Bruce Campbell ST-617 HW 1

Wed Jun 29 00:06:24 2016

Chapter 3

Problem 13

a)

```
rm(list = ls())

X <- rnorm(100, 0, 1)

epsilon <- rnorm(100, 0, 0.25)

Y <- -1 + 0.5 * X + epsilon
```

c)

The length of y is 100, $\beta_0 = -1$ and $\beta_1 = 0.5$

d)

We note that there is a positive linear relation between X and Y, that the midpoint of theh range of X is rougly 0 and the midpoint of the range of Y is rougly -1. We also note the dispersion of the data along the diagonal is about $\frac{1}{4}$

e)

```
DF <- data.frame(predictor = X, response = Y)
lm.fit <- lm(response ~ predictor, data = DF)
summary(lm.fit)
```

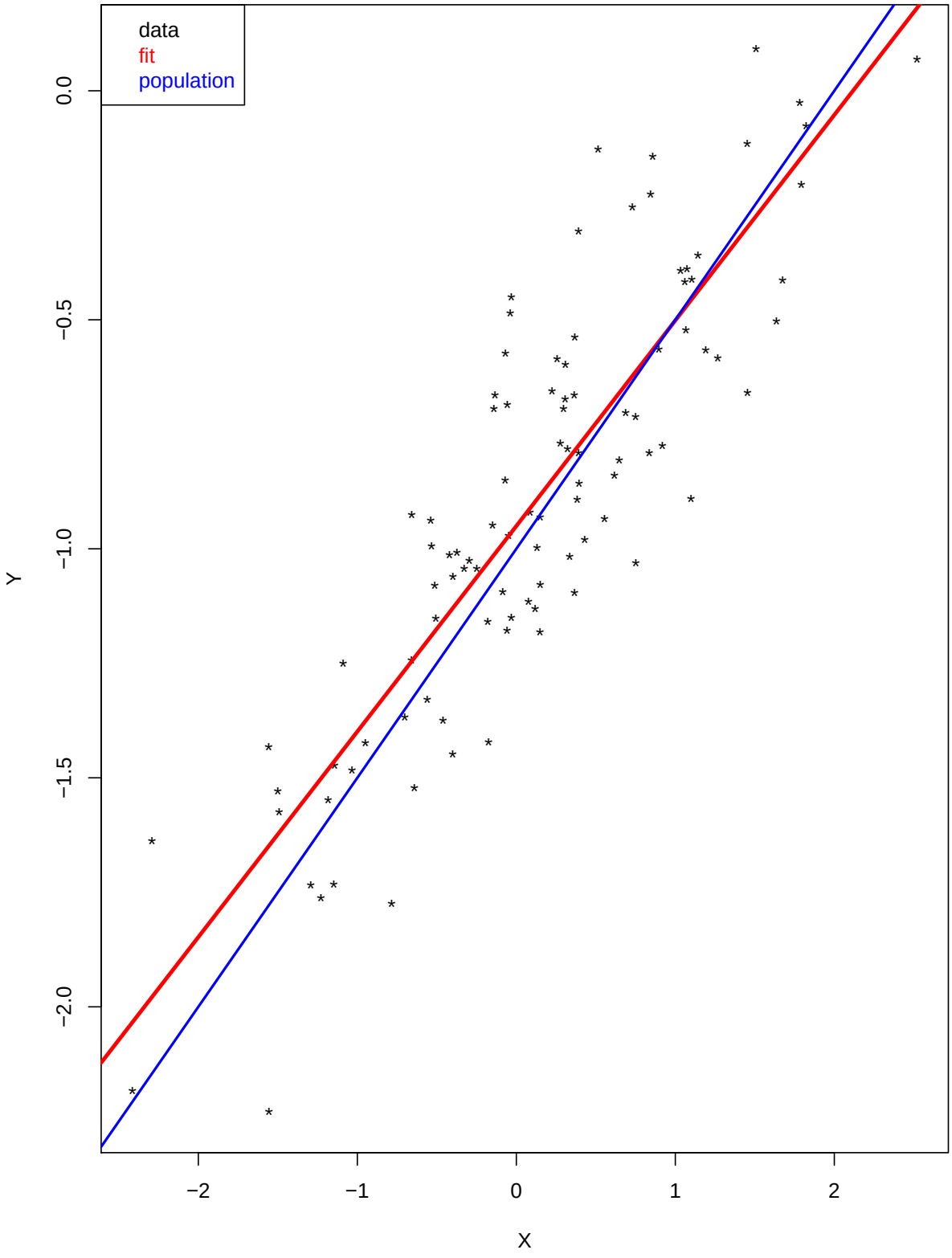
```
##
## Call:
## lm(formula = response ~ predictor, data = DF)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.57715 -0.19317 -0.00692  0.13008  0.59469
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.95023    0.02406  -39.49   <2e-16 ***
```

```
## predictor    0.44876    0.02619   17.13   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2388 on 98 degrees of freedom
## Multiple R-squared:  0.7497, Adjusted R-squared:  0.7472
## F-statistic: 293.6 on 1 and 98 DF,  p-value: < 2.2e-16
```

