RF_FF_Tests

Daniel Kojis

August 1, 2019

```
source('simData_methods.R')
library(WGCNA)
```

```
## Loading required package: dynamicTreeCut
```

```
## Loading required package: fastcluster
```

```
##
## Attaching package: 'fastcluster'
```

```
## The following object is masked from 'package:stats':
##
##     hclust
```

```
##
```

```
##
## Attaching package: 'WGCNA'
```

```
## The following object is masked from 'package:stats':
##
##     cor
```

```
library(randomForest)
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
library("fuzzyforest")
library(dplyr)
```

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

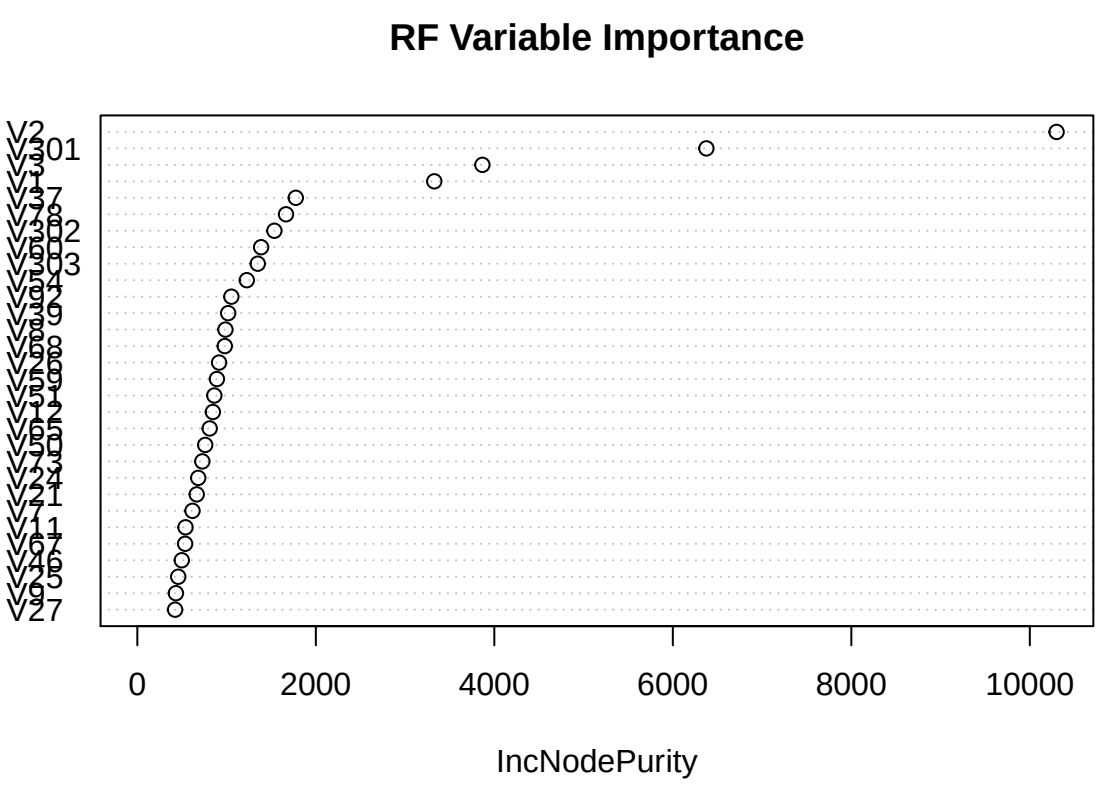
```
##
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:randomForest':
##
##     combine
```

