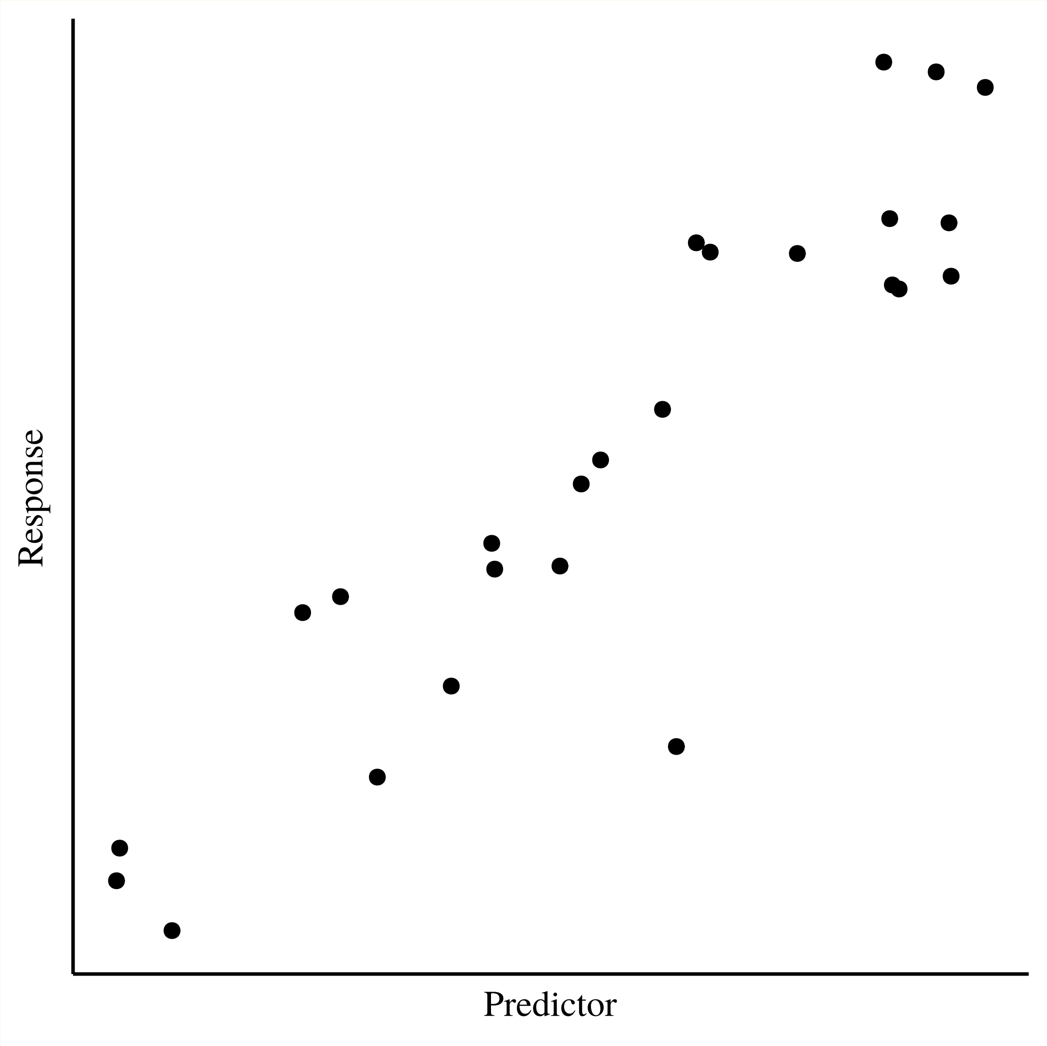
Simple Linear Regression (SLR) Model Inference

