Confidence in Models

Author: Nicholas G Reich

This material is part of the **statsTeachR** *project*

Today's Lecture

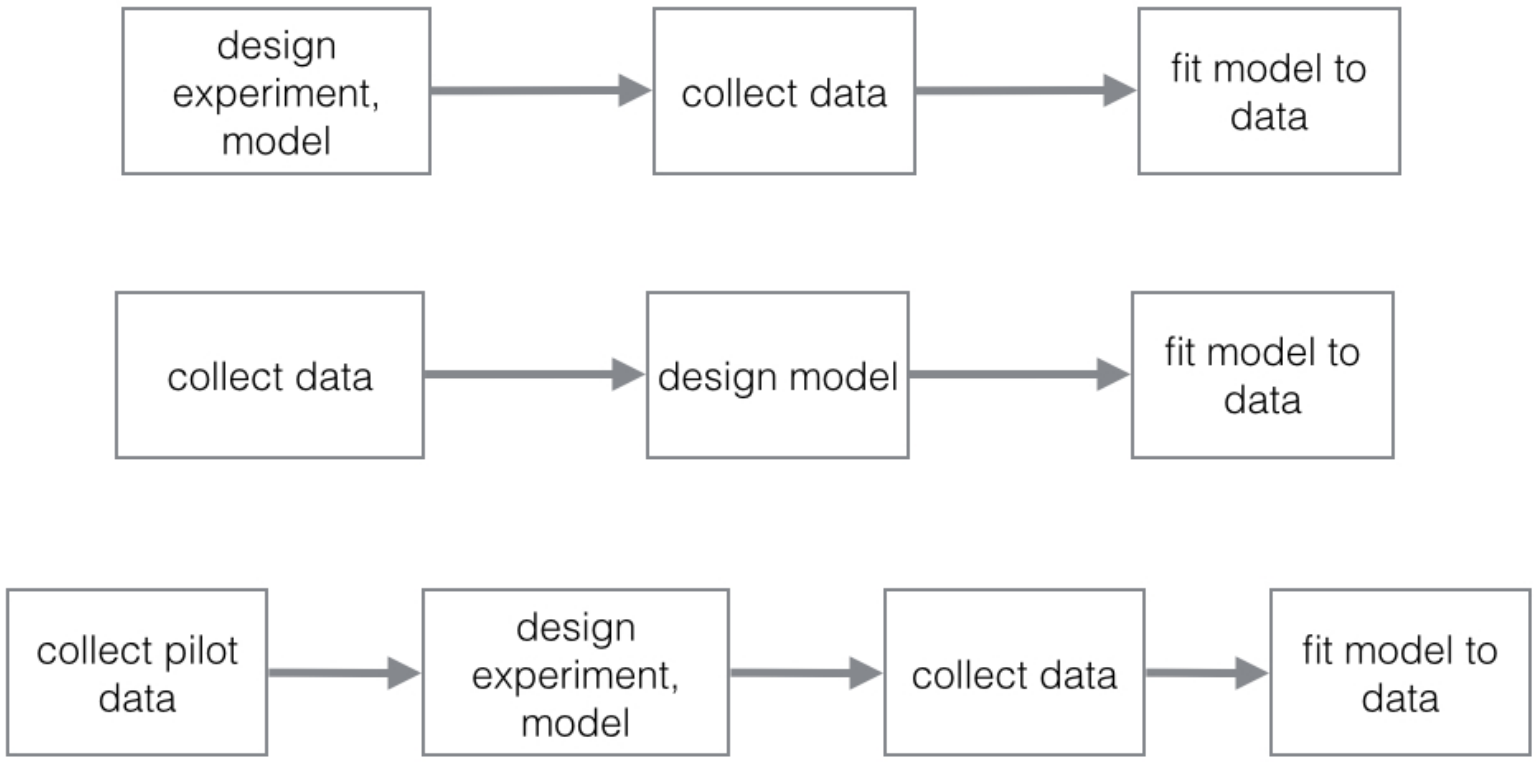
It aint what you dont know that gets you into trouble. Its what you know for sure that just aint so. -Mark Twain

