

Linear Regression Analysis

SP

12/10/2019

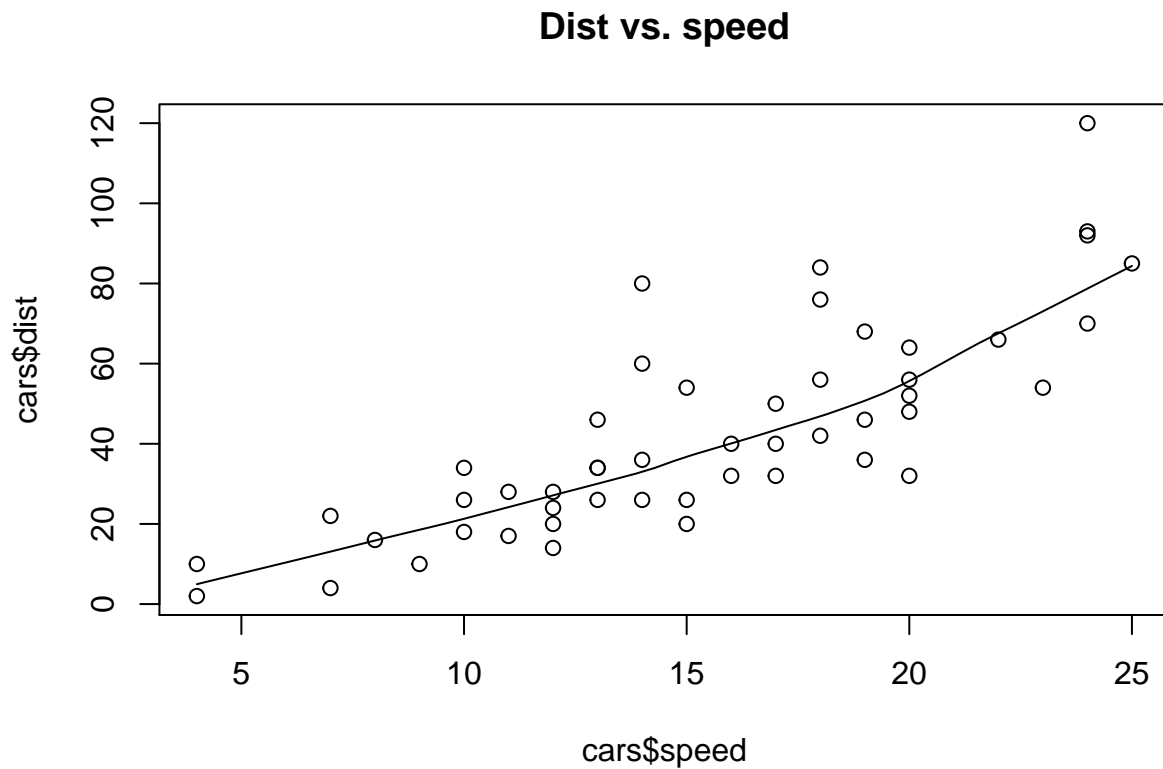
Linear Regression : Finding the mathematical relation for a continuous response variable Y as function of one or more X variables

```
head(cars)
```

```
##   speed dist
## 1     4     2
## 2     4    10
## 3     7     4
## 4     7    22
## 5     8    16
## 6     9    10
```

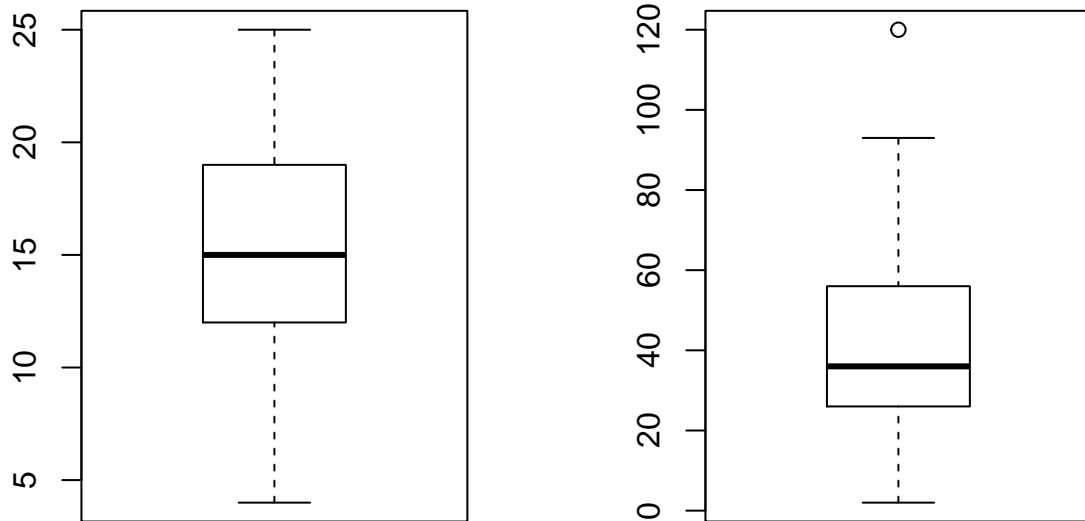
Check the relation ship

```
scatter.smooth(x=cars$speed, y=cars$dist, main="Dist vs. speed")
```



Check the out liers

```
par(mfrow=c(1,2))
boxplot(cars$speed)
boxplot(cars$dist)
```