Stat 230

April 04 2022

Overview



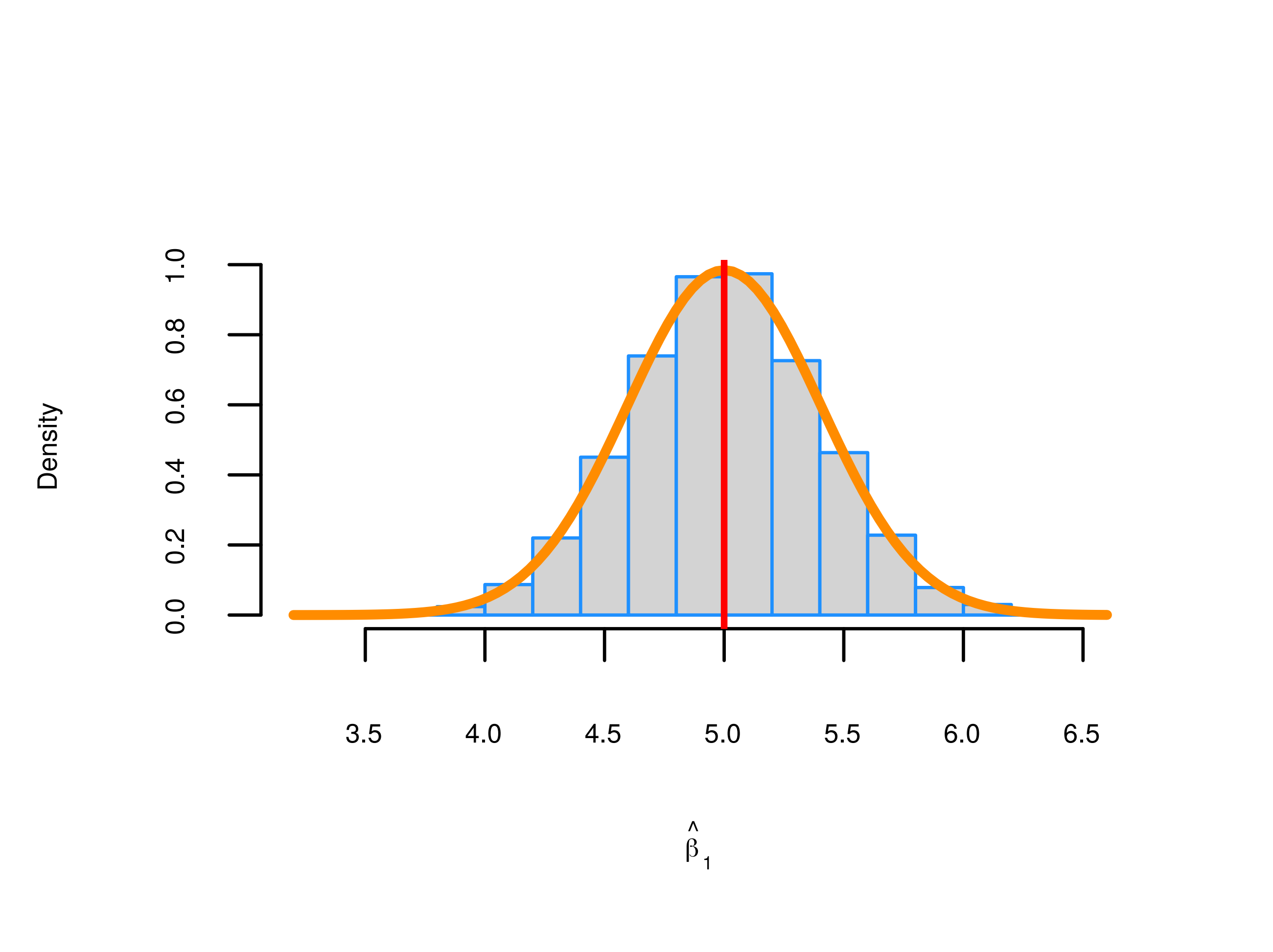
Today:

Inference for the SLR model

- mean parameters
- mean value
- predicted value

Sampling Distribution: Simulation results

```r
set.seed(123)  # just makes simulation reprodu
n <- 10000
beta_0 = -20
beta_1 = 5
sigma = 15
x <- cars$speed
s.x = sd(x)
sd_beta_1_hat <- sigma*sqrt(1/((50-1)*s.x^2))
slopes <- replicate(n, sim_slr(x, beta_0, beta
hist(slopes, prob = TRUE, breaks = 20,
     xlab = expression(hat(beta)[1]),
     main = "",
     border = "dodgerblue",
     cex.lab=0.5, cex.axis=0.5)
curve(dnorm(x, mean = beta_1, sd = sd_beta_1_h
      col = "darkorange", add = TRUE, lwd = 3)
abline(v=5,col="red", lwd=2)
```