Today's central question

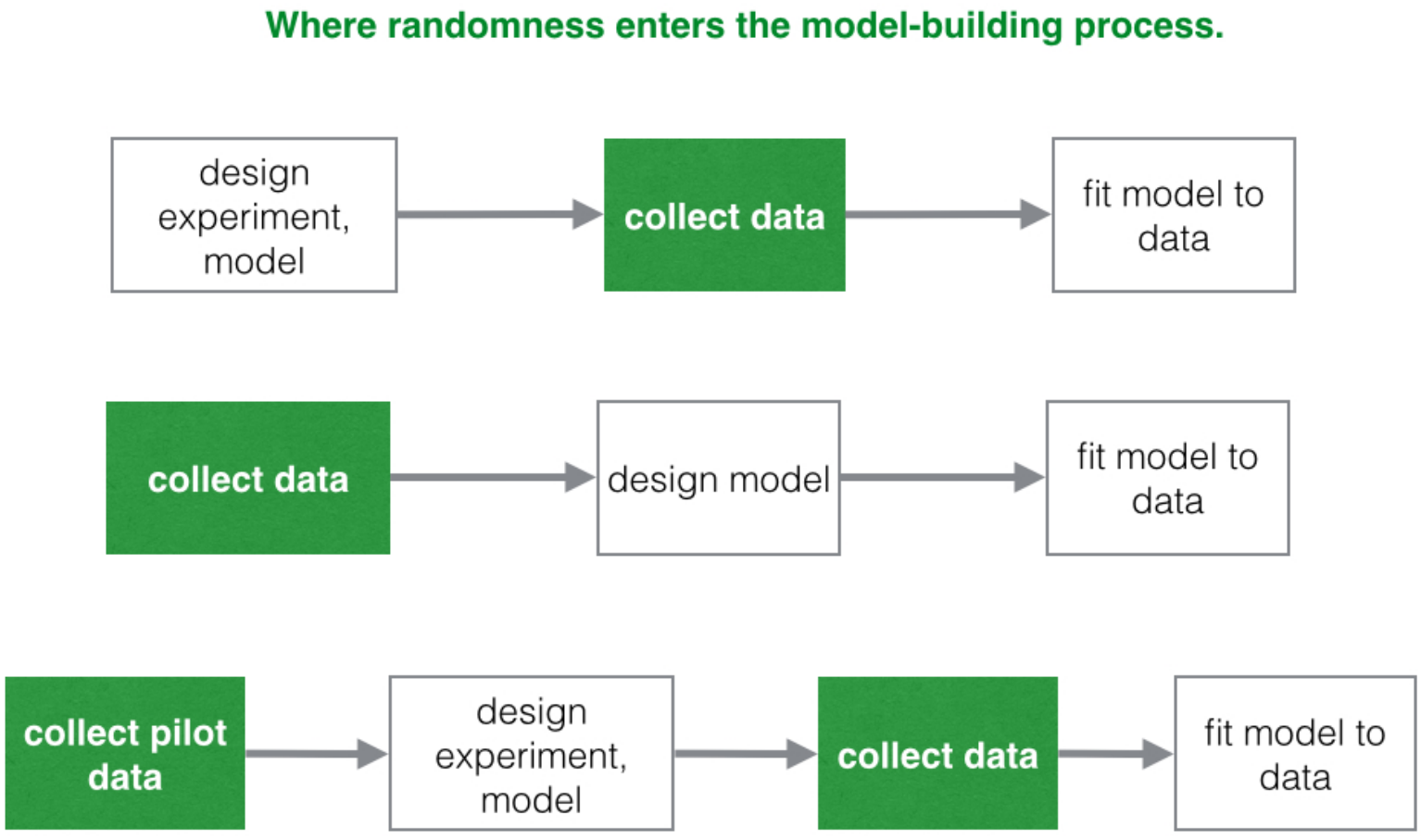
What do linear regression models tell us about what we know and do not know about a particular dataset?

Based loosely on Kaplan, Chapter 12.