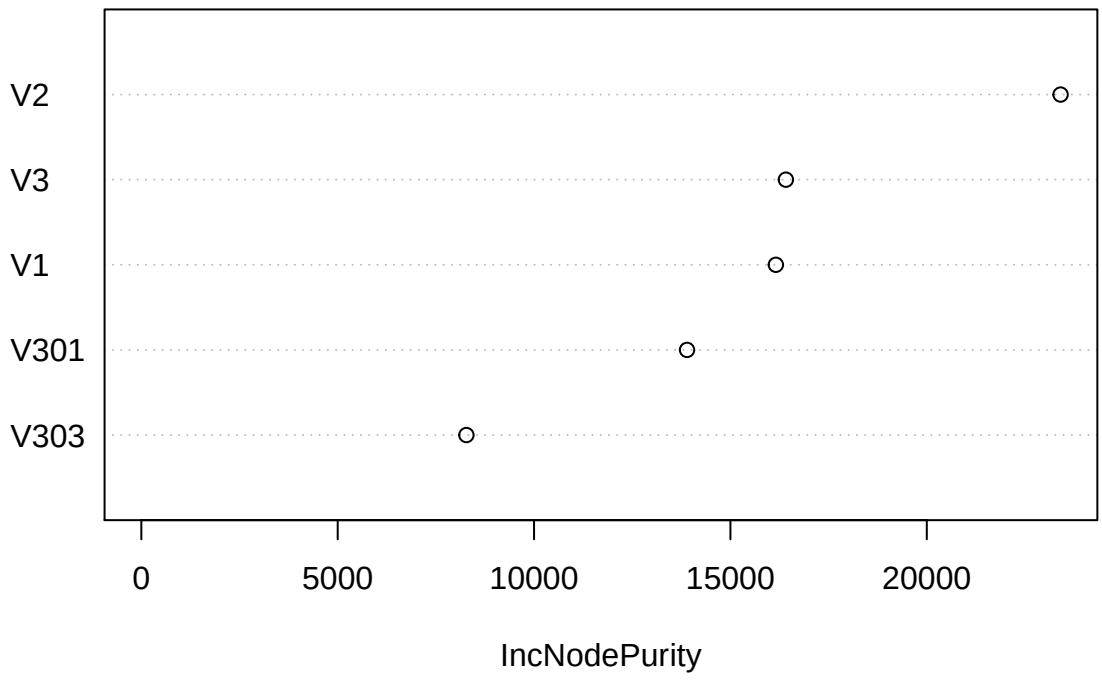
```
## The following object is masked from 'package:MASS':
##
##     select
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

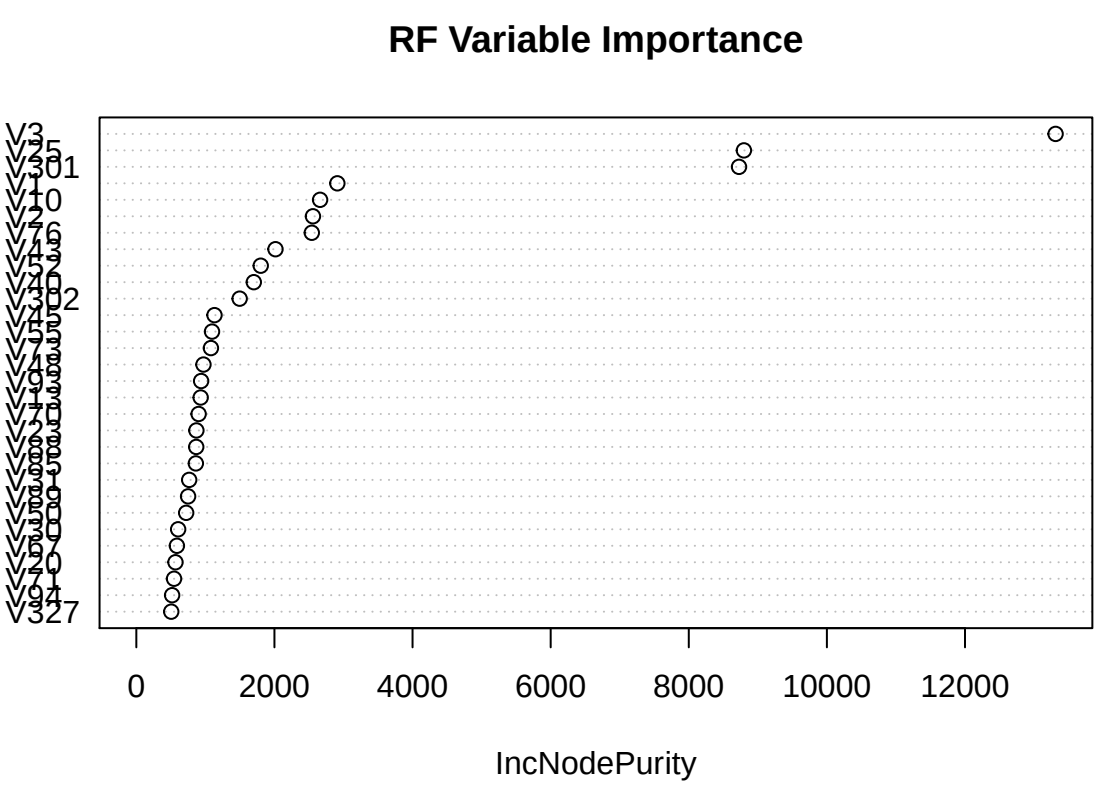
```
library(magrittr)



