Confidence-Interval.R

Shraddha Somani

```
# Confidence Interval
# Q1) For the prostate data infaraway, fit a model with lpsa as the response
and the other variables as predictors. Compute 90% and 95% CIs for the
parameter associated with age
require(faraway)

## Loading required package: faraway

head(prostate)

##       lcavol lweight age      lbph svi      lcp gleason pgg45     lpsa
## 1 -0.5798185  2.7695  50 -1.386294   0 -1.38629       6     0 -0.43078
## 2 -0.9942523  3.3196  58 -1.386294   0 -1.38629       6     0 -0.16252
## 3 -0.5108256  2.6912  74 -1.386294   0 -1.38629       7    20 -0.16252
## 4 -1.2039728  3.2828  58 -1.386294   0 -1.38629       6     0 -0.16252
## 5  0.7514161  3.4324  62 -1.386294   0 -1.38629       6     0  0.37156
## 6 -1.0498221  3.2288  50 -1.386294   0 -1.38629       6     0  0.76547

g = lm(lpsa ~ ., data = prostate)
confint(g, "age", level = 0.90)

##             5 %         95 %
## age -0.0382102 -0.001064151

confint(g, "age", level = 0.95)

##           2.5 %      97.5 %
## age -0.04184062 0.002566267

# Q2) Compute and display a 95% joint confidence region for the parameters
associated with age and lbph. Plot the origin and report the outcome of the
appropriate hypotheses test. Affirm this conclusion with an appropriate
partial F-test.
require(ellipse)

## Loading required package: ellipse

plot(ellipse(g, c("age", "lbph")), level = 0.95, type = "l", main = "Joint
Confidence Region")

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

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

## Warning in axis(side = side, at = at, labels = labels, ...): "level" is
not
```

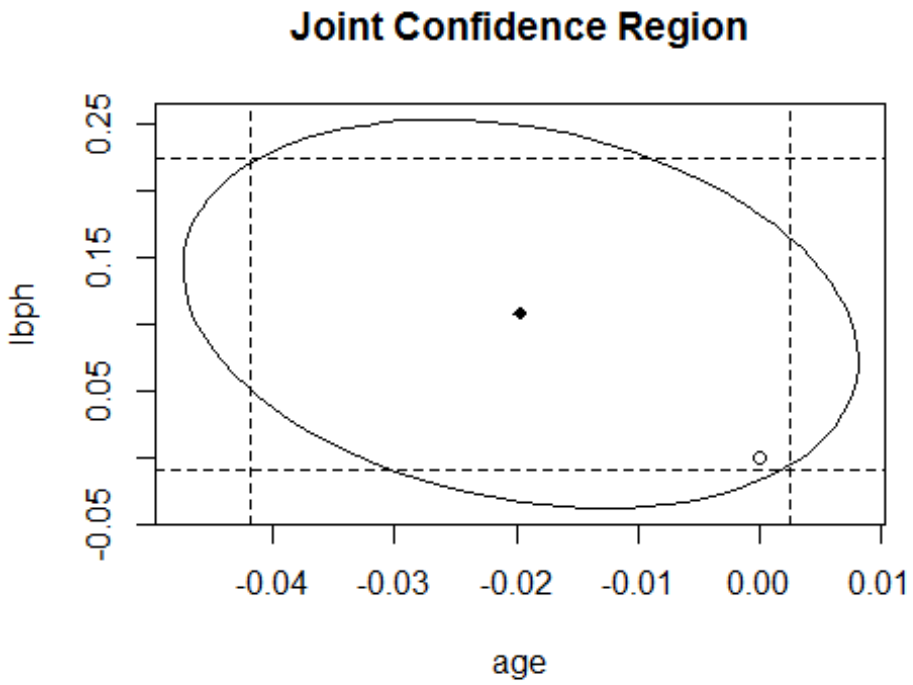
```
## a graphical parameter

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

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

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

points(0,0)
points(coef(g)["age"], coef(g)["lbph"], pch = 18)
abline(v = confint(g)["age",], lty = 2)
abline(h = confint(g)["lbph",], lty = 2)
```




```
# Q3) Predict lpsa (95%) for a new patient with lcavol = 1.22692, lweight =
3.62301, age = 65, lbph = -0.3001, svi = 0.0, lcp = -0.79851, gleason = 7.0,
pgg45 = 15.0. Do this again for the mean response. Using the exp() function,
obtain the new prediction and mean response for psa.