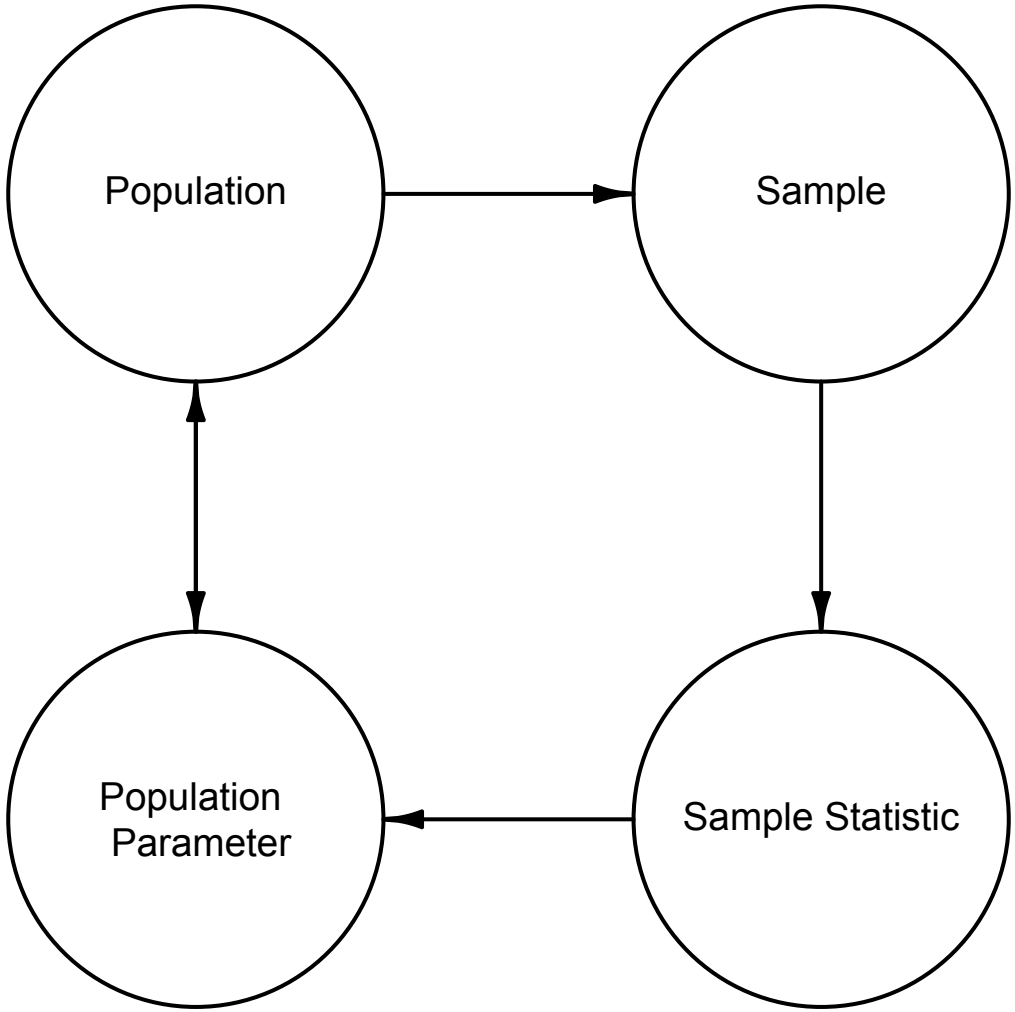
Process of building a statistical model



Process of building a statistical model



Circle of Life



How much will a sample tell us about the population

In practice we can very rarely sample the entire population of interest.
We can create a simple example of a population as a illustration. E.g. 8636 running times for the Cherry Blossom Ten Mile race in Washington DC in 2005:

```
race <- fetch::fetchData("ten-mile-race.csv")
head(race)
```

```
##   state time  net age sex
## 1    VA 6060 5978  12   M
## 2    MD 4515 4457  13   M
## 3    VA 5026 4928  13   M
## 4    MD 4229 4229  14   M
## 5    MD 5293 5076  14   M
## 6    VA 6234 5968  14   M
```

A simple model for the race data

$$net \sim age + sex$$

or

$$net = \beta_0 + \beta_1 \cdot age + \beta_2 \cdot sex$$

Using all the data, i.e. the entire "population"

```
fm <- lm(net ~ age + sex, data=race)
coef(fm)

## (Intercept)         age        sexM
##  5339.15545    16.89362  -726.61948
```

A sample of the race data gives different results

$$net = \beta_0 + \beta_1 \cdot age + \beta_2 \cdot sex$$

```
coef(fm)

## (Intercept)         age        sexM
##  5339.15545    16.89362  -726.61948
```