SLR: Hypothesis test

$$H_0 : \beta_j = 0 \quad H_A : \beta_j \neq 0$$

t-test statistic:

$$\frac{\hat{\beta}_j - 0}{SE(\hat{\beta}_j)}$$

Two tailed p-value computed from the t-distribution with $n - 2$ degrees of freedom:

$$\text{p-value} = 2 \times P(T > |t|)$$

- If H_A is directional (e.g. $<$ or $>$), then compute one-tailed p-value.

Testing mean parameters in R

```
cars_lm <- lm(dist ~ speed, data = cars)
summary(cars_lm)
```

```

Call:
lm(formula = dist ~ speed, data = cars)

Residuals:
    Min      1Q  Median      3Q     Max 
-29.069  -9.525  -2.272   9.215  43.201 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -17.5791     6.7584  -2.601   0.0123 *  
speed         3.9324     0.4155   9.464 1.49e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 15.38 on 48 degrees of freedom
Multiple R-squared:  0.6511,	Adjusted R-squared:  0.6438 
F-statistic: 89.57 on 1 and 48 DF,  p-value: 1.49e-12
```

Testing mean parameters in R

```
              Estimate Std. Error   t value     Pr(>|t|)
(Intercept) -17.579095  6.7584402 -2.601058 1.231882e-02
speed         3.932409  0.4155128  9.463990 1.489836e-12
```

$$H_0 : \beta_1 = 0 \quad H_A : \beta_1 \neq 0$$

$$t = \frac{3.9324 - 0}{0.415513} = 9.4640$$

```
2*(1-pt(9.4640, df = 50 - 2)) # need left area
[1] 1.489919e-12
```

The estimated effect of speed on stopping distance is statistically significant at the level of 5% significance $(t = 9.4640, p = 1.49 * 10^{-12} \approx 0)$.

SLR: $C\%$ Confidence Intervals

$$\hat{\beta}_j \pm t^* SE(\hat{\beta}_j)$$

t^* is the $(100 - C)/2$ percentile from the t-distribution with $df = n - 2$ degrees of freedom

R code:

```
confint(cars_lm)
```

```
                 2.5 %    97.5 %
(Intercept) -31.167850 -3.990340
speed         3.096964  4.767853
```

CI for mean parameters

$$95\% \text{ CI} : 3.932409 \pm t^*(0.4155128)$$

```
qt(0.975, df = 50 -2)
[1] 2.010635
```

```
                  2.5 %    97.5 %
(Intercept) -31.167850 -3.990340
speed         3.096964  4.767853
```

$$95\% \text{ CI: } 3.932409 \pm 2.0106 * (0.4155128) = (3.0970, 4.7679)$$

```
cars_lm$coefficients[2] + c(-1,1)*qt(0.975, df = 50 -2)*summary(cars_lm)[[4]][2,2]
[1] 3.096964 4.767853
```

We are 95% confident that a 1 miles per hour increase in cars speed is associated with a 3.0970 to 4.7679 ft increase in average stopping distance.

R broom package

```
# useful for presenting a tidy summary of a model
library(broom)
tidy(cars_lm, conf.int = TRUE)
# A tibble: 2 × 7
  term        estimate std.error statistic  p.value conf.low conf.high
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>    <dbl>     <dbl>
1 (Intercept)   -17.6      6.76      -2.60 1.23e- 2   -31.2      -3.99
2 speed           3.93     0.416      9.46 1.49e-12     3.10      4.77
```

```
library(knitr)
# for nice tables in nicer formats
kable(tidy(cars_lm, conf.int = TRUE), digits = 4, format = "html")
```

term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	-17.5791	6.7584	-2.6011	0.0123	-31.1678	-3.9903
speed	3.9324	0.4155	9.4640	0.0000	3.0970	4.7679