# CS Structured Data
######################

# data <- read.csv("data_CS.csv")
# data %<>% dplyr::select(-'X')
# Alternatively, generate new CS data
 data <- as.data.frame(sim_2(100,5))

X = data[,-ncol(data)] # remove y value from for X matrix
y = data[,ncol(data)] # assign target variable


# Random Forest
rf <- randomForest(X,y)
varImpPlot(rf,type=2,main="RF Variable Importance")
```



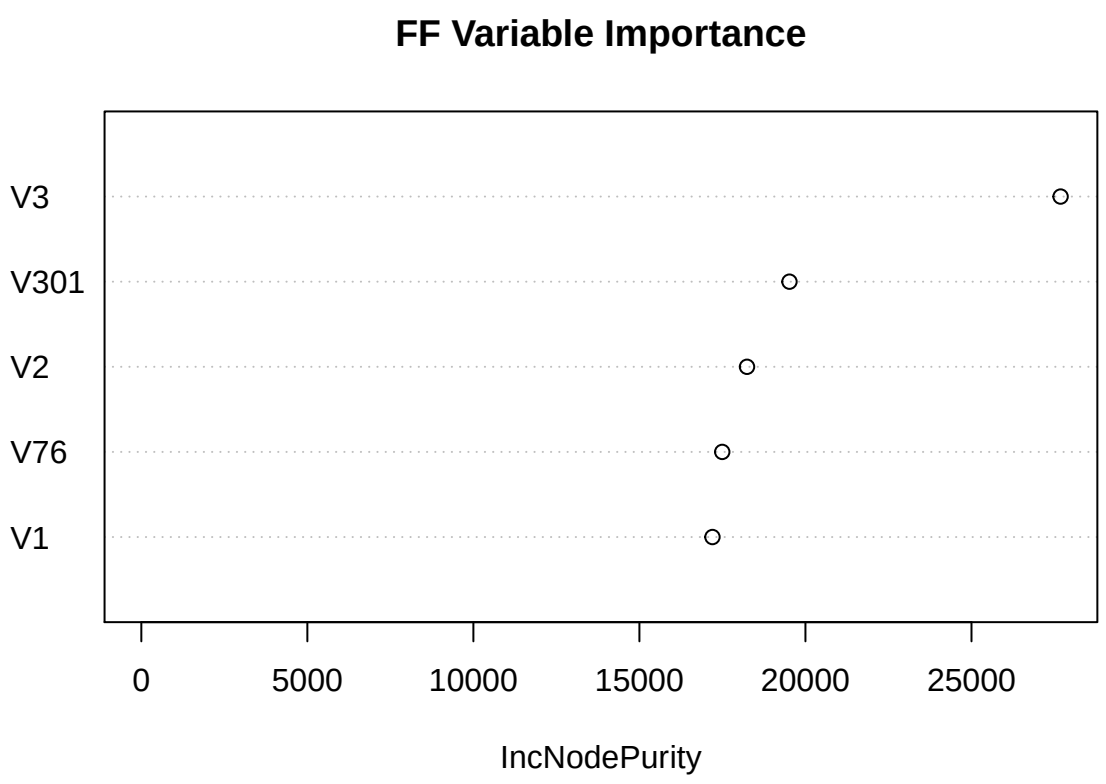