x0 = data.frame(lcavol = 1.22692, lweight = 3.62301, age = 65, lbph = -
0.3001, svi = 0.0, lcp = -0.79851, gleason = 7.0, pgg45 = 15.0,
stringsAsFactors = FALSE)
predict(g, x0, level = 0.95, interval = "prediction")

##        fit       lwr     upr
## 1 2.195654 0.7708983 3.62041

exp(predict(g, x0, level = 0.95, interval = "prediction"))
```

```
##        fit      lwr      upr
## 1 8.985877 2.161707 37.35288

# Q4) Repeat the above exercise with new patient age = 20
x0 = data.frame(lcavol = 1.22692, lweight = 3.62301, age = 20, lbph = -
0.3001, svi = 0.0, lcp = -0.79851, gleason = 7.0, pgg45 = 15.0,
stringsAsFactors = FALSE)
predict(g, x0, level = 0.95, interval = "prediction")

##        fit      lwr      upr
## 1 3.079327 1.357826 4.800828

exp(predict(g, x0, level = 0.95, interval = "prediction"))

##        fit      lwr      upr
## 1 21.74376 3.887732 121.6111

#Analysis:
#The origin is inside the ellipse, so we do not reject the null hypothesis.


# Q5) For the model in exercise 1, remove all the predictors that are not
significant at the 5% level. Recompute the predictions for exercises 3 and 4.
Compare CIs. On the psa scale, which CIs do you prefer?
g.sm = lm(lpsa ~ lcavol + lweight + age + svi , data = prostate)
# For age = 65
x1 = data.frame(lcavol = 1.22692, lweight = 3.62301, age = 65, svi = 0.0,
stringsAsFactors = FALSE)
predict(g.sm, x1, level = 0.95, interval = "prediction")

##        fit       lwr      upr
## 1 2.238483 0.8031959 3.673771

exp(predict(g.sm, x1, level = 0.95, interval = "prediction"))

##        fit      lwr      upr
## 1 9.379095 2.232665 39.40019

# For age = 20
x2 = data.frame(lcavol = 1.22692, lweight = 3.62301, age = 20, svi = 0.0)
predict(g.sm, x2, level = 0.95, interval = "prediction")

##        fit       lwr      upr
## 1 2.656402 0.9569783 4.355825

exp(predict(g.sm, x2, level = 0.95, interval = "prediction"))

##        fit      lwr      upr
## 1 14.24494 2.603817 77.93108

# Analysis:
# - The prediction intervals from the second model should be narrower than
those from the original model theoretically because all significant values
```

```
have been removed
# - Therefore,the second model one should explain the response more
accurately than the first model.
# - The length of the second prediction interval is longer than the first
because of age.
# - The narrower prediction intervals are preferred.


# Test the "small" model in exercise 5 against the "big"" model in exercise 1
at probability type I error ??=0.05??=0.05. Which model is preferred?
# g = lm(lpsa ~ ., data = prostate)
# g.sm = lm(lpsa ~ lcavol + lweight + age + svi , data = prostate)
anova(g, g.sm)
```

```
## Analysis of Variance Table
##
## Model 1: lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason +
##     pgg45
## Model 2: lpsa ~ lcavol + lweight + age + svi
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     88 44.163
## 2     92 47.382 -4   -3.2193 1.6037 0.1805
```

```
# Analysis:
# - Since the P-Value for the F-stat is 0.1805 and is larger than the
significance level 0.05, we accept the small model.
# - Small model is preferred because it is simple.
```