Assignment2

Lorenzo Ricciardulli

2024-01-19

```
## Warning: il pacchetto 'rstanarm' è stato creato con R versione 4.1.3
```

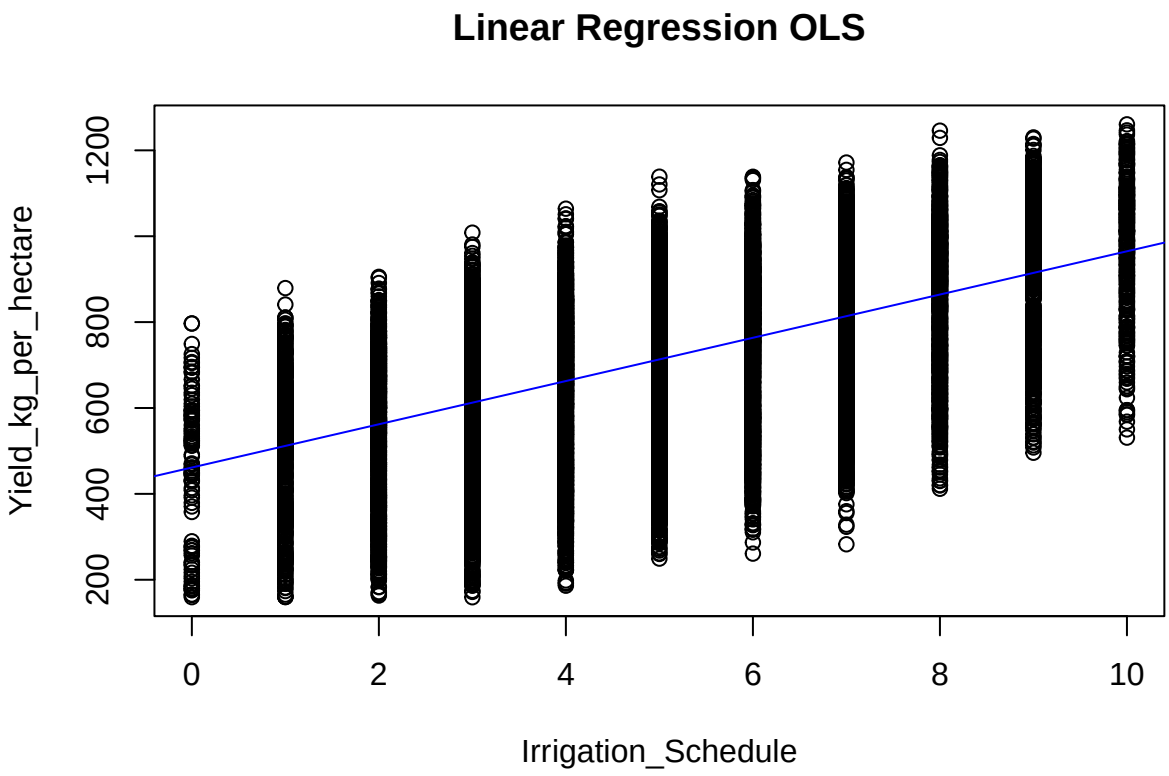
##(A) Given the dataset collected in the previous assignment, fit a linear regression model with a single predictor using the method of least squares.

```
dati_senza_outlier = read.csv("dati_senza_outlier.csv")
formula = dati_senza_outlier$Yield_kg_per_hectare ~ dati_senza_outlier$Irrigation_Schedule
model = lm(formula, data = dati_senza_outlier)
summary(model)
```

```
##
## Call:
## lm(formula = formula, data = dati_senza_outlier)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -531.28 -124.21   34.43  125.82  425.50
##
## Coefficients:
##                                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          461.365      3.413   135.2   <2e-16 ***
## dati_senza_outlier$Irrigation_Schedule   50.356      0.635    79.3   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 166 on 15507 degrees of freedom
## Multiple R-squared:  0.2885, Adjusted R-squared:  0.2885
## F-statistic:  6288 on 1 and 15507 DF,  p-value: < 2.2e-16
```