The estimated values of the regression coefficients are very close to the actual values of -1 and .5.

f)

```
plot(X, Y, pch = "*")
abline(lm.fit, col = "red", lwd = 3)
abline(-1, 0.5, col = "blue", lwd = 2)
legend("topleft", title.col = "black", c("data", "fit", "population"), text.col = c("black",
    "red", "blue"), text.font = 1, cex = 1)
```


data
fit
population
0.0
−0.5
−1.0
−1.5
−2.0
Y
−2
−1
0
1
2
X

g)

```
lm_poly.fit <- lm(response ~ predictor + I(predictor^2), data = DF)
summary(lm_poly.fit)
```

```
##
## Call:
## lm(formula = response ~ predictor + I(predictor^2), data = DF)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.54699 -0.18516 -0.01235  0.12549  0.58390
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.93546    0.02917 -32.066   <2e-16 ***
## predictor       0.45012    0.02626  17.140   <2e-16 ***
## I(predictor^2) -0.01768    0.01970  -0.897    0.372
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.239 on 97 degrees of freedom
## Multiple R-squared:  0.7518, Adjusted R-squared:  0.7467
## F-statistic: 146.9 on 2 and 97 DF,  p-value: < 2.2e-16
```

There is very little evidence that adding a ploynomial term to the regression has improves the fit. RSE and R squared were not markedly affected by the addition of the quadratic term. We also note the pvalue of the quadratic term is high and the coefficient is near 0, reflecting it's insignificance in the model.

