Data Scientist_ML_DTS

Andra

2022-07-04

```r
library(readr)
library(caret)
```

```
## Loading required package: ggplot2

## Loading required package: lattice
```

```r
library(rpart)
```

Analisa terhadap data hasil observasi dari beberapa pohon cherry.

```r
##Analisa Pohon
trees_df <- read_csv("https://storage.googleapis.com/dqlab-dataset/trees.csv")
```

```
## Rows: 31 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (3): Girth, Height, Volume
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
trees_df
```

```
## # A tibble: 31 x 3
##    Girth Height Volume
##    <dbl>  <dbl>  <dbl>
##  1   8.3     70   10.3
##  2   8.6     65   10.3
##  3   8.8     63   10.2
##  4  10.5     72   16.4
##  5  10.7     81   18.8
##  6  10.8     83   19.7
##  7  11       66   15.6
##  8  11       75   18.2
##  9  11.1     80   22.6
## 10  11.2     75   19.9
## # ... with 21 more rows
```

```r
names(trees_df)
```

```
## [1] "Girth"  "Height" "Volume"
```

```r
str(trees_df)
```

```
## spec_tbl_df [31 x 3] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Girth : num [1:31] 8.3 8.6 8.8 10.5 10.7 10.8 11 11 11.1 11.2 ...
##  $ Height: num [1:31] 70 65 63 72 81 83 66 75 80 75 ...
##  $ Volume: num [1:31] 10.3 10.3 10.2 16.4 18.8 19.7 15.6 18.2 22.6 19.9 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Girth = col_double(),
##   ..   Height = col_double(),
##   ..   Volume = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```r
names(trees_df)[1] <- "Diameter"
trees_df$diameter_ft <- trees_df$Diameter*0.08333
head(trees_df)
```

```
## # A tibble: 6 x 4
##   Diameter Height Volume diameter_ft
##      <dbl>  <dbl>  <dbl>       <dbl>
## 1      8.3     70   10.3       0.692
## 2      8.6     65   10.3       0.717
## 3      8.8     63   10.2       0.733
## 4     10.5     72   16.4       0.875
## 5     10.7     81   18.8       0.892
## 6     10.8     83   19.7       0.900
```

```r
summary(trees_df)
```

```
##     Diameter         Height       Volume       diameter_ft
##  Min.   : 8.30   Min.   :63   Min.   :10.20   Min.   :0.6916
##  1st Qu.:11.05   1st Qu.:72   1st Qu.:19.40   1st Qu.:0.9208
##  Median :12.90   Median :76   Median :24.20   Median :1.0750
##  Mean   :13.25   Mean   :76   Mean   :30.17   Mean   :1.1040
##  3rd Qu.:15.25   3rd Qu.:80   3rd Qu.:37.30   3rd Qu.:1.2708
##  Max.   :20.60   Max.   :87   Max.   :77.00   Max.   :1.7166
```

```r
##Shapiro Test
shapiro.test(trees_df$diameter_ft)
```

```
## 
##  Shapiro-Wilk normality test
## 
## data:  trees_df$diameter_ft
## W = 0.94117, p-value = 0.08893
```

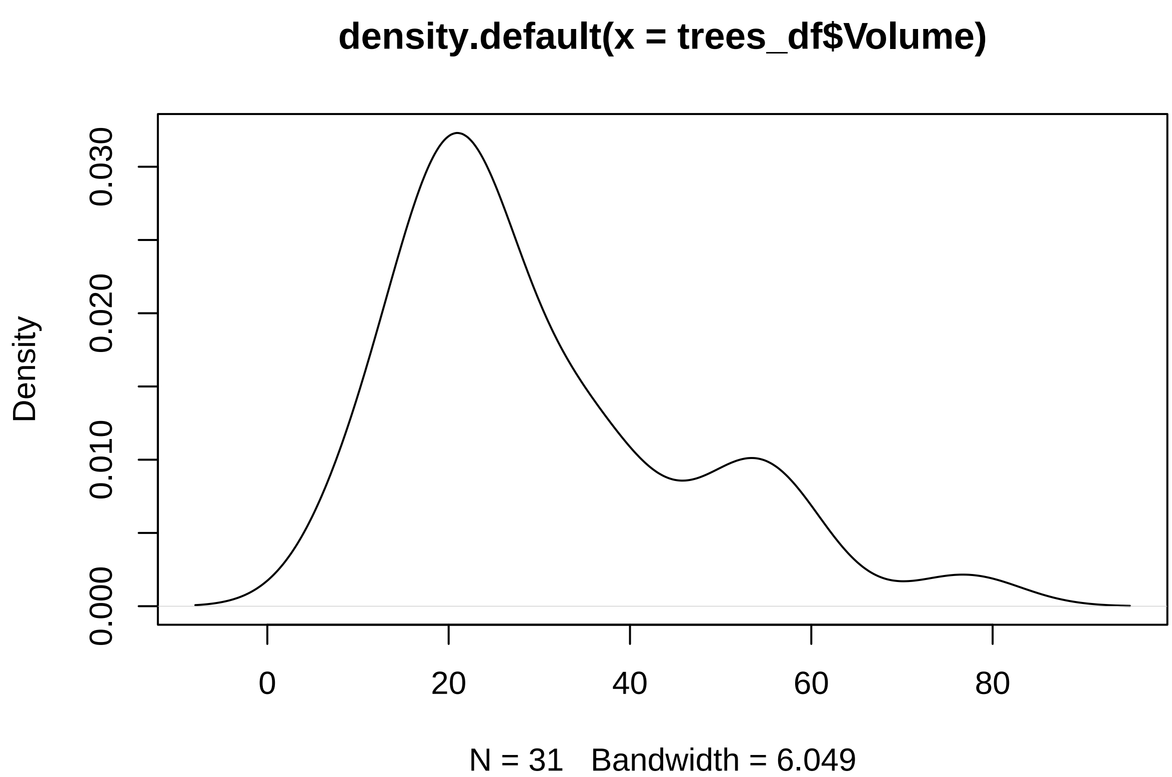
```r
shapiro.test(trees_df$Height)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  trees_df$Height
## W = 0.96545, p-value = 0.4034
```

```r
shapiro.test(trees_df$Volume)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  trees_df$Volume
## W = 0.88757, p-value = 0.003579
```

```r
#Visualisasi distribusi Volume
plot(density(trees_df$Volume))
```