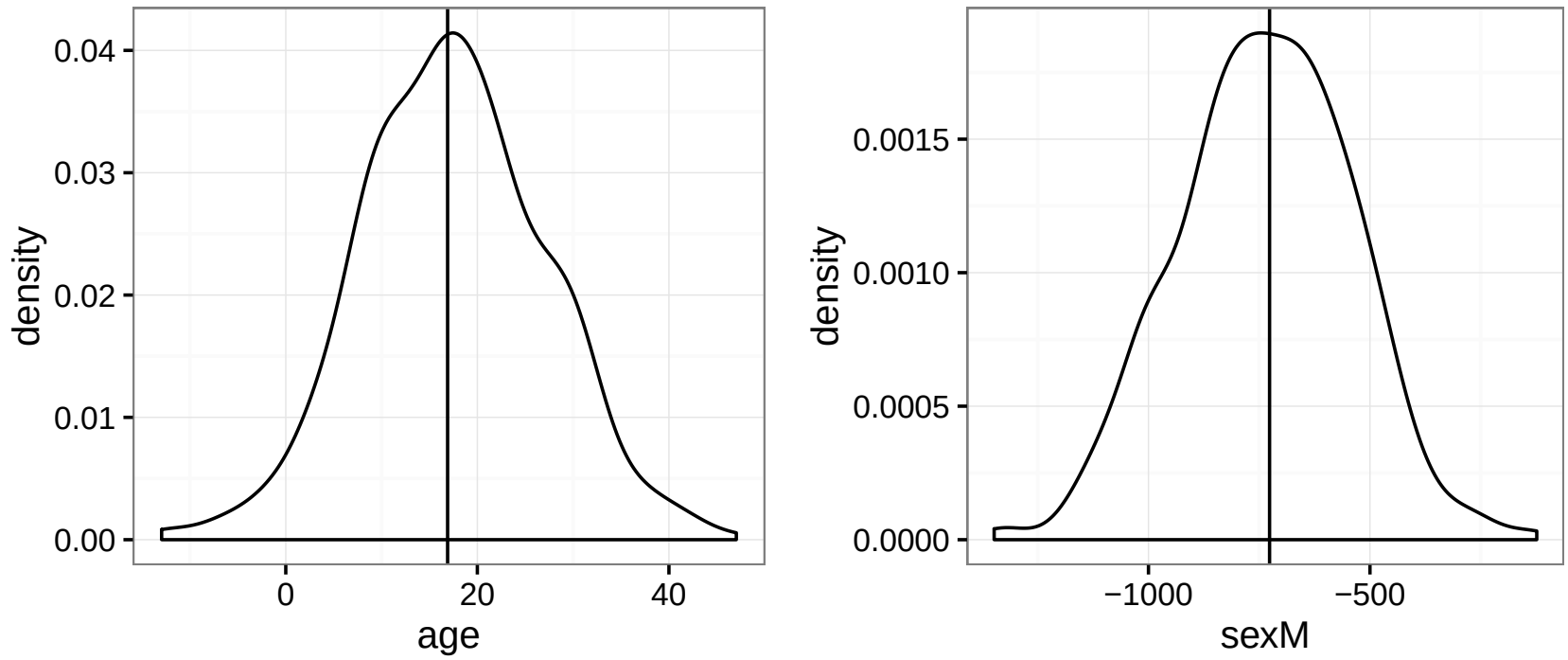
Using a sample of 100 times from the population:

```
library(mosaic)
s <- do(500) * lm(net ~ age + sex, data=sample(race, 100))
head(s[,1:5])

##   Intercept      age       sexM     sigma  r.squared
## 1  5104.653 29.05782  -638.6229 1166.5269 0.09386305
## 2  5318.716 14.82012  -541.6456  783.5531 0.09576590
## 3  4854.663 31.00987 -1114.1963  931.8768 0.26550805
## 4  5258.390 11.05543  -379.4704  870.4800 0.04309318
## 5  5436.315 12.77006  -657.4816  951.1627 0.09701434
## 6  5321.410 16.16511  -653.1509  852.3441 0.13417287
```