##(B)Graph the data along with the fitted line

```
plot(dati_senza_outlier$Irrigation_Schedule, dati_senza_outlier$Yield_kg_per_hectare,
main = "Linear Regression OLS",
xlab = "Irrigation_Schedule", ylab = "Yield_kg_per_hectare")
intercetta = coef(model)[1]
pendenza = coef(model)[2]
abline(model, col = "blue")
```




```
print(model)
```

```
##
## Call:
## lm(formula = formula, data = dati_senza_outlier)
##
## Coefficients:
##                               (Intercept)  dati_senza_outlier$Irrigation_Schedule
##                                    461.37                                   50.36
```

##(C)Interpret the estimated parameters and their uncertainties

Estimate: 461.365 Interpretation: The estimated value of the dependent variable when all independent variables are zero. In this context, when Irrigation_Schedule is zero (or absent), the estimated value of Y is 461.365.

Estimate: 50.356 Interpretation: Represents the estimated change in the dependent variable (Y) for a one-unit increase in the Irrigation_Schedule variable (likely a dummy-coded variable). In this case, a one-unit increase in Irrigation_Schedule is associated with an estimated increase of 50.356 units in the dependent variable.

Significance Test (t value and p-value): The t value is a ratio of the coefficient estimate to its standard error. In this case, the high t value (79.3) indicates that the coefficient for Irrigation_Schedule is statistically significant. The p-value associated with the coefficient for Irrigation_Schedule is very close to zero (<2e-16), indicating statistical significance. Overall Results:

Residual standard error: 166 Multiple R-squared: 0.2885 Adjusted R-squared: 0.2885 F-statistic: 6288 with 1 and 15507 degrees of freedom p-value F-statistic: $<$ 2.2e-16 These overall results provide an overview of

the overall goodness of fit of the model. The low residual standard error indicates that the model explains (partially) variability in the data.

##(D)Provide and discuss the coefficient of determination, R2

Multiple R-squared: 0.2885 This value (approximately 29%) indicates that about 29% of the variance in the dependent variable is explained by the independent variable(s) included in the model. The remaining 71% of the variance is not explained by the model. Adjusted R-squared: 0.2885 Adjusted R-squared considers the number of predictors in the model and is useful for comparing models with different numbers of predictors. In this case, the adjusted R-squared is the same as the multiple R-squared, suggesting there is only one predictor. Discussion: An R-squared of 0.2885 indicates that the model has limited explanatory power; only about 29% of the variation in the dependent variable is accounted for by the independent variable(s). The high statistical significance (low p-value) of the F-statistic suggests that at least one of the predictors is related to the response variable. However, individual coefficients should be considered to understand their specific contributions. The coefficients of the model indicate that both the intercept and Irrigation_Schedule are statistically significant. It's important to note that while the model is statistically significant, the R-squared suggests that there is substantial unexplained variability, and additional factors might influence the dependent variable.

##(E)Assume to observe a new value of your predictor. What is the predicted response using your regression model?

Let's assume that a new value of irrigation schedule is observed, for example, 7. Using the coefficients of the regression:

```
newdata = data.frame(Irrigation_schedule=7)
prediction = predict(model, newdata = newdata, interval="prediction",level=.95)
```

```
## Warning: 'newdata' ha 1 riga ma la variabile trovata ha 15509 righe
```

```
print(prediction[15509,])
```

```
##       fit       lwr       upr
##  763.5028  438.1767 1088.8289
```

