Lab3 Linear Regression Solution

Note: In this solution, I deliberately not use y and x, but damage and distance. You don't always need to write y, x1, x2, ..., but can use the more meanful variable names.

1) Use R to estimate the unknown parameters of the hypothesised model. Find the least squares estimates of the slope $\hat{\beta}_1$ and intercept $\hat{\beta}_0$ on the printout. Write down the least squares equation for this model.

```
damage <- c(26.2, 17.8, 31.3, 23.1, 27.5, 36.0, 14.1, 22.3, 19.6, 31.3, 24.0, 17.3, 43.2, 36.4, 26.1)
distance <- c(3.4, 1.8, 4.6, 2.3, 3.1, 5.5, 0.7, 3.0, 2.6, 4.3, 2.1, 1.1, 6.1, 4.8, 3.8)
lm.fit <- lm(damage ~ distance)
```

From the output, $\hat{\beta}_1 = 4.919$ and $\hat{\beta}_0 = 10.278$. As a result, the least squares equation is $damage = 4.919 * distance + 10.278$.

2) Plot the dataset and the least squares line on the same figure.

```
plot(distance, damage,
     xlab ="distance", ylab = "damage",
     smain = "Plot of distance vs damage")
```

```
## Warning in plot.window(...): "smain" is not a graphical parameter
```

```
## Warning in plot.xy(xy, type, ...): "smain" is not a graphical parameter
```

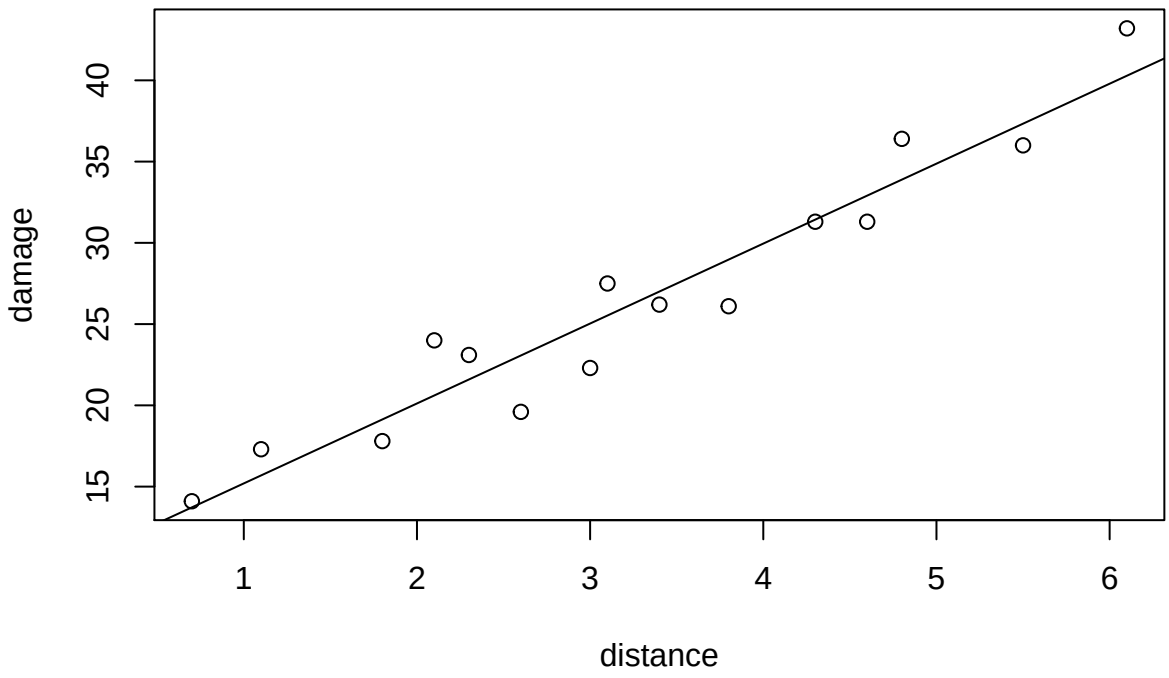
```
## Warning in axis(side = side, at = at, labels = labels, ...): "smain" is not
## a graphical parameter
```

```
## Warning in axis(side = side, at = at, labels = labels, ...): "smain" is not
## a graphical parameter
```

```
## Warning in box(...): "smain" is not a graphical parameter
```

```
## Warning in title(...): "smain" is not a graphical parameter
```

```
abline(lm.fit)
```



3) How to interpret the slope $\hat{\beta}_1$ and y-intercept $\hat{\beta}_0$ of the least squares line?