The sampling distribution of the βs

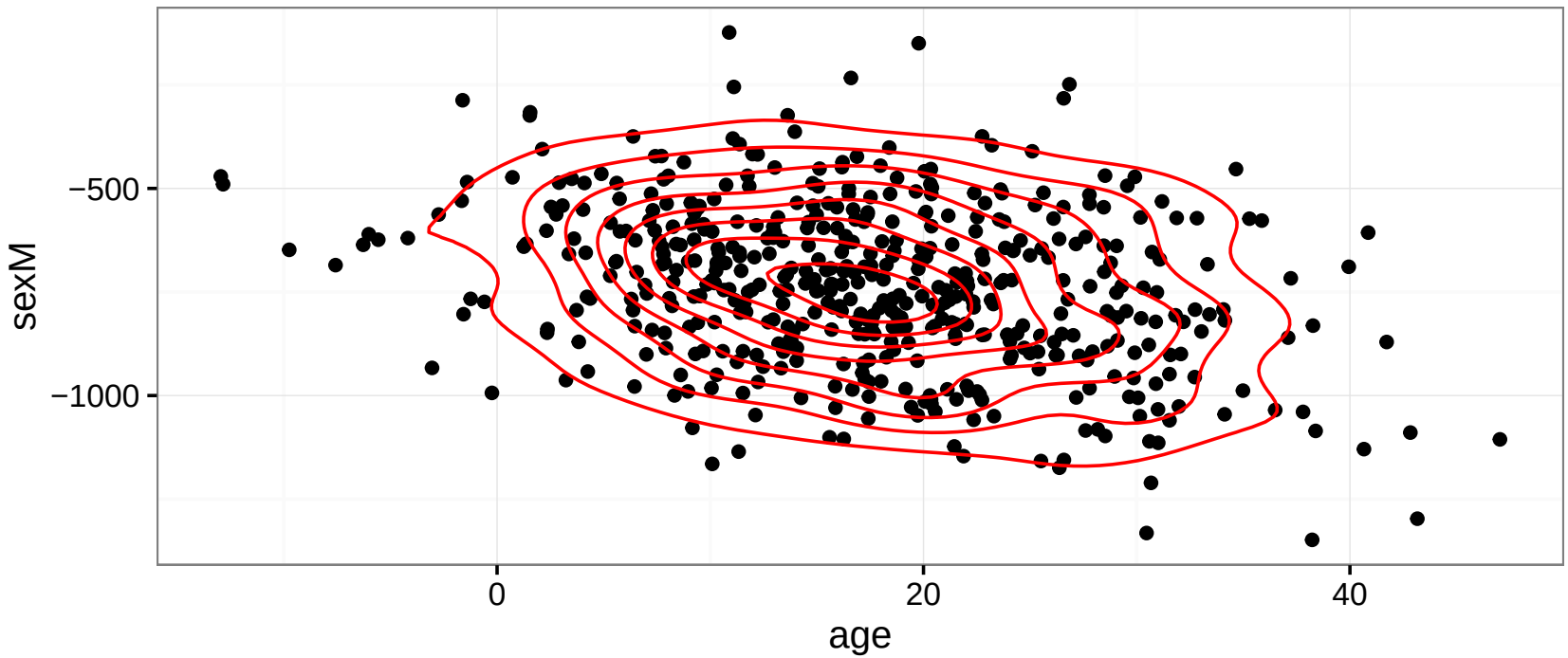
```
library(gridExtra)
p1 <- ggplot(s) + geom_density(aes(x=age)) +
    geom_vline(xintercept=coef(fm)["age"])
p2 <- ggplot(s) + geom_density(aes(x=sexM)) +
    geom_vline(xintercept=coef(fm)["sexM"])
grid.arrange(p1, p2, nrow=1)
```



The sampling distribution of the βs

Important to note that the sampling distributions may be correlated with each other.

```
ggplot(s, aes(x=age, y=sexM)) +
    geom_point() + geom_density_2d(color='red')
```



Sampling distribution terminology

Really important vocabulary!

- **sampling distribution**: the distribution of an estimated parameter, reflecting the randomness of the sampling (data collection) process.
- **standard error**: the standard deviation of a sampling distribution, measures the precision of our estimate or the amount of information we have about the parameter.
- **point estimate**: the exact numerical value that represents our best guess at the true parameter value. (In regression, this is the least-squares estimate of our β.)

The standard error depends on...

The quality of the data

- If your data collection process involves a measurement process that contains a lot of error, how will that impact the standard errors?
- In the setting of the race, what measurement procedures might lead to less or more error?

The standard error depends on...

The quality of the model

Models with lower residual error tend to have _____ standard errors than ones with larger residual error.