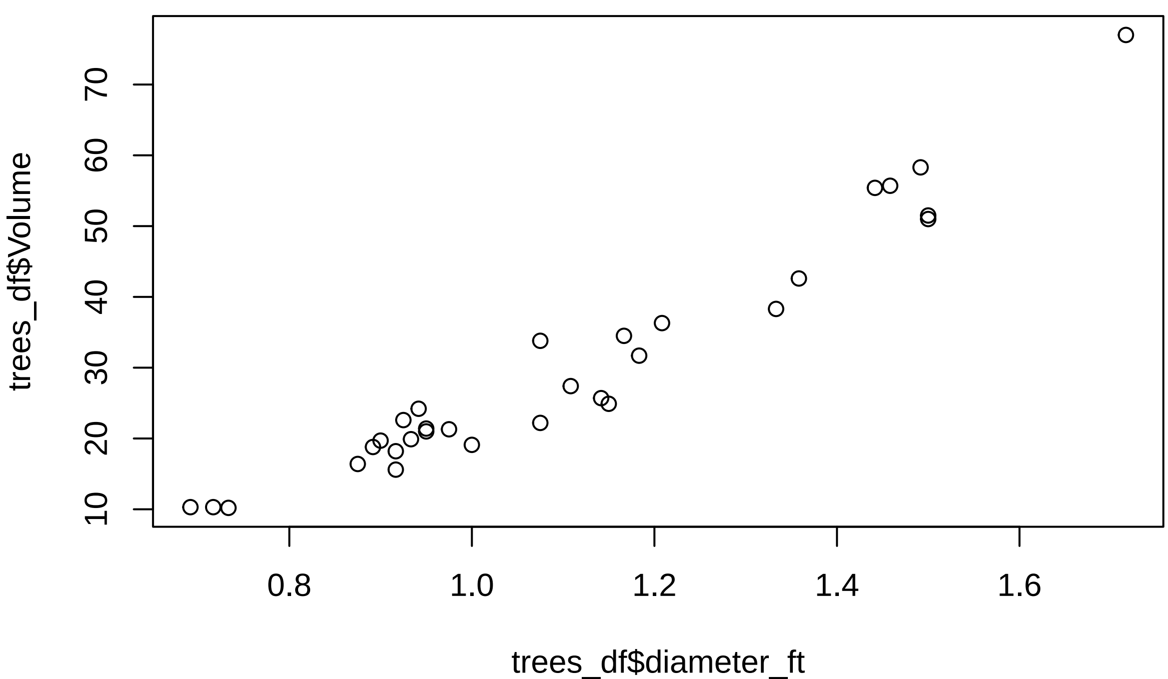



```r
##Mencari hubungan
lm(formula = Volume ~ Height + diameter_ft, data = trees_df)
```