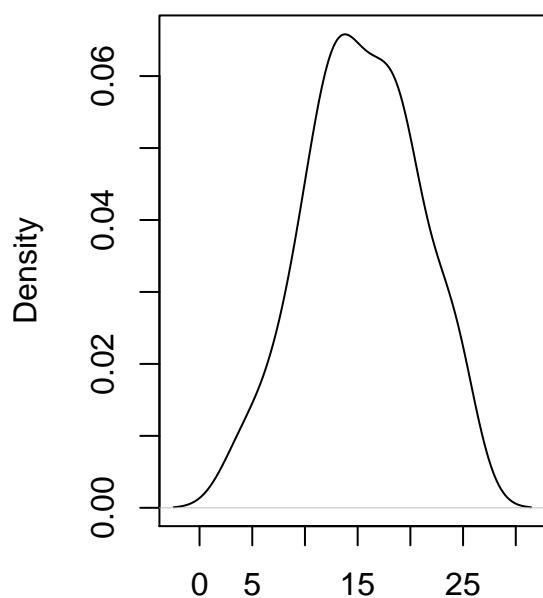
Density plots

```
library(e1071)
```

```
## Warning: package 'e1071' was built under R version 3.6.1
```

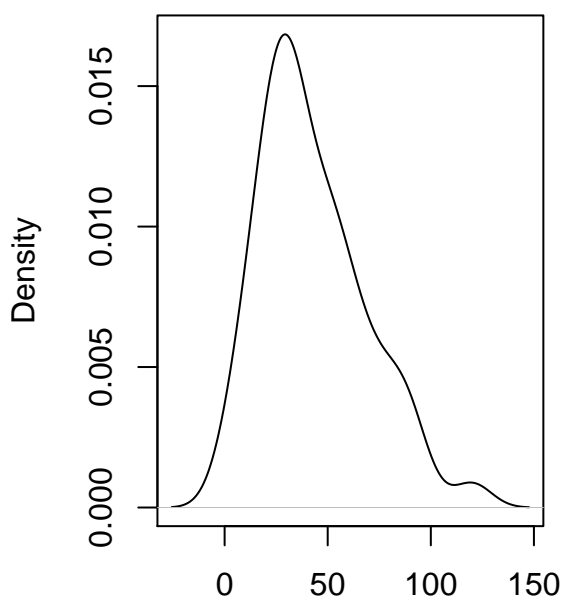
```
par(mfrow=c(1,2))
plot(density(cars$speed))
plot(density(cars$dist));
```

density.default(x = cars\$speed)



N = 50 Bandwidth = 2.15

density.default(x = cars\$dist)



N = 50 Bandwidth = 9.214

```
cor(cars$speed,cars$dist)
```

```
## [1] 0.8068949
```

Building the model

```
linearMod=lm(dist~speed, data=cars)
print(linearMod)
```

```
##
## Call:
## lm(formula = dist ~ speed, data = cars)
##
## Coefficients:
## (Intercept)      speed
##    -17.579       3.932
```

diagnosis

```
summary(linearMod)
```

```
##
## 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
```

```
AIC(linearMod)
```

```
## [1] 419.1569
```

```
BIC(linearMod)
```

```
## [1] 424.8929
```

```
set.seed(27)
trainRowIndex=sample(1:nrow(cars),0.8*nrow(cars))
traindata = cars[trainRowIndex,]
testdata = cars[-trainRowIndex,]
```

```
lmMod = lm(dist~speed, data = traindata)
distPred = predict(lmMod, testdata)
```

```
summary((lmMod))
```

```
##
## Call:
## lm(formula = dist ~ speed, data = traindata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -29.116  -9.608  -0.805  10.880  43.309
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -16.762     7.340  -2.284   0.0281 *
## speed        3.894     0.455   8.558 2.15e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.55 on 38 degrees of freedom
## Multiple R-squared:  0.6584, Adjusted R-squared:  0.6494
## F-statistic: 73.23 on 1 and 38 DF,  p-value: 2.147e-10
```

```
actuals_preds <- data.frame(cbind(actuals=testdata$dist, predicted=distPred))
correlation_accuracy <- cor(actuals_preds)
correlation_accuracy
```

```
##           actuals predicted
## actuals    1.0000000 0.7854391
## predicted 0.7854391 1.0000000
```

```
head(actuals_preds)
```

```
##    actuals predicted
## 6         10    18.28299
## 8         26    22.17687
## 13        20    29.96461
## 27        32    45.54010
## 30        40    49.43397
## 32        42    53.32784
```

```
# Min-Max Accuracy Calculation
```

```
min_max_accuracy <- mean(apply(actuals_preds, 1, min) / apply(actuals_preds, 1, max))
min_max_accuracy
```

```
## [1] 0.7466923
```

```
# MAPE Calculation
```

```
mape <- mean(abs((actuals_preds$predicted - actuals_preds$actuals))/actuals_preds$actuals)
mape
```

```
## [1] 0.333661
```

```
#install.packages('DMwR')
```

```
DMwR::regr.eval(actuals_preds$actuals, actuals_preds$predicted)
```

```
## Registered S3 method overwritten by 'xts':
##   method      from
##   as.zoo.xts zoo
```

```
## Registered S3 method overwritten by 'quantmod':
##   method      from
##   as.zoo.data.frame zoo
```

```
##           mae           mse           rmse           mape
## 12.552362 216.584424 14.716808 0.333661
```

```
library(DAAG)
```

```
## Warning: package 'DAAG' was built under R version 3.6.1
```

```
## Loading required package: lattice
```

```
#install.packages('DAAG')
```

```
cvResults <- suppressWarnings(CVlm(data = cars, form.lm=dist ~ speed, m=5, dots=FALSE, seed=29, legend.=
```

small symbols are predicted values while bigger ones are actuals.