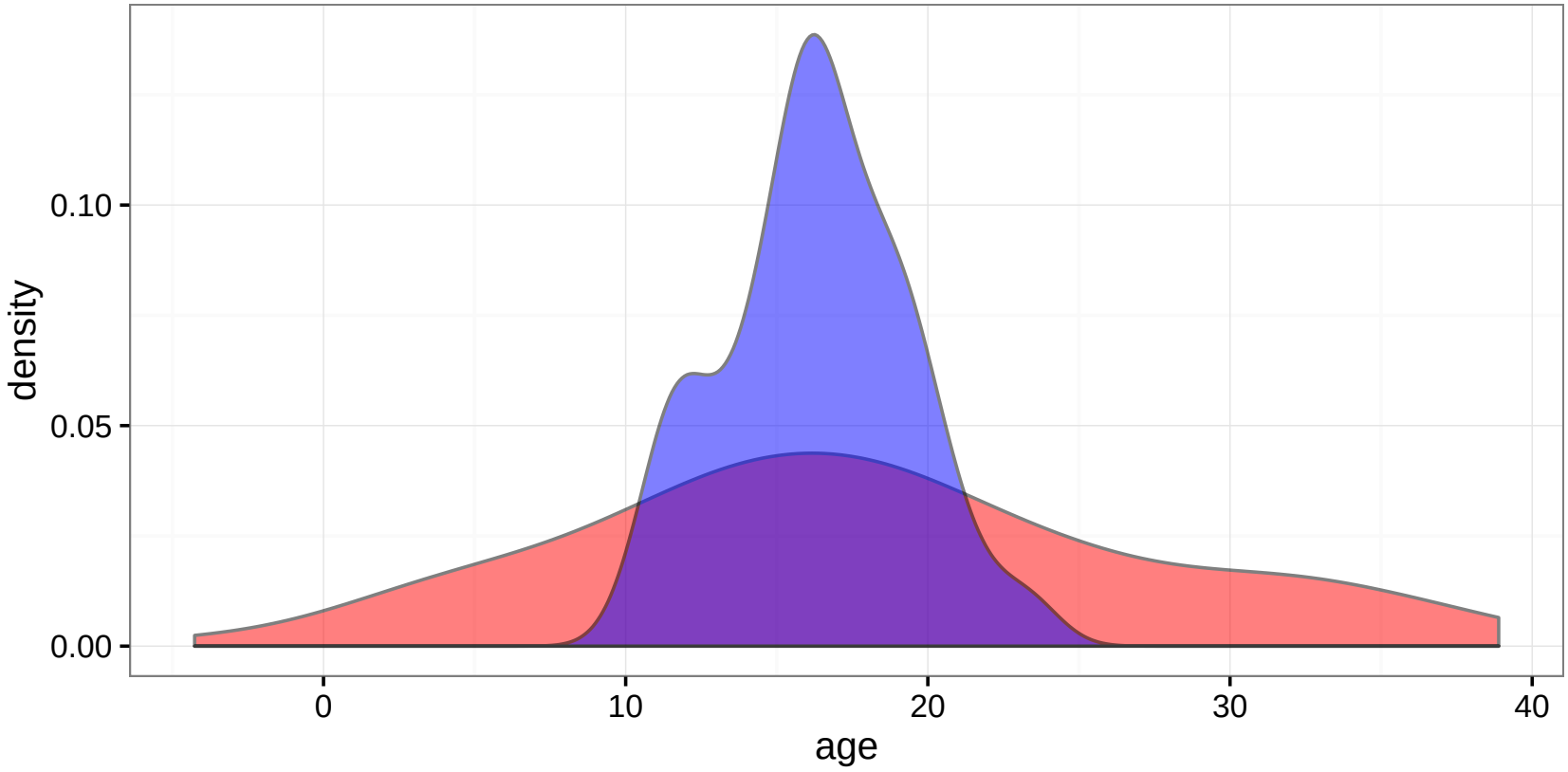
The standard error depends on...

The sample size

As the sample size increases, what happens to the standard error?

The standard error and sample size

```
s100 <- do(100) * lm(net ~ age + sex, data=sample(race, 100))
s1000 <- do(100) * lm(net ~ age + sex, data=sample(race, 1000))
ggplot() + geom_density(aes(x=age), fill="red", alpha=.5, data=s100) +
    geom_density(aes(x=age), fill="blue", alpha=.5, data=s1000)
```



The standard error and sample size (con't)

The formula for the standard error is proportional to $1/\sqrt{n}$. This is kind of a slow decrease: "to make the standard error 10 times smaller you need to make the dataset 100 times larger"!

And now, back to our true sample

In reality, we don't have the luxury of measuring the entire population!

- We can use information in the original sample to make a good guess at what the sampling distribution is (see Kaplan Ch 5.2).
- The guess is based on an approximation that has good properties when the assumptions of our model aren't broken.