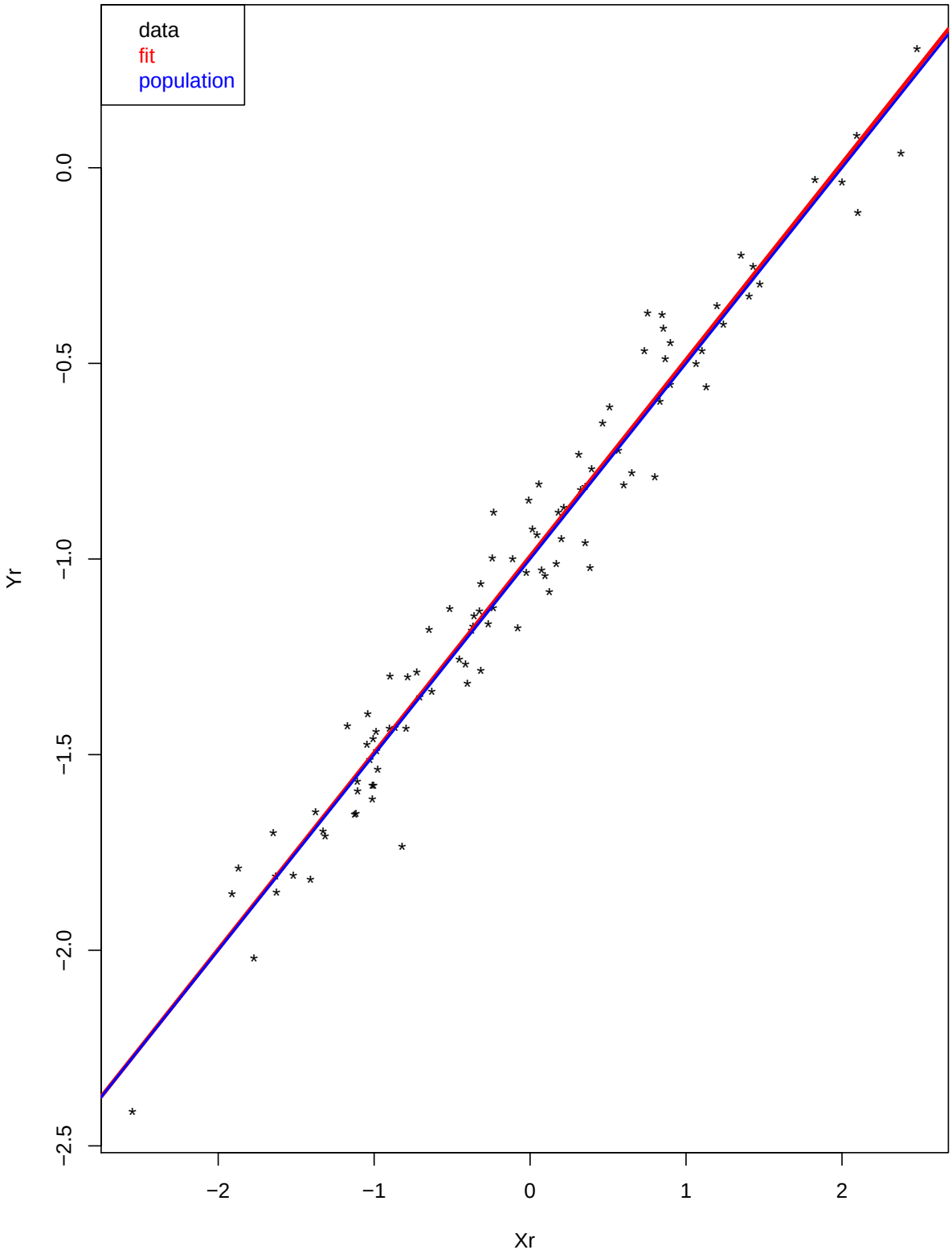
h) Reducing error

```
Xr <- rnorm(100, 0, 1)
epsilon <- rnorm(100, 0, 0.1)
Yr <- -1 + 0.5 * Xr + epsilon
DF <- data.frame(predictor = Xr, response = Yr)
lm_r.fit <- lm(response ~ predictor, data = DF)
summary(lm_r.fit)
```

```
##
## Call:
## lm(formula = response ~ predictor, data = DF)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.32746 -0.05538 -0.00114  0.06576  0.24613
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.992857   0.010434  -95.16   <2e-16 ***
## predictor    0.502269   0.009936   50.55   <2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1038 on 98 degrees of freedom
## Multiple R-squared:  0.9631, Adjusted R-squared:  0.9627
## F-statistic:  2555 on 1 and 98 DF,  p-value: < 2.2e-16
```

```
plot(Xr, Yr, pch = "*")
abline(lm_r.fit, col = "red", lwd = 3)
abline(-1, 0.5, col = "blue", lwd = 2)
legend("topleft", title.col = "black", c("data", "fit", "population"), text.col = c("black",
    "red", "blue"), text.font = 1, cex = 1)
```


data
fit
population
Yr
Xr
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
−2
−1
0
1
2

When we reduce the error in the data the median residual and RES are decreased. Mutliple R square is increased. All indicates of inproved performance in the fit.

i)

Confidence interval for the first model

```
confint(lm.fit)
```

```
##                  2.5 %     97.5 %
## (Intercept) -0.9979860 -0.9024779
## predictor    0.3967797  0.5007328
```

Confidence interval for the second model

```
confint(lm_r.fit)
```

```
##                 2.5 %     97.5 %
## (Intercept) -1.013563 -0.9721515
## predictor    0.482551  0.5219880
```

As expected our confidence interval in the second model is smaller than the first.