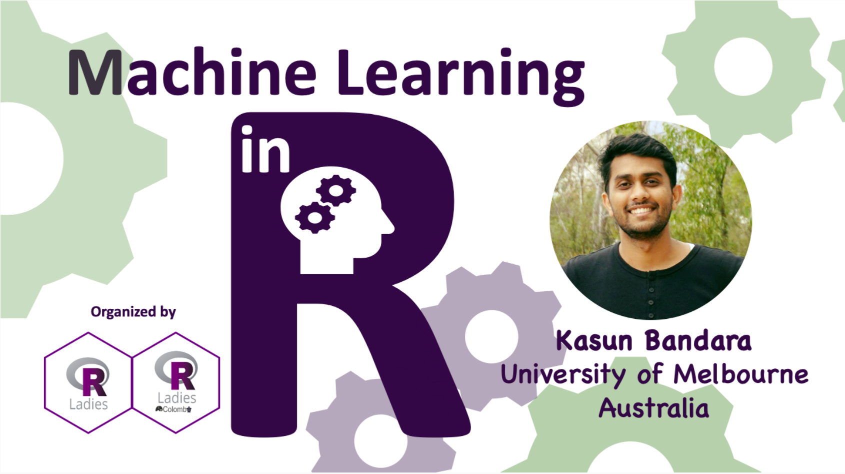
Machine Learning in R

R-Ladies Colombo Chapter

Kasun Bandara

29 March, 2021

Melbourne Centre for Data Science, University of Melbourne, Australia.


Machine Learning
in R
Organized by
R Ladies
R Ladies Colombo
Kasun Bandara
University of Melbourne
Australia

Introduction

About me

- 2015 Graduated in Computer Science from University of Colombo School of Computing
- 2015 Joined WSO2 Inc. as a Software Engineer
- 2016-2020 Ph.D. in Computer Science, Monash University, Australia
    - Topic: Forecasting In Big Data With Recurrent Neural Networks
    - Machine Learning for Time Series Forecasting
    - Research Internship at Walmart Labs, San Francisco, USA
    - Research Scientist at Turning Point, Melbourne, Australia
    - Data Science Tutor, Faculty of IT, Monash University
- 2021 Research Fellow, University of Melbourne

About me (2)

- Research Interests
  - Global Forecasting Models
  - Hierarchical Forecasting
  - Retail sales/demand forecasting
  - Renewable energy production forecasting (solar)
- Competition Fanatic !
  - M5 Forecasting Competition (**Gold Medalist**)
  - IEEE CIS Energy Forecasting Competition (**4th Place**)
  - Air-Liquide Energy Forecasting Competition (**4th Place**)
  - ANZ Customer Segmentation Challenge (**Top Performer**)

What is Data Science ?

Data Science is an interdisciplinary field that permits you to extract information from organized or unstructured data.

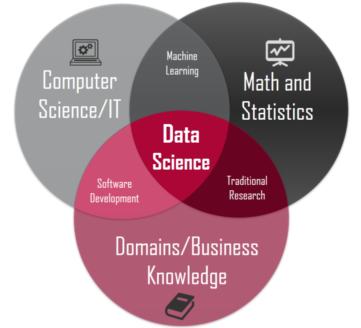


Figure 1: An intersection of many fields of science[1]

[1] Image source: https://medium.com/believing-these-8-myths-about-what-is-data-science-keeps-you-from-growing-528f1bd240dc

Data Science Life Cycle

Known as the O.S.E.M.N. framework.

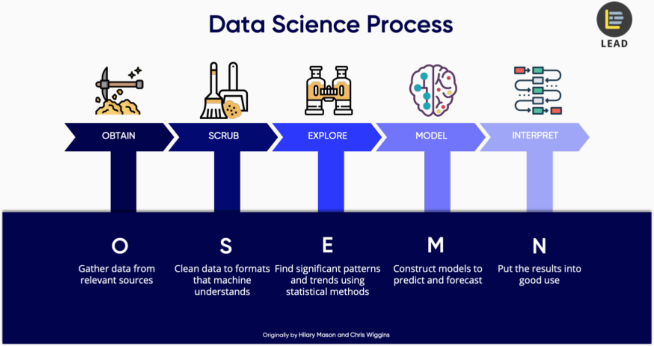


Figure 2: Data Science Process[2]

[2] Image source: https://towardsdatascience.com/5-steps-of-a-data-science-project-lifecycle-26c50372b492

Obtain (O)

- Retrieving data from multiple sources of inputs.
  - Structured Data: RDBMS, Tabular Data, CSV, TSV.
  - Unstructured Data: NoSQL Databases, API Data (Twitter, Facebook).
- Databases: `{odbc}`
- Scraping data from websites: `{rvest}`
- Data platforms: Kaggle, UCI, Competition Datasets, Government APIs