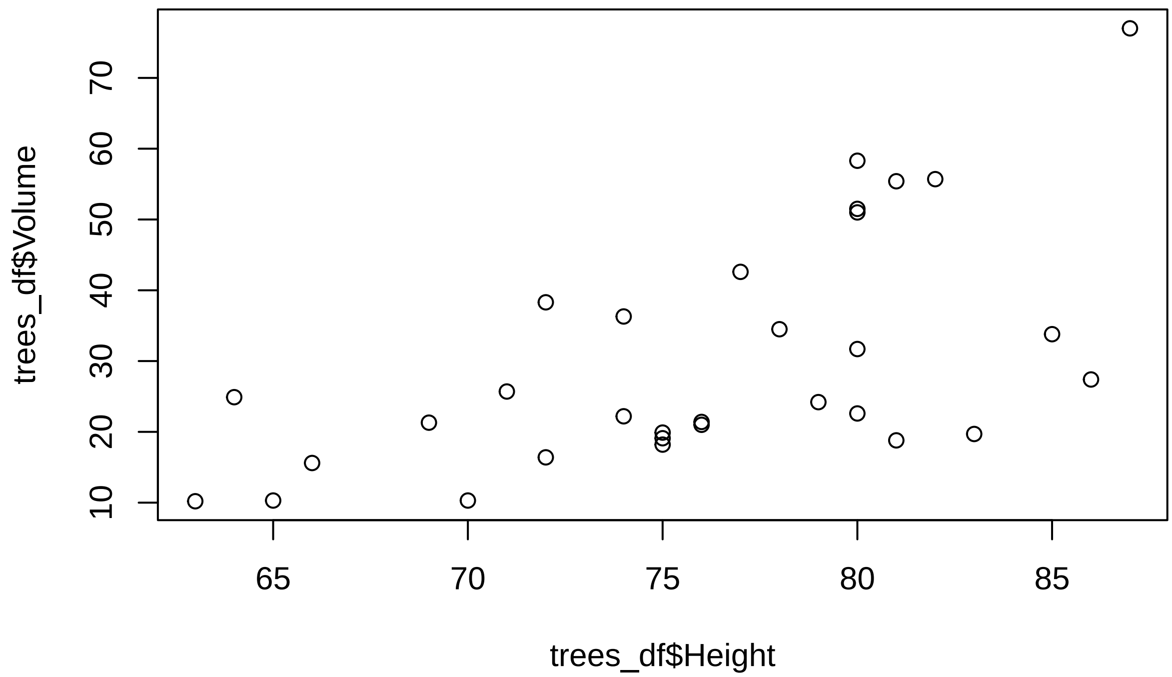
```
##
```

```
## Call:
## lm(formula = Volume ~ Height + diameter_ft, data = trees_df)
##
## Coefficients:
## (Intercept)       Height  diameter_ft
##    -57.9877       0.3393      56.5002
```

```
plot(trees_df$diameter_ft, trees_df$Volume)
```




```
plot(trees_df$Height,trees_df$Volume)
```



Penggunaan Machine Learning dalam analisis data terkait biaya listrik rumah tangga

```r
#read dataset
electric_bill <- read.csv("https://storage.googleapis.com/dqlab-dataset/electric_bill.csv")
```

```r
model <- lm(amount_paid ~ num_people+housearea, data = electric_bill)
model
```

```
##
## Call:
## lm(formula = amount_paid ~ num_people + housearea, data = electric_bill)
##
## Coefficients:
## (Intercept)   num_people    housearea
##     482.920        4.834        0.118
```

Analisa Data dengan Decision Tree

```r
iris <- read.csv("https://storage.googleapis.com/dqlab-dataset/iris.csv")
```

```r
#memecah dara train dan test
trainIndex <- createDataPartition(iris$Species, p=0.8, list=FALSE)
training_set <- iris[trainIndex, ]
testing_set <- iris[-trainIndex, ]

dim(training_set)
```

```
## [1] 120   5
```

```r
dim(testing_set)
```

```
## [1] 30  5
```

```r
#membuat model decison tree
set.seed(123)
model_dt <- rpart(Species ~., data = training_set, method = "class")
prediction_dt <- predict(model_dt, newdata = testing_set, type = "class")
```

```r
#evaluasi model dengan data test baru
testing_species = factor(testing_set$Species)
#memperlihatkan hasil evaluasi
eval_result <- confusionMatrix(prediction_dt, testing_species)
eval_result
```

```
## Confusion Matrix and Statistics
##
##             Reference
## Prediction   setosa versicolor virginica
##   setosa         10          0         0
##   versicolor      0         10         1
##   virginica       0          0         9
##
## Overall Statistics
##
##                Accuracy : 0.9667
##                  95% CI : (0.8278, 0.9992)
##     No Information Rate : 0.3333
##     P-Value [Acc > NIR] : 2.963e-13
##
##                   Kappa : 0.95
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: setosa Class: versicolor Class: virginica
## Sensitivity                 1.0000            1.0000           0.9000
## Specificity                 1.0000            0.9500           1.0000
## Pos Pred Value              1.0000            0.9091           1.0000
## Neg Pred Value              1.0000            1.0000           0.9524
## Prevalence                  0.3333            0.3333           0.3333
## Detection Rate              0.3333            0.3333           0.3000
## Detection Prevalence        0.3333            0.3667           0.3000
## Balanced Accuracy           1.0000            0.9750           0.9500
```