."

log odds scale.

Age: The odds of diabetes multiply by $e^{.053}$ for a 1 year increase.

Parameter interpretations

Similar to linear regression, coefficients give the change in log-odds associated with incremental changes in the explanatory variables.

On the scale of the linear predictor:

> A one-unit increase in BMI is associated with an estimated 0.1 increase in log odds of diabetes after adjusting for age and sex

On the scale of the odds:

> A one-unit increase in BMI is associated with an estimated 10% increase in the odds of diabetes after adjusting for age and sex

On the probability scale, the increase depends on the starting value of BMI.

Confidence intervals

One can also give confidence intervals. These are based on large-sample approximations.

```r
# confidence intervals
ci <- confint(fit)
exp(ci["BMI", ])
```

```
   2.5 %   97.5 % 
1.089027 1.123143 
```

> With 95% confidence, each 1-unit increase in BMI is associated with an estimated increase in odds of diabetes between 8.9% and 12.3% after adjusting for age and sex

Fitted values (predictions)

The fitted values for logistic regression are fitted *probabilities* (not outcomes).

$$\hat{p_i} = \frac{1}{1 + e^{-x_i'\hat{\beta}}}$$

In R, we can get linear predictor values (log-odds):

```
1  # log odds
2  head(fit$linear.predictors, 5)
```

```
         1          2          3          4          5 
-5.4435581 -2.5614276 -0.2622829 -0.2622829 -5.9827228 
```

$$\hat{p} = \frac{e^{X\beta}}{1+e^{X\beta}} \frac{e^{-X\beta}}{e^{-X\beta}} = \frac{1}{e^{-X\beta}+1}$$

Fitted values

To obtain probabilities, one could manually back-transform:

```
# compute fitted probabilities 'by hand'
fitted_probs <- 1/(1 + exp(-fit$linear.predictors))
head(fitted_probs, 5)
```

```
          1           2           3           4           5
0.004305453 0.071662511 0.434802598 0.434802598 0.002515606
```

Or more simply just get the fitted.values

```
# fitted probabilities
head(fit$fitted.values, n=5)
```

```
          1           2           3           4           5
0.004305453 0.071662511 0.434802598 0.434802598 0.002515606
```

Classification

For each observation (or new observations), probabilities can be computed directly from the fitted model:

$$\hat{p_i} = \frac{1}{1 + e^{-x_i' \hat{\beta}}}$$

But what if we want to classify a person as diabetic or not diabetic? Should we declare a case when…

- more probable than not: $\hat{p} > 0.5$?
- highly probable, say $\hat{p} > 0.8$?
- somewhat probable, say $\hat{p} > 0.2$?

set $\hat{y} = 1$?

set $\hat{y} = 1$?

Sensitivity

If we use a *low* threshold for classification, say:

$$\hat{Y} = 1 \quad \Longleftrightarrow \quad \hat{p} > 0.1$$

Then the classifications will be more *sensitive* to cases – most cases of diabetes will be correctly classified.

Specificity

If we use a *high* threshold instead, say:

$$\hat{Y} = 1 \quad \iff \quad \hat{p} > 0.9$$

Then the classification will not be very sensitive to cases, but they will be fairly *specific* – classifications will be correct for most people without diabetes.

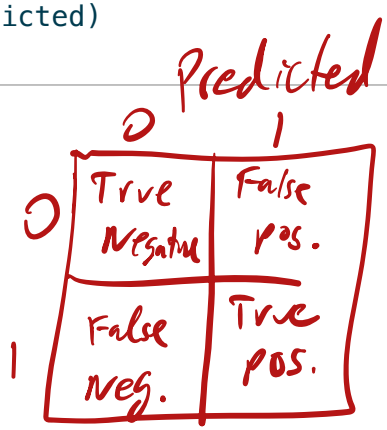
Cross-tabulation

For any given classification threshold, we can cross-tabulate the classifications with the observed outcomes:

```
1  # confusion matrix with threshold 0.5
2  threshold <- 0.5
3  predicted <- as.integer(fitted(fit) > threshold)
4  conf_matrix <- table(Observed = y, Predicted = predicted)
5  conf_matrix
```

```
          Predicted
Observed    0    1
       0 4446   19
       1  358    8
```

- rows are observed outcomes
- columns are predicted outcomes



Misclassification Rate:

$$\frac{\#\text{ of mis. class.}}{\text{total obs.}}$$

Using the more-likely-than-not criterion is very *specific* (high true negative rate) but not at all *sensitive* (low true positive rate).