Some more notation

$$\mu_{y|x} = \mathbf{E}[Y|x] = \beta_0 + \beta_1 x$$

We use $\hat{\mu}_{y|x}$ as our estimate of $\mu_{y|x} = E[Y|x]$

The predicted value is a function of the x value

$$\hat{y}(x) = \hat{\beta}_0 + \hat{\beta}_1 x$$

Two additional inference problems

Estimate the mean stopping distance for all cars in 1920s that are travelling at 22 mph

$$\mu_{y|x} = \mathrm{E}[Y \mid 22] = \beta_0 + \beta_1(22)$$

Predict the stopping distance for an individual car in 1920s that is travelling at 22 mph

$$Y = \beta_0 + \beta_1(22) + \epsilon$$

Estimating the average response vs predicting one response

Both the estimation and prediction problem have the same "point" estimate/prediction:

$$\hat{\beta}_0 + \hat{\beta}_1(22) = -17.5791 + 3.9324(22) = 68.9339$$

- But, the uncertainty in these two problems is different!

There is less variability when estimating a mean response then when predicting one individual response.

Estimating a mean response $\mu_{y|x=x_0}$

Parameter:

$$\mu_{y|x_0} = \beta_0 + \beta_1 x_0$$

Estimate:

$$\hat{\mu}_{y|x_0} = \hat{\beta}_0 + \hat{\beta}_1 x_0$$

SE of our estimate:

$$SE\left(\hat{\mu}_{y|x_0}\right) = \hat{\sigma}\sqrt{\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{(n-1)s_x^2}}$$

SE grows as x_0 gets further from the average predictor value $\bar{x}$

A 95% confidence interval for the mean response $\mu_{y|x_0}$:

$$\hat{\mu}_{y|x_0} \pm t^*_{df=n-2} SE\left(\hat{\mu}_{y|x_0}\right)$$

CI for a mean response $\mu_{y|x=x_0}$ in R

$$\text{predict(my\_lm, newdata, interval} = \text{“confidence”)}$$

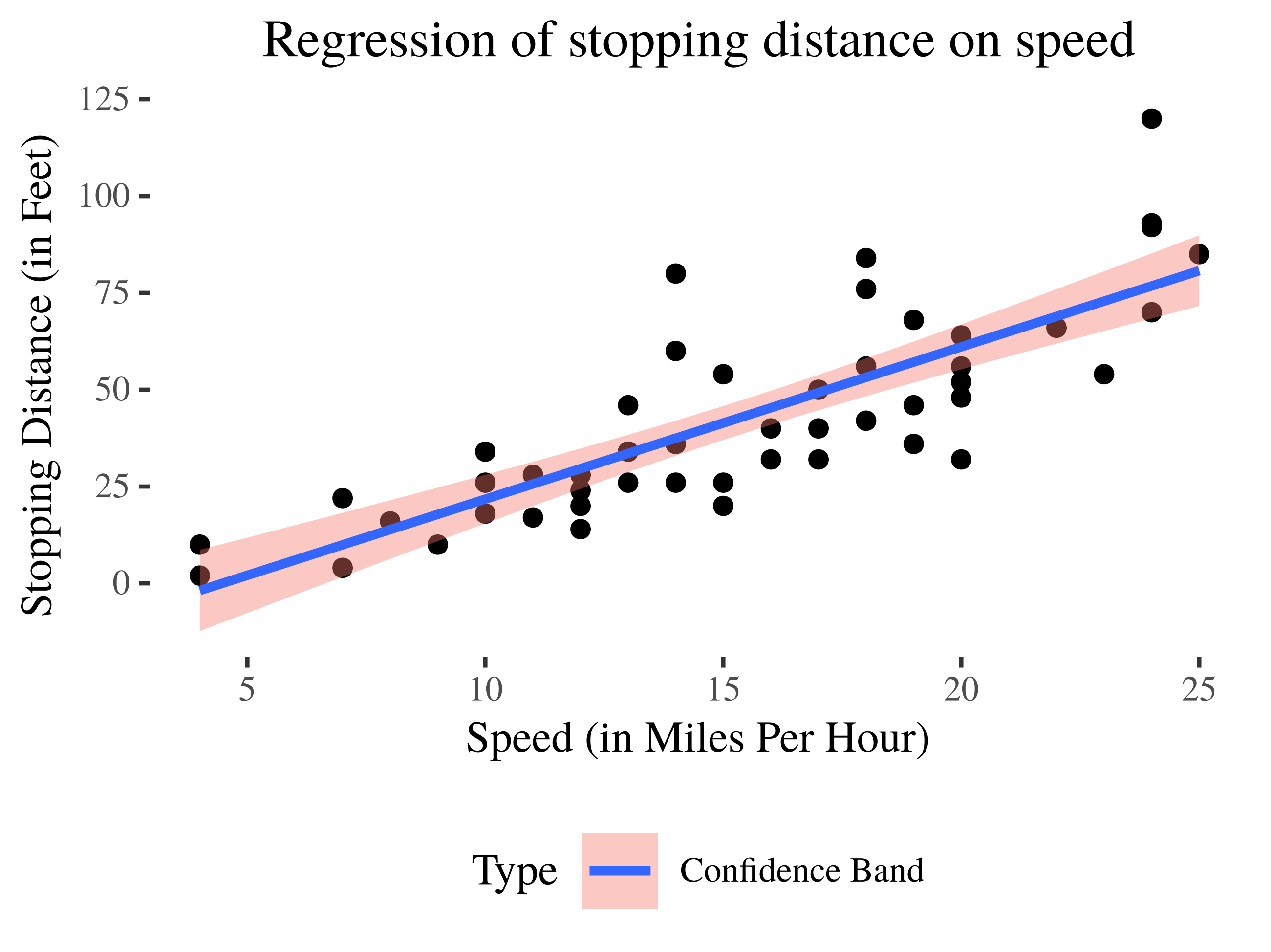
```
predict(cars_lm,  # model object
        newdata = data.frame(speed = 22), # new data
        interval = "confidence")    # interval type
```

```
      fit      lwr      upr
1 68.9339 61.8963 75.97149
```