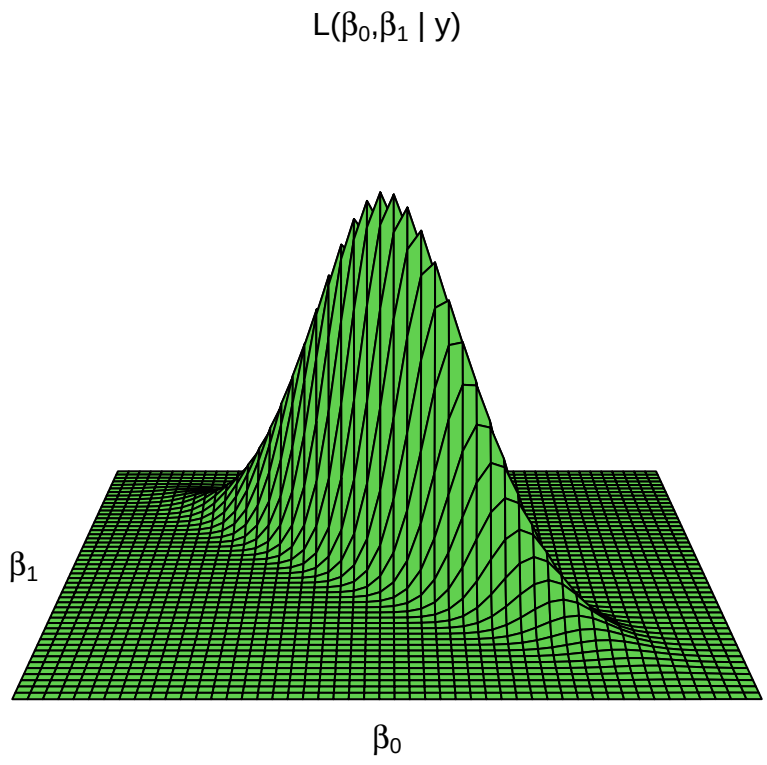
##(F)Plot the likelihood function of your simple linear regression model

```
summ = summary(model)
summ$coefficients
```

```
##                                       Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                           461.36534  3.4128681 135.18405        0
## dati_senza_outlier$Irrigation_Schedule  50.35625  0.6350406  79.29611        0
```

```
beta0 = summ$coefficients[1,1]
beta1 = summ$coefficients[2,1]
sigma = summ$sigma
Mx = mean(dati_senza_outlier$Irrigation_Schedule)
SSx = sum((dati_senza_outlier$Irrigation_Schedule-Mx)^2)
SE_beta0 = sigma * sqrt(1/nrow(dati_senza_outlier) + Mx^2/SSx)
SE_beta1 = sigma/sqrt(SSx)
Cov_betas = -sigma^2 * (Mx/SSx)
rng.beta0 = beta0 + SE_beta0*c(-4,4)
```

```r
rng.beta1 = beta1 + SE_beta1*c(-4,4)
beta0.grid = seq(rng.beta0[1], rng.beta0[2], length=50)
beta1.grid = seq(rng.beta1[1], rng.beta1[2], length=50)
loglike = array(NA , c(length(beta0.grid),length(beta1.grid)))
for (i in 1:length(beta0.grid)){
  for (j in 1:length(beta1.grid)){
    loglike[i,j] <- sum(dnorm(dati_senza_outlier$Yield_kg_per_hectare,
        mean = beta0.grid[i]+beta1.grid[j]*
        dati_senza_outlier$Irrigation_Schedule,
        sd = sigma, log=TRUE))
  }
}
max_loglike = max(loglike)
loglike = loglike - max_loglike
like = exp(loglike)
trans3d <- function(x,y,z, pmat) {
tr = cbind(x,y,z,1) %*% pmat
list(x = tr[,1]/tr[,4], y= tr[,2]/tr[,4])}
par(mar=c(0,0,0,0), mgp=c(2,0.5,0), tck=-0.01, pty='s')
res = persp(beta0.grid, beta1.grid, like,
xlab=expression(beta[0]),ylab='', zlab="likelihood", d=2, box = FALSE,
axes=TRUE, expand=.6, col = 3)
text(trans3d(mean(rng.beta0), rng.beta1[1]-.12*(rng.beta1[2]-rng.beta1[1]), 0, pm = res),
expression(beta[0]))
text(trans3d(rng.beta0[1]-.08*(rng.beta0[2]-rng.beta0[1]), mean(rng.beta1), 0, pm = res),
expression(beta[1]))
mtext(expression(paste("L(",beta[0],",", beta[1]," | y)")), side=3, line=-1.5)
```



