

## PROJECT: BIKESHARE CAPITAL RENTAL FORECAST

### Forecasting Bicycle Rental Demand

#### Feature Selection

**Este código contém comandos para feature selection**

#### 1. Working Directory

### Configurando o diretório de trabalho

```
setwd("C:/Users/Utilizador/repos/Formacao_cientista_de_dados/big_data_analytics_R_microsoft_azure_machine_learning")  
getwd()
```

**Observação:** Este código contém comandos para filtrar e transformar os dados de aluguel de bikes,

**Este código foi criado para executar tanto no Azure, quanto no RStudio;**

**Para executar no Azure, altere o valor da variavel Azure para TRUE;**

**Ou seja, se o valor for FALSE, o código sera executado no RStudio;**

### Variável que controla a execução do script

```
Azure <- FALSE  
if(Azure){ source("src/Tools.R") bikes <- maml.mapInputPort(1) bikes$dteday <- set.asPOSIXct(bikes)  
}else{ bikes <- bikes }  
dim(bikes) any(is.na(bikes))
```

### Random Forest

**Criando um modelo para descobrir as variaveis mais relevantes para depois construir o modelo preditivo**

**Criando um modelo para identificar os atributos com maior importância para o modelo preditivo**

```
require(randomForest)
```

**Avalidando a importância de todas as variaveis**

**Target: cnt**

```
modelo <- randomForest(cnt ~ . , data = bikes, ntree = 100, nodesize = 10, importance = TRUE)
```

**Removendo variáveis colineares**

```
modelo <- randomForest(cnt ~ . - mnth - hr - workingday - isWorking - dayWeek - xformHr - workTime  
- holiday - windspeed - monthCount - weathersit, data = bikes, ntree = 100, nodesize = 10, importance =  
TRUE)
```

**Plotando as variáveis por grau de importância**

```
varImpPlot(modelo)
```

**Gravando o resultado**

```
df_saida <- bikes[, c("cnt", rownames(modelo$importance))]  
if(Azure) maml.mapOutputPort("df_saida")
```