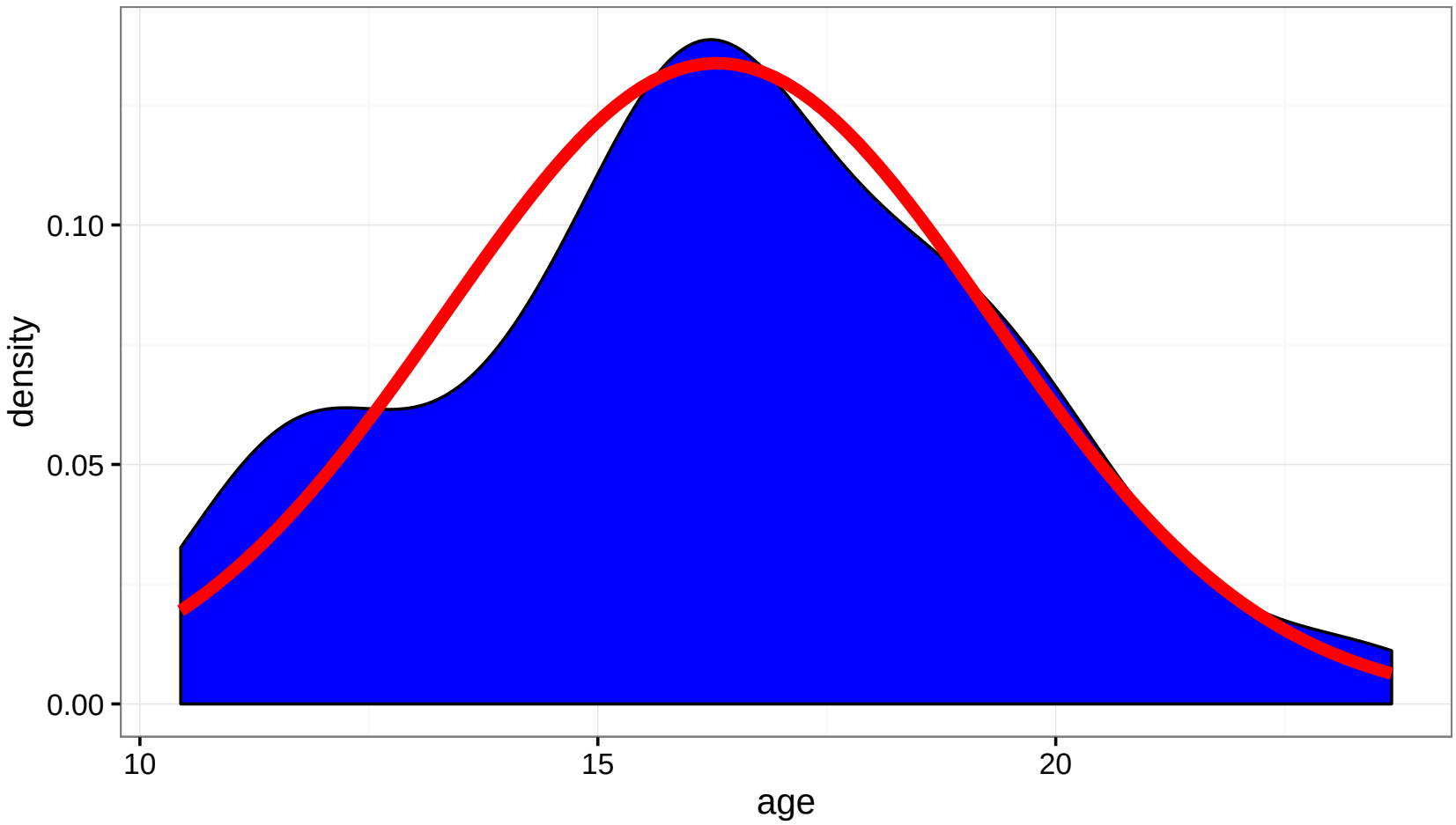
```
fm <- lm(net ~ age + sex, data=race)
summary(fm)$coef
```

```
##                Estimate Std. Error   t value      Pr(>|t|)
## (Intercept) 5339.15545 35.0486629 152.33550  0.000000e+00
## age           16.89362  0.9443776  17.88863  2.660668e-70
## sexM        -726.61948 20.0181263 -36.29808 1.281442e-268
```

Full data vs. partial data sampling distribution

Our estimated sampling distribution for 1000 observations (in blue) vs. the estimated sampling distribution for all data (in red).

```
ggplot(s1000, aes(x=age)) + geom_density(fill='blue') +
    stat_function(fun = dnorm, lwd = 2, col = 'red',
                  args = list(mean = mean(s1000$age), sd = sd(s1000$age
```



Confidence Interval

A confidence interval summarizes our uncertainty about a point estimate.

- For example: "our analysis suggests that the age coefficient in the model is 17 $\pm$ 2, with 95% confidence."
- More precisely, we could do the calculation as: 16.9 $\pm$ 2*0.94.
- We multiply the standard error by two because this approximates a 95% coverage interval of the sampling distribution.
- NOTE: when your sample size is very small (e.g. $n < 20$) the multiplier of 2 is misleading, and larger values should be used. See, e.g. Table 12.1 in Kaplan.