$\hat{\beta}_1 = 4.919$ implies that the estimated mean damage increases by \$4,919 for each additional mile from the fire station. Note that the interpretation is valid over the range of x, or from 0.7 to 6.1 miles from the station.

$\hat{\beta}_0 = 10.278$ implies that a fire 0 miles from the fire station has an estimated mean damage of \$10,278.

4) Measuring the extent to which the model fits the data.

a. What is the RSE of this model? How to interpret it?

```
summary(lm.fit)
```

```
##
## Call:
## lm(formula = damage ~ distance)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4682 -1.4705 -0.1311  1.7915  3.3915
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  10.2779     1.4203   7.237 6.59e-06 ***
## distance      4.9193     0.3927  12.525 1.25e-08 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.316 on 13 degrees of freedom
## Multiple R-squared:  0.9235, Adjusted R-squared:  0.9176
## F-statistic: 156.9 on 1 and 13 DF,  p-value: 1.248e-08
```

RSE is 2.316.

This implies that any prediction on the observed fire damage (y) values based on the distance would be off by about 2.316 thousand dollars on average when using the least squares line.

b. What is the R-squared value? How to interpret it?

Adjuested R-squared is 0.9176 (after correction).

which implies that about 91.76% of the sample variation in fire damage (y) is explained by the distance (x) between the fire and the fire station.

5) Test the null hypothesis that $\hat{\beta}_1$ is 0. Find the t-value and p-value. What conclusions can you draw from the two values?

t-value = 12.525, p-value = 1.248e-08

t-value is large and p-value is small. We can conclude that $\hat{\beta}_1 \neq 0$ and there is a relationship between fire damage and the distance from the nearest fire station.

6) Predict the value of damage for a new set of distances (distance = 0.5, 1.5, 2.5, 3.5). Can you obtain the above new damages simultaneously?

```
predict(lm.fit,
        newdata = data.frame(distance = 0.5:3.5) )
```

```
##        1        2        3        4
## 12.73759 17.65692 22.57626 27.49559
```

7) Find the 95% confidence interval for parameters $\hat{\beta}_0$ and $\hat{\beta}_1$. How to interpret the intervals?

```
confint(lm.fit, level = 0.95)
```

```
##                2.5 %    97.5 %
## (Intercept) 7.209605 13.346252
## distance    4.070851  5.767811
```

We estimate (with 95% confidence) that the interval from $4,070 to $5,768 encloses the mean increase ($\hat{\beta}_1$) in fire damage per additional mile distance from the fire station.

We estimate (with 95% confidence) that the interval from $7209.61 to $13346.25 encloses the mean damage ($\hat{\beta}_0$) that caused when distance is 0 (the fire station is on fire!).

8) Find the confidence interval for the prediction of damage for a new set of distance (distance = 0.5, 3, 5.5). Can you obtain the above confidence intervals simultaneously?

```
predict(lm.fit,
        data.frame(distance=(c(0.5,3,5.5))),
        interval="confidence")
```

```
##        fit      lwr      upr
## 1 12.73759 10.04812 15.42707
## 2 25.03592 23.72219 26.34965
## 3 37.33425 35.05007 39.61843
```

9) Find the prediction interval for the prediction of damage (y) for a new set of distance (distance = 0.5, 3, 5.5). Can you obtain the above confidence intervals simultaneously?

```
predict(lm.fit,
        data.frame(distance=(c(0.5,3,5.5))),
        interval="prediction")
```

