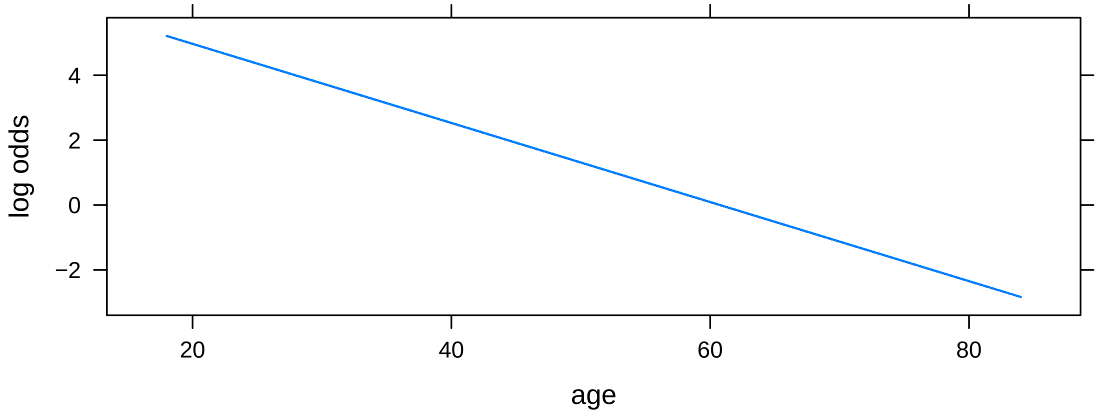
Agenda

1. Logistic Regression
2. Assessing Fit in Logistic Regression

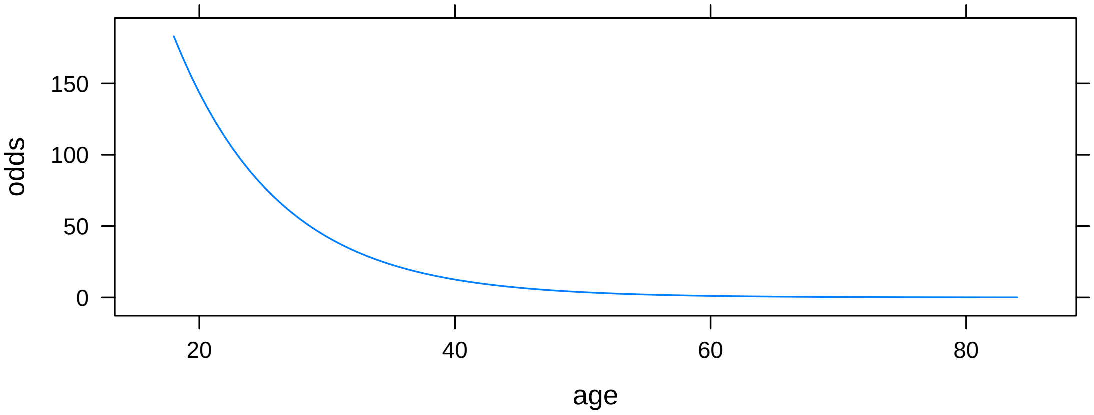
Binary response

- What do to when response variable p is *binary*?
- Linear model will produce illogical estimates (eg. $\hat{p} > 1$ or $\hat{p} < 0$)

```r
require(mosaic)
require(Stat2Data)
data(Whickham)
Whickham = Whickham %>%
  mutate(isAlive = 2 - as.numeric(outcome))
xyplot(isAlive~jitter(age), data=Whickham, pch=19, cex=1.5, alpha=0.05, col="black")
```



```r
plotModel(lm(isAlive~age, data=Whickham), pch=19)
```



This doesn't make sense outside the $[0, 1]$ range. One solution might be to summarize the data to be frequencies at each age.

```
alive = Whickham %>%
  group_by(age, isAlive) %>%
  summarize(total = n()) %>%
  mutate(freq = total / sum(total)) %>%
  filter(isAlive==1)

## Error in summarize(., total = n()):  argument "by" is missing, with no default

xyplot(freq˜age, data=alive,pch=19, col="black")

## Error in eval(substitute(groups), data, environment(x)):  object ’alive’ not found
```

```
xyplot(freq˜age, data=alive, pch=19, type=c("p", "r"))

## Error in eval(substitute(groups), data, environment(x)):  object ’alive’ not found
```

But, this still has strange interpretations.

Logistic Regression What’s the solution? Logistic regression! This uses the logit function as a ‘link.’

- logit produces S-curve that is always in $[0, 1]$
- Fit via *maximum likelihood estimation*, not OLS
- No such thing as R^2 or sum of squares

Warmup– probability and odds Probabilities and odds express the same information, but have different interpretation. Lets fill in this chart to help warm up our intuition about their relationship.

