Project Overview

This is a transnational data set which contains all the transactions occurring between 01/12/2010 and 09/12/2011 for a UK-based and registered non-store online retail.The company mainly sells unique all-occasion gifts. Many customers of the company are wholesalers.

The following libraries were used in this project:

**library**(methods)

**library**(recommenderlab)

**library**(data.table)

**library**(ggplot2)

**library**(knitr)

Data Pre-preprocessing

Some pre-processing of the data available is required before creating the recommendation system.

df\_data <- fread('../input/data.csv')

df\_data[ ,InvoiceDate := as.Date(InvoiceDate)]

Data Imputation

There is negatif Quantity & Unit Price, also NULL/NA Customer ID. We will delete all the NA Row

df\_data[Quantity<=0,Quantity:=NA]

df\_data[UnitPrice<=0,UnitPrice:=NA]

df\_data <- na.omit(df\_data)

Item Dictionary

Create a Item Dictionary which allows an easy search of a Item name by any of its StockCode

setkeyv(df\_data, c('StockCode', 'Description'))

itemCode <- unique(df\_data[, c('StockCode', 'Description')])

setkeyv(df\_data, NULL)

Convert from long fromat to wide format

Convert from transactional to binary metrix, 0 for no transaction and vice versa

df\_train\_ori <- dcast(df\_data, CustomerID ~ StockCode, value.var = 'Quantity',fun.aggregate = sum, fill=0)

CustomerId <- df\_train\_ori[,1] *#!*

df\_train\_ori <- df\_train\_ori[,-c(1,3504:3508)]

*#Fill NA with 0*

**for** (i **in** names(df\_train\_ori))

df\_train\_ori[is.na(get(i)), (i):=0]

Convert Wide Format to sparse matrix

In order to use the ratings data for building a recommendation engine with recommenderlab, I convert buying matrix into a sparse matrix of type realRatingMatrix.

df\_train <- as.matrix(df\_train\_ori)

df\_train <- df\_train[rowSums(df\_train) > 5,colSums(df\_train) > 5]

df\_train <- binarize(as(df\_train, "realRatingMatrix"), minRatin = 1)

Training

We will use Item Base Collaboratife Filtering or IBCF. Jaccard is used becouse our data is binary

Split Dataset

Dataset is split Randomly with 80% for training and 20% for test

which\_train <- sample(x = c(TRUE, FALSE), size = nrow(df\_train),replace = TRUE, prob = c(0.8, 0.2))

y <- df\_train[!which\_train]

x <- df\_train[which\_train]

Training parameter

Let’s have a look at the default parameters of IBCF model. Here, k is the number of items to compute the similarities among them in the first step. After, for each item, the algorithm identifies its k most similar items and stores the number. method is a similarity funtion, which is Cosine by default, may also be pearson. I create the model using the default parameters of method = Cosine and k=30.

recommender\_models <- recommenderRegistry$get\_entries(dataType ="binaryRatingMatrix")

recommender\_models$IBCF\_binaryRatingMatrix$parameters

## $k

## [1] 30

##

## $method

## [1] "Jaccard"

##

## $normalize\_sim\_matrix

## [1] FALSE

##

## $alpha

## [1] 0.5

Training Dataset

method <- 'IBCF'

parameter <- list(method = 'Jaccard')

n\_recommended <- 5

n\_training <- 1000

recc\_model <- Recommender(data = x, method = method, parameter = parameter)

model\_details <- getModel(recc\_model)

Predict

Test Dataset is split randomly, We only use 20% for test.Return value of prediction is top-N-List of recommendation item for each user in test dataset.

recc\_predicted <-predict(object = recc\_model, newdata=y,n = n\_recommended, type="topNList")

Recomendation for

Recomendation item for first 5 user in training dataset

as(recc\_predicted,"list")[1:5]

## $`4`

## [1] "21668" "21672" "22845" "21671" "21670"

##

## $`6`

## [1] "22697" "20972" "72709" "37450" "22057"

##

## $`7`

## [1] "20675" "20674" "21240" "21242" "21243"

##

## $`8`

## [1] "22057" "48194" "21523" "37446" "21524"

##

## $`11`

## [1] "21975" "22417" "48138" "23307" "22951"

user\_1 <- CustomerId[as.integer(names(recc\_predicted@items[1]))]

these are the recommendations for user: 12352

vvv <- recc\_predicted@items[[1]]

vvv <- rownames(model\_details$sim)[vvv]

itemCode[vvv]

## StockCode Description

## 1: 21668 RED STRIPE CERAMIC DRAWER KNOB

## 2: 21672 WHITE SPOT RED CERAMIC DRAWER KNOB

## 3: 22845 VINTAGE CREAM CAT FOOD CONTAINER

## 4: 21671 RED SPOT CERAMIC DRAWER KNOB

## 5: 21670 BLUE SPOT CERAMIC DRAWER KNOB

Compaire to actual purchase

Bellow is actual purchase of user: 12352. If we look name or description of the goods,Recommendations given are close to the actual purchase.

user\_1\_buy <- df\_data[CustomerID==user\_1, sum(Quantity), by=StockCode]

merge(itemCode,user\_1\_buy, by='StockCode')

## StockCode Description V1

## 1: 21669 BLUE STRIPE CERAMIC DRAWER KNOB 12

## 2: 22138 BAKING SET 9 PIECE RETROSPOT 3

## 3: 22178 VICTORIAN GLASS HANGING T-LIGHT 6

## 4: 22624 IVORY KITCHEN SCALES 2

## 5: 22627 MINT KITCHEN SCALES 2

## 6: 22635 CHILDS BREAKFAST SET DOLLY GIRL 2

## 7: 22654 DELUXE SEWING KIT 6

## 8: 22668 PINK BABY BUNTING 5

## 9: 22844 VINTAGE CREAM DOG FOOD CONTAINER 4

## 10: 22978 PANTRY ROLLING PIN 6

## 11: 22982 PANTRY PASTRY BRUSH 12

## 12: 23088 ZINC HEART FLOWER T-LIGHT HOLDER 16

## 13: 23089 GLASS BON BON JAR 12

## 14: 23096 PETIT TRAY CHIC 12

## 15: 23367 SET 12 COLOUR PENCILS SPACEBOY 16

## 16: 23368 SET 12 COLOUR PENCILS DOLLY GIRL 16

## 17: 23559 WOODLAND BUNNIES LOLLY MAKERS 6

## 18: 84050 PINK HEART SHAPE EGG FRYING PAN 12

## 19: M Manual 3

## 20: POST POSTAGE 3