I'm 95% confident that the mean stopping distance is between 61.8963 to 75.9715 ft when the speed is 22 mph.

Visualizing CI for a mean response

```
ggplot(cars, aes(x = speed, y = dist)) +
  geom_point() +
  theme(legend.position = "bottom") +
  labs(x='Speed (in Miles Per Hour)',
       y='Stopping Distance (in Feet)',
       title='Regression of stopping distance
       fill = "Type") +
  theme(plot.title = element_text(hjust = 0.5)
  geom_smooth(method = "lm", aes(fill = "Confi
```



Predicting unseen/future response Y given $x = x_0$

Unknown Response: $Y = \beta_0 + \beta_1 x_0 + \epsilon$

Prediction: $\text{pred}_{y|x_0} = \hat{y}(x_0) = \hat{\beta}_0 + \hat{\beta}_1 x_0$

SE of our prediction:

$$SE\left(\text{pred}_{\,y|x_0}\right) = \hat{\sigma}\sqrt{\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{(n-1)s_x^2} + 1} = \sqrt{SE\left(\hat{\mu}_{y|x_0}\right)^2 + \hat{\sigma}^2}$$

- Mathematically $SE(\text{ pred }_{y|x_0}) > SE\left(\hat{\mu}_{y|x_0}\right)$

A 95% prediction interval for an individual response at $x = x_0$

$$\text{pred}_{y|x_0} \pm t^*_{df=n-2} SE\left(\text{pred}_{y|x_0}\right)$$

Prediction interval in R

$$\text{predict}(\text{my\_lm}, \text{newdata}, \text{interval} = \text{“prediction”})$$

```
predict(cars_lm,  # model object
        newdata = data.frame(speed = 22), # new data
        interval = "prediction")    # interval type
```

```
       fit      lwr      upr
1 68.9339 37.22044 100.6474
```

I'm 95% confident that the stopping distance for a new car traveling with speed 22 mph is between 37.2204 to 100.6474 ft.