Example of {rvest}

```r
library(rvest)
library(dplyr)
set.seed(1234)

# reading the HTML page (Lord of the Rings)
lor_movie <- read_html("https://www.imdb.com/title/tt0120737/")

# Scraping the movie rating.
lor_movie %>%
  html_node("strong span") %>%
  html_text() %>%
  as.numeric()
#[1] 8.8

# Scraping the cast.
lor_movie %>%
  html_nodes("#titleCast .itemprop span") %>%
  html_text()

# Scraping the movie poster.
lor_movie %>%
  html_nodes("#img_primary img") %>%
  html_attr("src")
```

Scrub (S)

- Also known as **data pre-processing**, **data wrangling**.
- Converting the data into a unified, suitable format
  - Easier for the data exploration process.
  - What your predictive algorithm expects ?
  - **tidyverse** `{dplyr,tidyr,stringr,tibble,purr,ggplot2}`
- Handles data issues
  - Cleaning: Missing values, Outliers, Noisy data.
  - Transformation: Normalisation, Feature Discretization.
  - Reduction: Feature selection, Dimensionality reduction.

Missing Value Imputation

```r
library(simputation)
set.seed(1234)

# Loading iris dataset and randomly inserting NAs.
df <- iris
df_NA <- as.data.frame(lapply(df, function(imp) imp[ sample(c(TRUE, NA),
        prob = c(0.85, 0.15), size = length(imp), replace = TRUE)]))

# Using median to impute the missing values.
median_imputed <- impute_median(df_NA,
                                Sepal.Length ~ Species)

# Using linear regression to impute the missing values.
linear_imputed <- impute_lm(df_NA, Sepal.Length ~ Sepal.Width + Species)

# Using CART algorithm to impute the missing values.
cart_imputed <- impute_cart(df_NA, Species ~ .)

# Imputing multiple variables at once.
multivariable_imputed <- impute_rlm(df_NA, Sepal.Length + Sepal.Width
                                    ~ Petal.Length + Species)

# Imputing using a pre-trained model.
model <- lm(Sepal.Length ~ Sepal.Width + Species, data=iris)
model_imputed <- impute(df_NA, Sepal.Length ~ model)
```

Dealing with Outliers

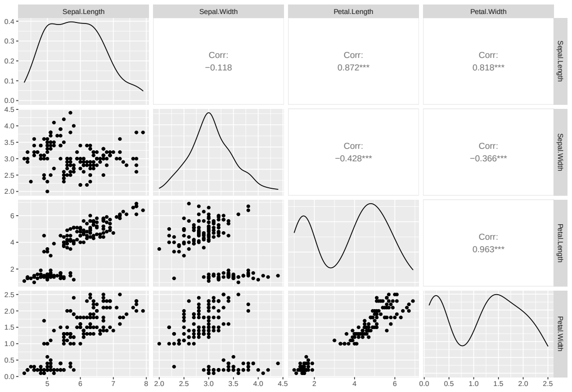
- A data point that differs significantly from other observations.
- Observations that distort your analysis.
    - Boxplot visualisation: `{ggplot2}`
    - Grubbs's test, Dixon's test, Rosner's test: `{outliers}`
    - Outlier detection algorithms: `{OutlierDetection}`
    - **outlierTest()** from `{car}`
    - **lofactor()** from `{DMwR}` (Local Outlier Factor)
- Anomaly detection is itself a different research area !
    - One Class SVM, IsolationForest
    - Unsupervised algorithms (Clustering)
    - Time series: `{tsoutliers,oddstream,stray}`

Feature Selection

- Removing redundant features from the dataset.
- Computational complexity, Address model overfitting.
- Filter Methods
    - Features are selected based on a statistical score.
    - Independent of any machine learning algorithm.
    - Pearson's Correlation, Chi-Square, PCA
- Wrapper Methods
    - A subset of features are used to train a model.
    - Forward, Backward, Recursive elimination.
    - Inbuilt penalization functions: LASSO, RIDGE regression
    - `{Boruta,caret,glmnet}`

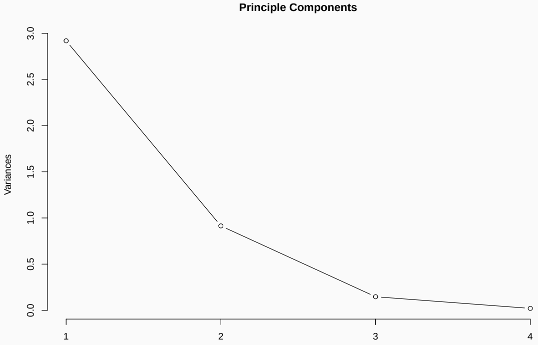
Using Correlation

```r
library(GGally)
library(dplyr)
set.seed(1234)

# Plotting the feature correlations.
iris %>% select(-Species) %>% ggpairs()
```