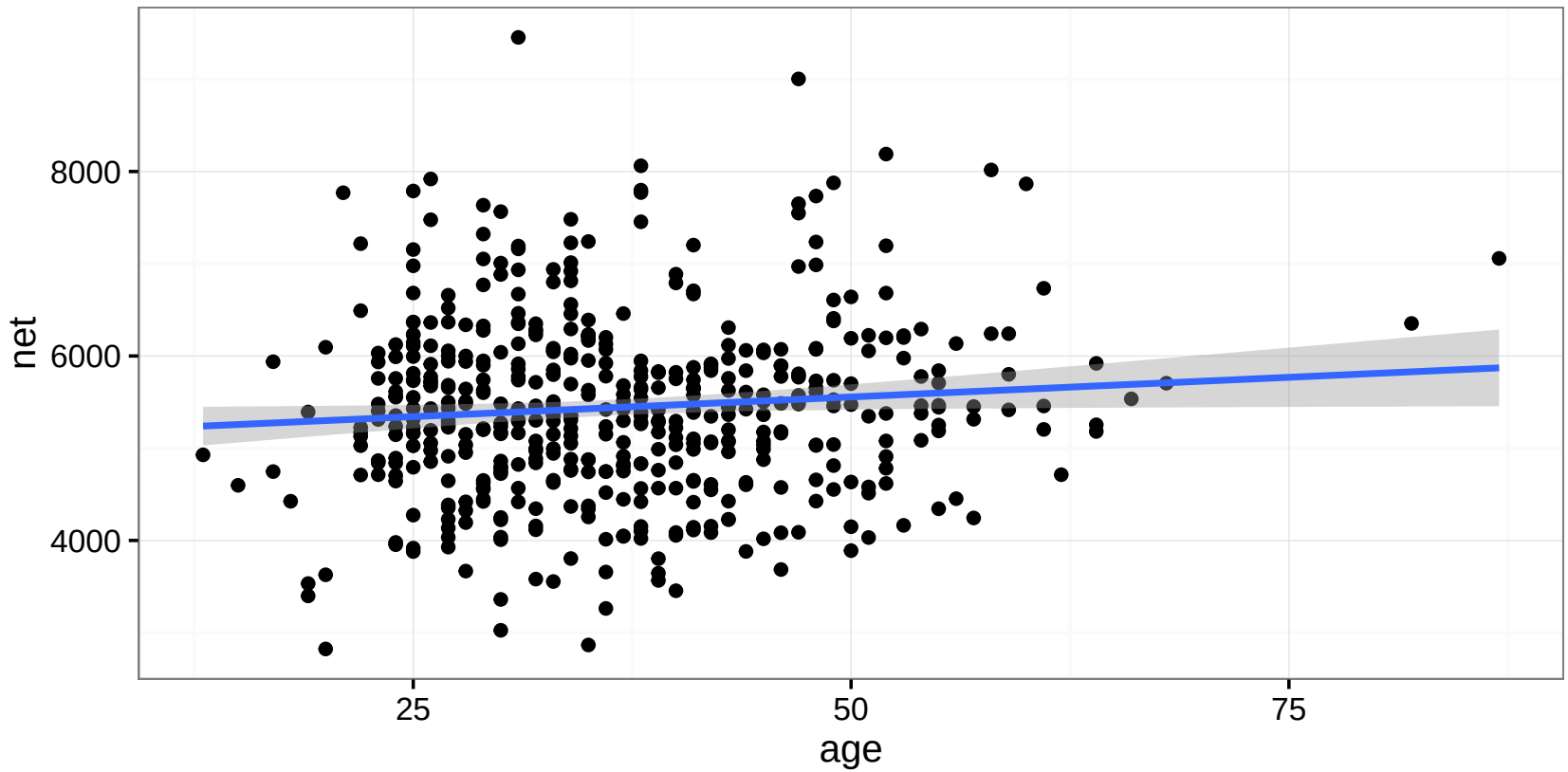
```
summary(fm)$coef["age",]

##     Estimate   Std. Error      t value     Pr(>|t|)
## 1.689362e+01 9.443776e-01 1.788863e+01 2.660668e-70
```

Confidence in predictions

Confidence intervals are not appropriate for making predictions about individual data-points!

```
r <- sample(race, 500)
qplot(age, net, data=r) + geom_smooth(method="lm")
```



E.g. 95% of 60-year-olds will not have times within $\pm$ 200 of the predicted value ($\sim$ 5900?).

Confidence in predictions

Confidence intervals for regression coefficients represent the uncertainty in the coefficient, but not in the predictions at certain, fixed values. Recall that the line has to pass through the point $(\bar{x}, \bar{y})$. Small changes in slope/intercept will have minimal changes to where the line passes near that fulcrum, and larger changes at the fringes.