```
# Fuzzy Forest
wff = wff(X,y)
varImpPlot(wff$final_rf,type=2,main=" FF Variable Importance")
```



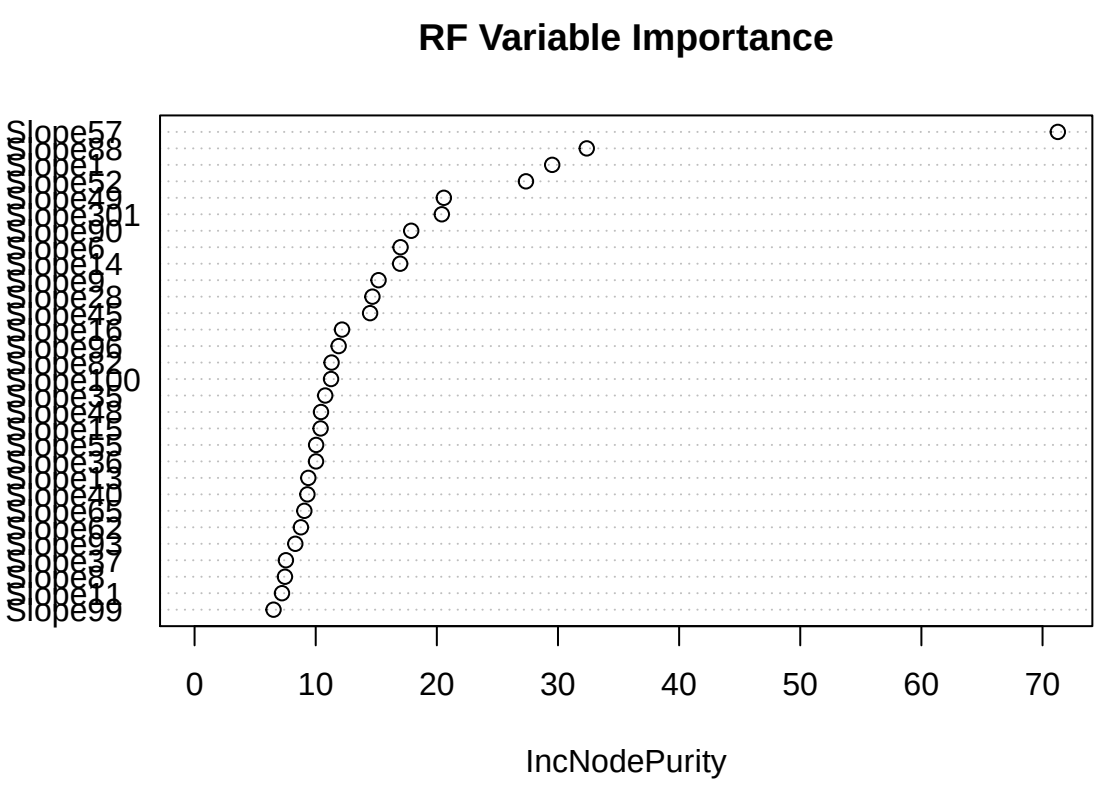

```
# AR Structured Data
######################
# data <- read.csv("data_AR.csv")
# data %<>% dplyr::select(-'X')
data <- as.data.frame(simAR_dif_cor(n=100,T=5,group_cor=c(0.7,0.75,0.8,0) ))

X = data[,-ncol(data)] # remove y value from for X matrix
y = data[,ncol(data)] # assign target variable