Probability of success (π)	Odds ($\pi/(1-\pi)$)
1/2	1/1
1/3	1/2
1/4	
1/5	
2/3	
3/4	

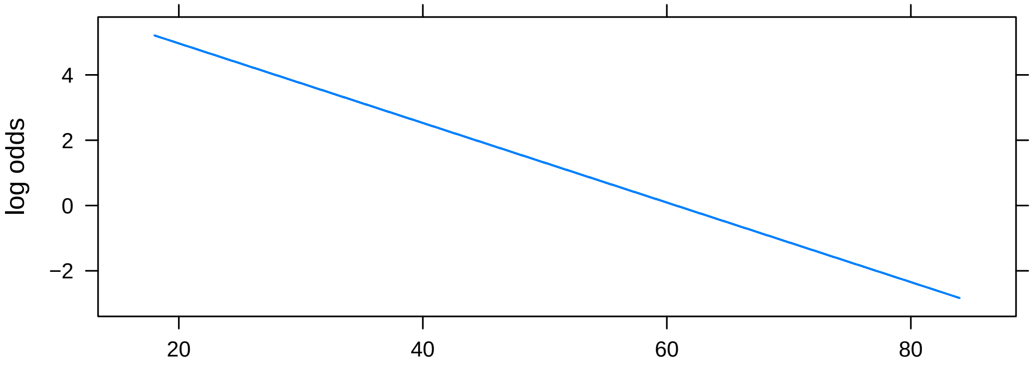
"Spaces" We often talk about three 'spaces' for logistic regression. These are just different ways of writing the same thing, but they have different interpretations so they are useful for different tasks.

- Log odds space

$$\log\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 \cdot X$$

Thinking about log odds is useful when you want a linear form of a regression line. You can interpret the coefficients in the standard way we have been doing for linear regression, "A one unit increase in x is associated with a β_1 increase in the log odds of y"

```
m1 = glm(isAlive~age, data=Whickham, family=binomial)
xyplot(log(fitted.values(m1)/(1-fitted.values(m1)))~age, data=Whickham, type=c("l"),ylab="log odds")
```

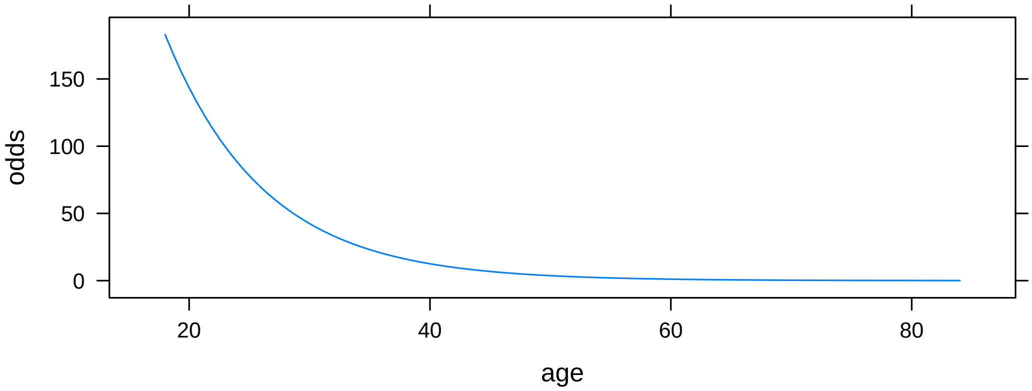


- Odds space

$$\frac{\pi}{1-\pi} = e^{\beta_0+\beta_1 \cdot X}$$

Odds are useful when you want to interpret the slope coefficient. We can use the interpretation, "A one unit increase in x is associated with changing y by a factor of e^{β_1}.

```
xyplot(fitted.values(m1)/(1-fitted.values(m1))~age, data=Whickham,  type="spline", ylab="odds")
```

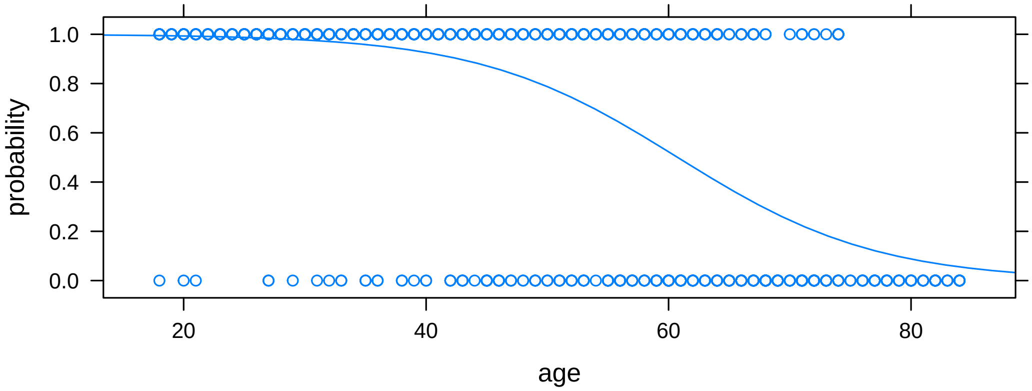


- Probability space

$$\pi = \frac{e^{\beta_0+\beta_1 \cdot X}}{1+e^{\beta_0+\beta_1 \cdot X}}$$

The probability form is how the model gets fit, but it does not have an easy interpretation for what happens with a change in x.

```
plotModel(m1, ylab="probability")
```


probability
age