##(G)Fit the linear regression model using the Bayesian approach assuming weakly informative priors. Discuss the differences with the analysis of point (a).

```
Bayes <- stan_glm(dati_senza_outlier$Yield_kg_per_hectare ~ dati_senza_outlier$Irrigation_Schedule,
data = dati_senza_outlier, refresh = 0)
print(Bayes)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      dati_senza_outlier$Yield_kg_per_hectare ~ dati_senza_outlier$Irrigation_Schedule
##  observations: 15509
##  predictors:   2
## ------
##                                       Median MAD_SD
## (Intercept)                           461.3    3.6
## dati_senza_outlier$Irrigation_Schedule  50.4    0.7
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 166.0    1.0
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

Intercept ((Intercept)) and Irrigation_Schedule: Median: The median of the intercept ((Intercept)) is 461.4, suggesting the estimated central value for yield when Irrigation_Schedule is zero. The median of Irriga-

tion_Schedule is 50.3, indicating the estimate of the average variation in yield associated with a one-unit increase in Irrigation_Schedule.

Auxiliary Parameter (sigma): Median: The median of sigma is 166.0. This represents an estimate of the standard deviation of errors in the data. In simple terms, it indicates how much the observed data can vary around the model prediction. In this case, 166.0 indicates an estimate of the dispersion of yield relative to the regression line.

MAD_SD (Median Absolute Deviation from Median): MAD_SD represents the median absolute deviation from the median, which is a robust indicator of dispersion. For example, a MAD_SD of 3.5 for the intercept indicates that the distribution of samples is on average 3.5 units away from the median of the intercept.

Comparison with the Classical Linear Model (lm):

In the Bayesian Approach, we obtain estimates such as medians and median absolute deviations (MAD_SD). This is typical of the Bayesian approach, which provides an estimate of the central tendency and dispersion in the posterior distribution of parameters.The Bayesian model also provides an estimate of the standard deviation of errors (sigma), which is useful for understanding how much the observed data can vary around the model prediction. Compared to the classical linear model, the Bayesian approach provides a more comprehensive description of the parameter distribution, allowing for the incorporation of uncertainty in the estimates.

Discussion of differences between traditional linear regression and the Bayesian approach using stan_glm.

Handling Uncertainty:

Linear Regression: In traditional linear regression, point estimates and confidence intervals are provided for the coefficients. However, these intervals are based on frequentist statistics and assume that the sample data is a random draw from a well-defined population. Confidence intervals give a range of values where we can reasonably expect the true parameter to lie, but they don't directly capture the uncertainty associated with the parameter estimates.

stan_glm takes a Bayesian approach, providing a posterior distribution for each parameter. Instead of a single point estimate, it gives a range of plausible values for each coefficient.The Bayesian framework naturally incorporates uncertainty by representing parameter estimates as probability distributions. This allows for a more nuanced understanding of the uncertainty surrounding the model parameters.

Model Flexibility:

Linear Regression: Traditional linear regression assumes that the errors are normally distributed and homoscedastic. It does not explicitly model the uncertainty in the coefficients.If the assumptions are violated, linear regression results may be biased or inefficient.

The Bayesian approach is more flexible, as it allows for the modeling of different distributions for the errors. It is less sensitive to violations of distributional assumptions.By providing complete posterior distributions, it acknowledges and quantifies uncertainties in a more robust manner, especially in cases where data distribution assumptions are unclear or violated.

Interpretability:

Linear Regression: Point estimates and confidence intervals are relatively straightforward to interpret. They provide a concise summary of the most likely values for the coefficients and their precision.

The posterior distributions provided by the Bayesian model can be more challenging to interpret for those unfamiliar with Bayesian statistics. However, they offer a richer representation of the uncertainty around parameter estimates.

Prior Information:

Linear Regression: Traditional regression typically assumes non-informative priors, treating all parameter values as equally likely before observing the data.

Bayesian models allow the incorporation of prior information into the analysis. This is particularly useful when there is existing knowledge about the parameters, helping to refine estimates.

##2(a) Simulate n data points from the linear model using the predictor of point 1(a) and assuming the estimated parameters as the true parameters

```
n = 15509
predictor = round(rnorm(n, mean = 5, sd = 2))
predictor = pmax(pmin(predictor, 10), 0)
response = intercetta +
pendenza * dati_senza_outlier$Irrigation_Schedule +
rnorm(n, mean = 0, sd = 166)
simulated_data = data.frame(Irrigation_Schedule = predictor, Yield_kg_per_hectare = response)
head(simulated_data)
```

```
##   Irrigation_Schedule Yield_kg_per_hectare
## 1                   8             689.7866
## 2                   9             819.5306
## 3                   8             718.6497
## 4                   7             796.7151
## 5                   5             496.8324
## 6                   2             783.6663
```