Prediction interval in R

```r
predict(cars_lm,  # model object
        newdata = data.frame(speed = 22), # new data
        interval = "prediction",    # interval type
        se.fit = TRUE)  # include SE for mean est.
```

```
$fit
       fit      lwr      upr
1 68.9339 37.22044 100.6474

$se.fit
[1] 3.500187

$df
[1] 48

$residual.scale
[1] 15.37959
```

```r
qt(.975, df = 50 - 2)
[1] 2.010635
```

$$\hat{y}(22) = 68.9339$$

$$\hat{\sigma} = 15.37959$$

$$SE\left(\hat{y}(22)\right) = \sqrt{3.50019^2 + 15.3796^2}$$

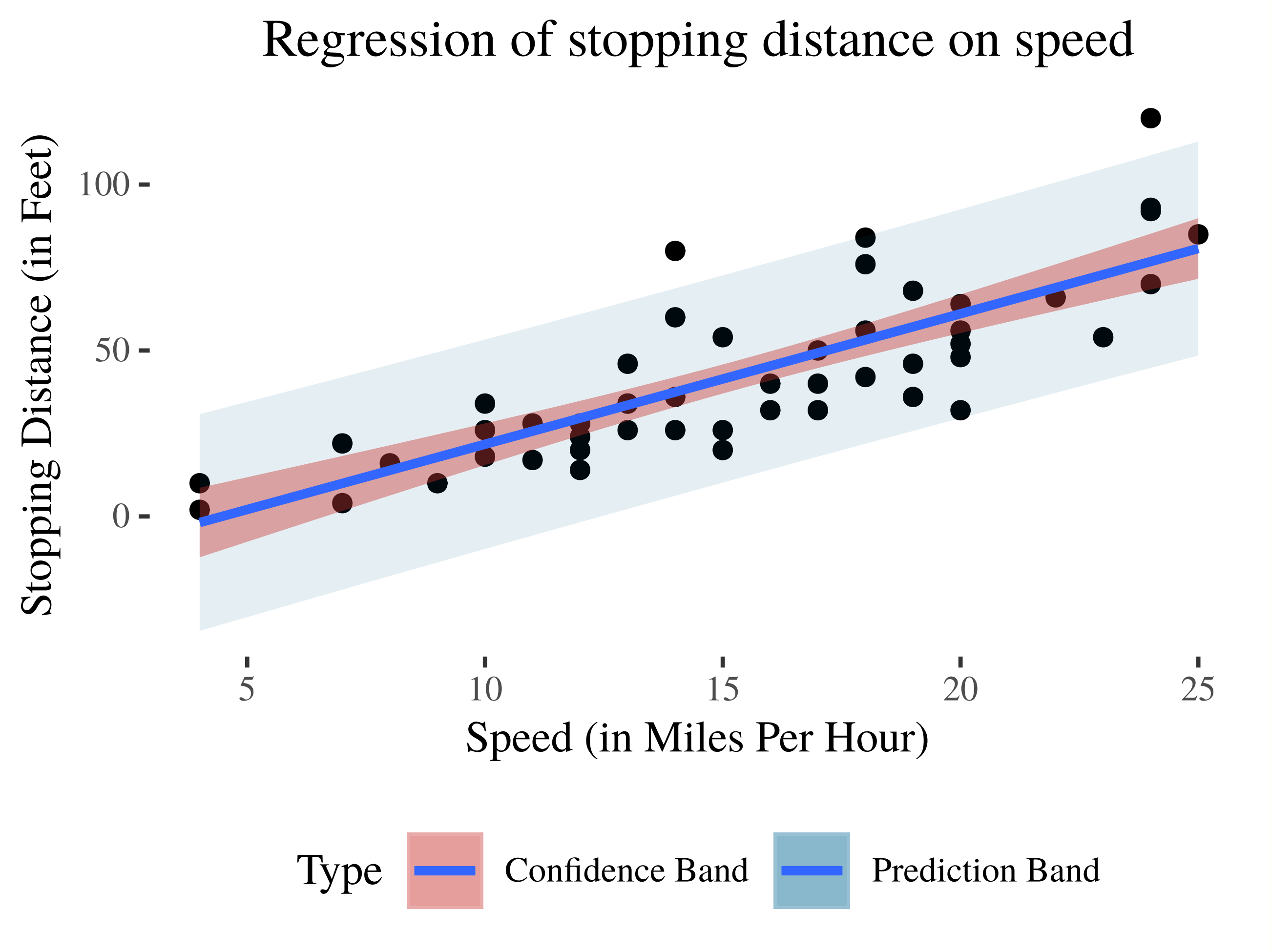
$$= 15.77286$$

$$68.9339 \pm (2.010635)(15.77286)$$

$$\implies (37.2204, 100.6474)$$

Visualizing both prediction interval and confidence interval

```
library(ggthemes)
ggplot(cars_pred, aes(x = speed, y = dist)) +
 geom_point() +  # plot data
 theme(legend.position = "bottom",
       plot.title = element_text(hjust = 0.5)) +
  labs(x='Speed (in Miles Per Hour)',
       y='Stopping Distance (in Feet)',
       title='Regression of stopping distance on speed',
       fill = "Type") +
 geom_ribbon(aes(ymin = lwr,  # lower prediction bound a