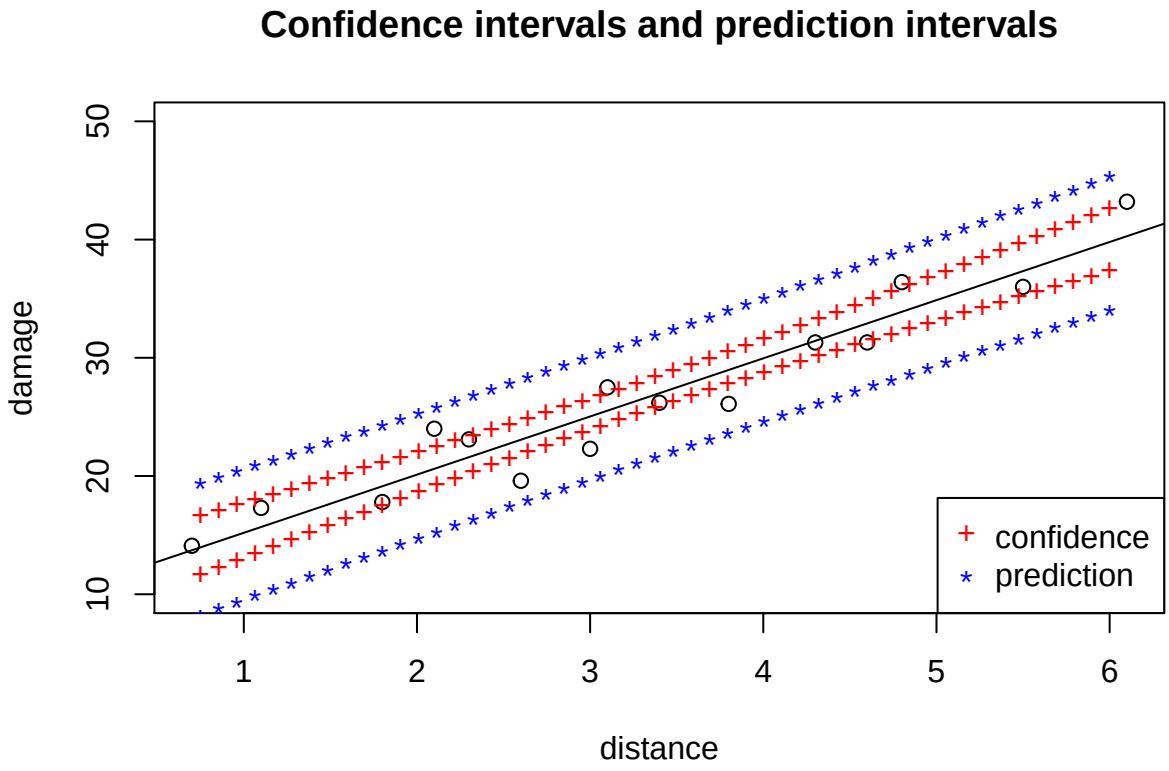
```
##        fit       lwr      upr
## 1 12.73759  7.056494 18.41869
## 2 25.03592 19.862187 30.20965
## 3 37.33425 31.833418 42.83508
```

10) Compare your results in 8) and 9) and comment on the confidence intervals and prediction intervals.

Prediction intervals are always wider than confidence intervals.

11) Plot the confidence intervals and prediction intervals in the same figure.

```
plot(distance, damage,
     xlab="distance", ylab = "damage",
     main = "Confidence intervals and prediction intervals",
     ylim = c(10,50)
     )
abline(lm.fit)
newDist  <- data.frame(distance=seq(0.75,6.0,length=51))
p_conf <- predict(lm.fit,newDist,interval="confidence")
p_pred <- predict(lm.fit,newDist,interval="prediction")
lines(newDist$distance,p_conf[,"lwr"],col="red", type="b",pch="+")
lines(newDist$distance,p_conf[,"upr"],col="red", type="b",pch="+")
lines(newDist$distance,p_pred[,"upr"],col="blue", type="b",pch="*")
lines(newDist$distance,p_pred[,"lwr"],col="blue",type="b",pch="*")
legend("bottomright",
       pch=c("+","*"),
       col=c("red","blue"),
       legend = c("confidence","prediction"))
```


Confidence intervals and prediction intervals
damage
distance
10
20
30
40
50
1
2
3
4
5
6
confidence
prediction