# Random Forest
rf <- randomForest(X,y)
varImpPlot(rf,type=2,main="RF Variable Importance")
```



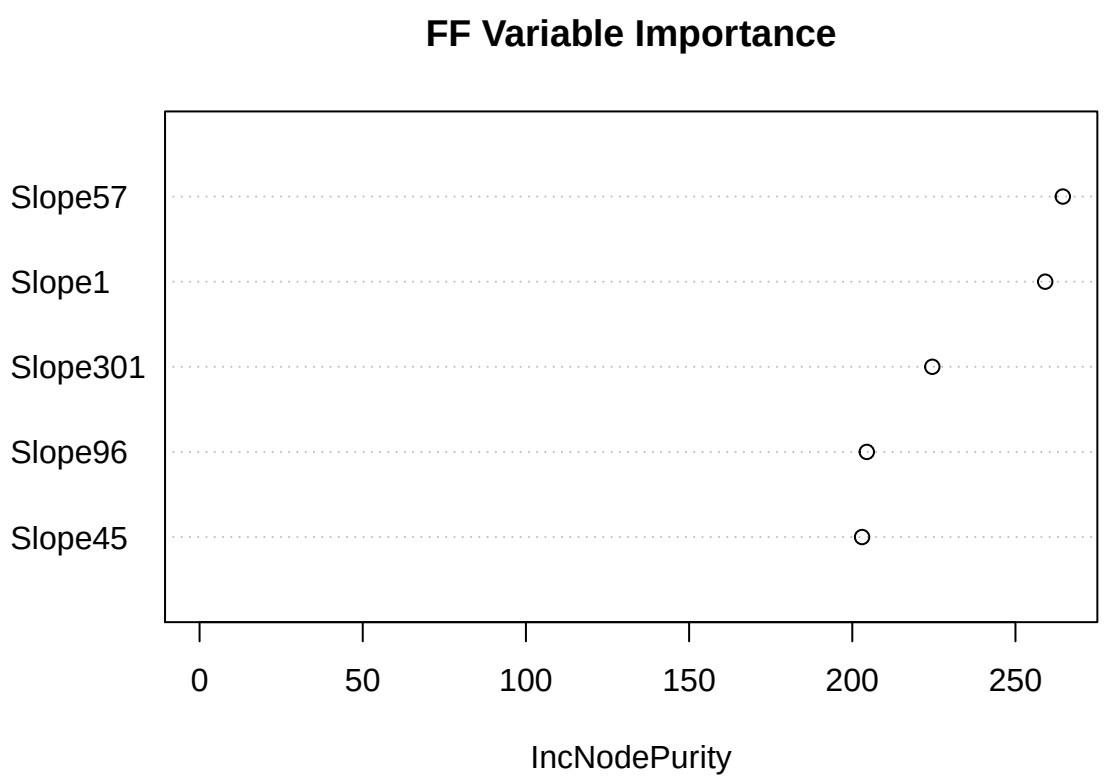

```
# Fuzzy Forest
wff = wff(X,y)
varImpPlot(wff$final_rf,type=2,main="FF Variable Importance")
```




```r
# Linear model reduced data
data <- read.csv('lm_reduced_data.csv')
# data <- data # run lm_reduce_data (create function)
X <- data %>%  dplyr::select(-c( 'X', 'time', 'y_cat', 'y') )
y <-  data$y


# Random Forest
rf <- randomForest(X,y)
varImpPlot(rf,type=2,main="RF Variable Importance")
```




```
# Fuzzy Forest
wff = wff(X,y)
varImpPlot(wff$final_rf,type=2,main="FF Variable Importance")
```




```
# ## categorical
#
# data <- read.csv('lm_reduced_data.csv')
# # data <- data # run lm_reduce_data (create function)
# X <- data %>%  dplyr::select(-c( 'X', 'time', 'y_cat', 'y') )
# y <-  data$y_cat
#
# # Random Forest
# rf <- randomForest(X,y)
# varImpPlot(rf,type=2,main="RF Variable Importance")
#
#
# # Fuzzy Forest
# wff = wff(X,y,)
# varImpPlot(wff$final_rf,type=2,main="FF Variable Importance")
#
```