 ymax = upr,  # upper prediction bound at a given speed
 fill = "Prediction Band"),  # quick way to get a legend
 alpha = .1) +  # alpha closer to 0 makes ribbon more tr
 geom_smooth(method = "lm",  # add confidence bands too
 aes(fill = "Confidence Band"),  # another fill for con
 alpha = .4) +
scale_fill_wsj()
```



Group HW 2

End of semester student evaluations for 463 courses taught by a sample of 94 professors from the University of Texas at Austin.

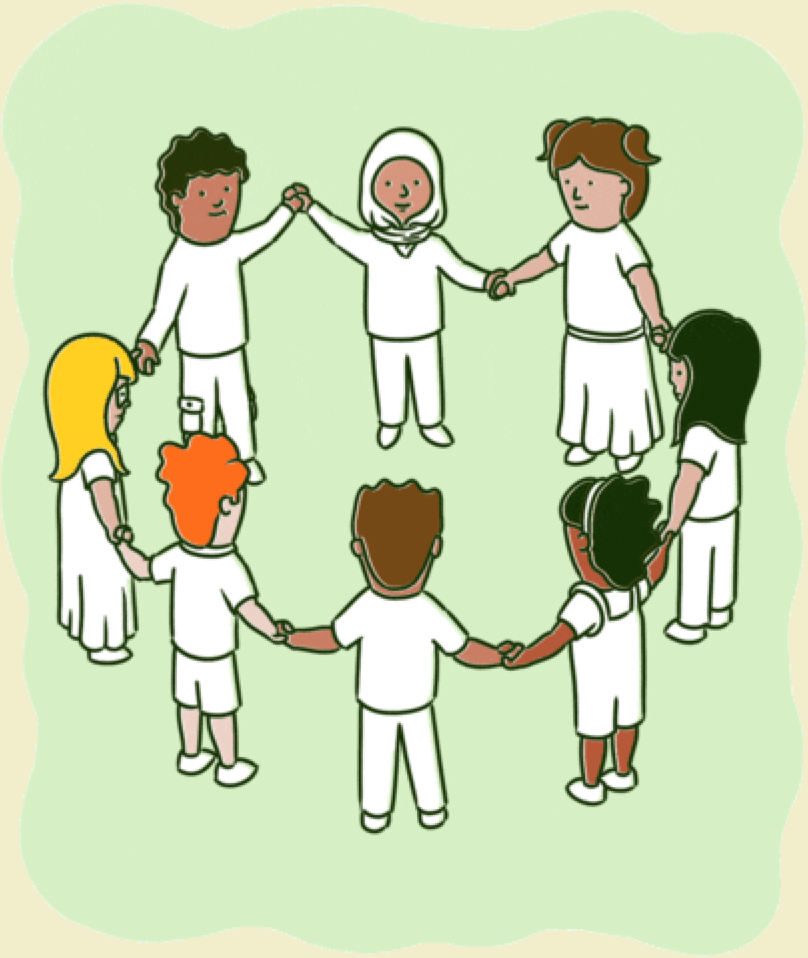
Is there a relationship between a teacher's physical appearance and their teaching evaluation?

- `score`: Average professor evaluation score, from (1) very unsatisfactory - (5) excellent
- `bty_avg`: Average beauty rating of professor, from (1) lowest - (10) highest

Beauty in the classroom: instructors’ pulchritude and putative pedagogical productivity, Economics of Education Review (2005)

Your Turn 1

10:00



- Get the in class activity file from moodle
- Use the dataset closing forces and the heights of the claws on `crabs`
- Try to repeat the inference steps in a group