Overall accuracy is misleading

The proportion of correctly classified observations is:

```
# proportion of correctly classified observations
sum(diag(conf_matrix)) / sum(conf_matrix)
```

```
[1] 0.9219623
```

This looks really good, but any method that classifies all or most observations as non-diabetic will achieve high accuracy because of the case imbalance in the data.

```
# proportion of non-diabetic respondents
mean(y == 0)
```

```
[1] 0.9242393
```

Use class-wise error rates

Examining class-wise error rates reveals how asymmetric the classifications are:

▶ Code

```
        Predicted
Observed           0           1
       0 0.995744681 0.004255319
       1 0.978142077 0.021857923
```

- same layout as confusion matrix, but with entries divided by the total number of outcomes in each class
- note 97.8% error rate among diabetes cases

A better classifier

In this case we can do better by choosing a low classification threshold $\hat{p} > 0.1$:

▶ Code

```
        Predicted
Observed    0    1
       0 3473  992
       1  105  261
```

- higher overall error rate $\frac{1097}{4831} = 0.227$
- but about 70% accurate within each class (diabetic and non-diabetic)

Class-wise errors:

▶ Code

```
        Predicted
Observed         0         1
       0 0.7778275 0.2221725
       1 0.2868852 0.7131148
```

Characterizing Misclassifications

Sources: [8][9][10][11][12][13][14][15] view · talk · edit

		Predicted condition			
	Total population $= P + N$	**Predicted positive**	**Predicted negative**	Informedness, bookmaker informedness (BM) $= TPR + TNR - 1$	Prevalence threshold (PT) $= \frac{\sqrt{TPR \times FPR} - FPR}{TPR - FPR}$
Actual condition	**Positive (P)** [a]	**True positive** (TP), hit[b]	**False negative** (FN), miss, underestimation	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power $= \frac{TP}{P} = 1 - FNR$	False negative rate (FNR), miss rate type II error [c] $= \frac{FN}{P} = 1 - TPR$
	Negative (N)[d]	**False positive** (FP), false alarm, overestimation	**True negative** (TN), correct rejection[e]	False positive rate (FPR), probability of false alarm, fall-out type I error [f] $= \frac{FP}{N} = 1 - TNR$	True negative rate (TNR), specificity (SPC), selectivity $= \frac{TN}{N} = 1 - FPR$
	Prevalence $= \frac{P}{P + N}$	Positive predictive value (PPV), precision $= \frac{TP}{TP + FP} = 1 - FDR$	False omission rate (FOR) $= \frac{FN}{TN + FN} = 1 - NPV$	Positive likelihood ratio (LR+) $= \frac{TPR}{FPR}$	Negative likelihood ratio (LR−) $= \frac{FNR}{TNR}$
	Accuracy (ACC) $= \frac{TP + TN}{P + N}$	False discovery rate (FDR) $= \frac{FP}{TP + FP} = 1 - PPV$	Negative predictive value (NPV) $= \frac{TN}{TN + FN} = 1 - FOR$	Markedness (MK), deltaP (Δp) $= PPV + NPV - 1$	Diagnostic odds ratio (DOR) $= \frac{LR+}{LR-}$
	Balanced accuracy (BA) $= \frac{TPR + TNR}{2}$	F_1 score $= \frac{2\,PPV \times TPR}{PPV + TPR} = \frac{2\,TP}{2\,TP + FP + FN}$	Fowlkes–Mallows index (FM) $= \sqrt{PPV \times TPR}$	Matthews correlation coefficient (MCC) $= \sqrt{TPR \times TNR \times PPV \times NPV} - \sqrt{FNR \times FPR \times FOR \times FDR}$	Threat score (TS), critical success index (CSI), Jaccard index $= \frac{TP}{TP + FN + FP}$