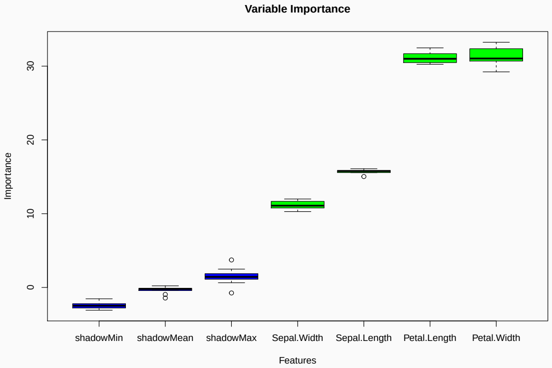
Using PCR

```r
library(dplyr)
set.seed(1234)

# Plotting the feature importance.
pcomp_df <- iris %>%
  select(-Species) %>% prcomp(scale. = T, center = T) %>%
  plot(type="l", main = "Principle Components")
```



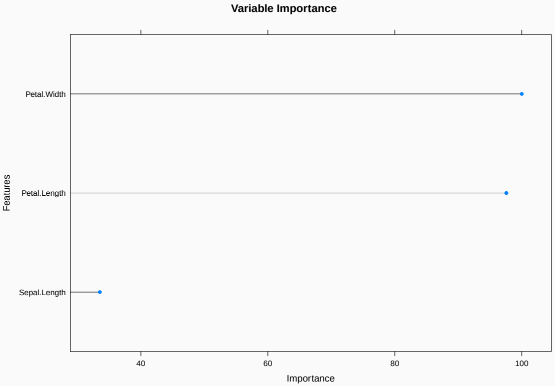
Example of {Boruta}

```r
library(Boruta)
set.seed(1234)

# Boruta is a feature selection algorithm based on the random forests algorithm.
boruta_df <- Boruta(Species ~ ., data=iris, doTrace=0)

# Plotting the feature importance.
plot(boruta_df, xlab="Features", main="Variable Importance")
```



Example of {caret}

```r
library(caret)
set.seed(1234)

# Build a decision tree model using rpart (Recursive Partitioning And Regression Trees)
rPart_df <- train(Species ~ ., data=iris, method="rpart")
rPart_imp <- varImp(rPart_df)

# Plotting the feature importance.
plot(rPart_imp, top = 3, main='Variable Importance', ylab = "Features")
```



Explore (E)

- Examination of data, features, and their characteristics.
    - Data types: numerical, ordinal, and nominal data.
    - Summary statistics.
    - Feature distributions.
    - Feature correlations (positive, negative).
    - Classification: class distribution (**Class Imbalance?**)
- Invest your time more on the data exploration process.
    - Frequency distribution: **Histograms**
    - Outlier detection: **Box plots**
    - Feature correlation analysis: **Scatter plots**
    - Time series analysis: **Trend and Seasonal plots**

Tools available for Exploration

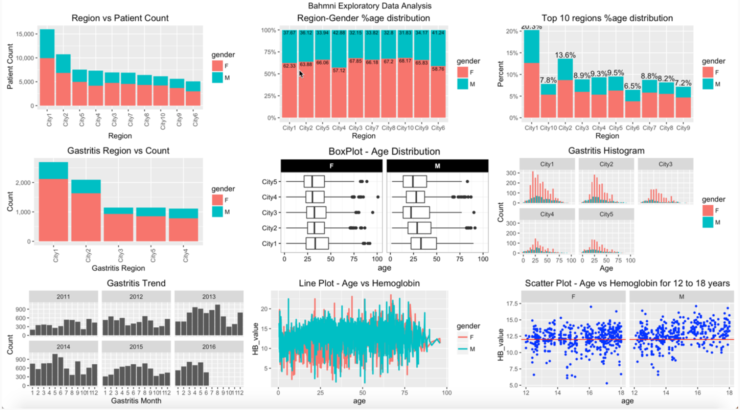


Figure 3: Plots available from `{ggplot2}`[3]

[3] Image source: https://www.pinterest.com.au/pin/281686151677624808/

Seasonal plot from {feasts}

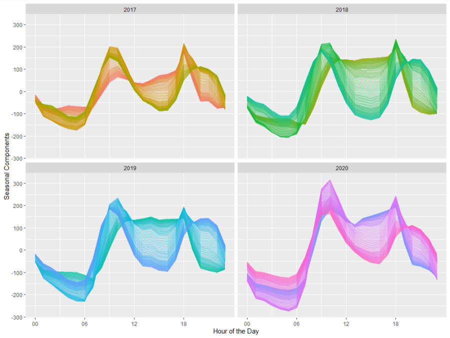


Figure 4: The presence of multiple seasonal cycles[4]

[4] Github repo: https://github.com/kasungayan/Meldatathon2020

Title formats

Animation (using $\LaTeX$)

- This is important

Animation (using $\LaTeX$)

- This is important
- Now this

Animation (using $\LaTeX$)

- This is important
- Now this
- And now this

Animation (using $\LaTeX$)

- This is really important
- Now this
- And now this

Simple list

- Kasun
- Now this
- And now this

Tables (using LaTeX})

References i