Notes on the CDNOW Master Data Set

Peter S. Fader
www.petefader.com

Bruce G. S. Hardie
www.brucehardie.com†

October 2013

1. Introduction

The file CDNOW_master.txt[1] contains the entire purchase history up to the end of June 1998 of the cohort of 23,570 individuals who made their first-ever purchase at CDNOW in the first quarter of 1997. (See Fader and Hardie (2001a) for further details of this dataset.) Each record in this file, 69,659 in total, comprises four fields: the customer's ID, the date of the transaction, the number of CDs purchased, and the dollar value of the transaction. A 1/10th systematic sample of the whole cohort (2357 customers)[2] has become a canonical dataset for those developing models of buyer behaviour in noncontractual settings.

The primary purpose of this note is to document the process of creating the 1/10th sample of the complete dataset using MATLAB. We also show how to create the summary of this dataset used in Fader and Hardie (2001a), and close with a comment on how the full dataset is used in Fader et al. (2005).

2. Creating the 1/10th Sample

Our goal is to extract a 1/10 sample from the full CDNOW database that is based on total repeat spend in the first 39 weeks, stratified by week of trial.

We have found that using what we will call the "xMAT" data structure simplifies the process of manipulating datasets such as this. The idea is that we record activity in "transaction" time, rather than calendar time. Element (i, j) of TransMAT gives us the time of customer i's jth transaction,

[1] See http://brucehardie.com/datasets/CDNOW_master.zip

[2] See http://brucehardie.com/datasets/CDNOW_sample.zip

element (i, j) of QuantMAT gives us the number of CDs associated with that transaction, and element (i, j) of SpendMAT gives us the dollar value of that transaction. We create these matrices in the following manner.

- We load the data into MATLAB (assuming the appropriate `path` command has been executed) and create a vector for each field:

```
load CDNOW_master.txt;

CustID = CDNOW_master(:,1);   % customer id
Date   = CDNOW_master(:,2);   % transaction date
Quant  = CDNOW_master(:,3);   % number of CDs purchased
Spend  = CDNOW_master(:,4);   % dollar value (excl. S&H)
```

- In order to specify the dimensions of TransMAT, QuantMAT and SpendMAT, we need to know the number of customers in the database and the maximum number of transactions made by any one individual:

```
NumCust = length(unique(CustID));
tmpMaxTrans = max(histc(CustID,unique(CustID)));
```

- We will find it useful to have transaction time recorded in such a manner that 1 January 1997 corresponds to day 1:

```
PurchDay = Datenump(Date) - Datenump(19970101) + 1;
```

  (See the appendix for details of the function `Datenump`.)

- The following code populates TransMAT, QuantMAT and SpendMAT. (Note that information on the jth transaction is stored in column $j+1$; column 1 contains the customer ID.) Our basic unit of time is day, so multiple transactions by a customer on any given day are aggregated into one; as a result, the final maximum number of transactions may be less than the value of tmpMaxTrans computed above.

```
CurrDay = 0;
CurrID = 0;
CustIDNum = 0;
MaxPurchNum = 0;

tmp_TransMAT = zeros(NumCust,tmpMaxTrans+1);
tmp_QuantMAT = zeros(NumCust,tmpMaxTrans+1);
tmp_SpendMAT = zeros(NumCust,tmpMaxTrans+1);
```

```
for i = 1:length(CustID)
   if CustID(i) ~= CurrID
      CurrID = CustID(i);
      CustIDNum = CustIDNum + 1;
      PurchNum = 0;
      CurrDay = 0;
      tmp_TransMAT(CustIDNum,1)=CustID(i);
      tmp_QuantMAT(CustIDNum,1)=CustID(i);
      tmp_SpendMAT(CustIDNum,1)=CustID(i);
   end

   if PurchDay(i) == CurrDay
      tmp_QuantMAT(CustIDNum,PurchNum+1) = ....
         tmp_QuantMAT(CustIDNum,PurchNum+1) + Quant(i);
      tmp_SpendMAT(CustIDNum,PurchNum+1) = ....
         tmp_SpendMAT(CustIDNum,PurchNum+1) + Spend(i);
   else
      PurchNum = PurchNum + 1;
      if PurchNum > MaxPurchNum
         MaxPurchNum = PurchNum;
      end
      tmp_TransMAT(CustIDNum,PurchNum+1) = PurchDay(i);
      tmp_QuantMAT(CustIDNum,PurchNum+1) = Quant(i);
      tmp_SpendMAT(CustIDNum,PurchNum+1) = Spend(i);
      CurrDay = PurchDay(i);
   end
end

master_TransMAT = tmp_TransMAT(:,1:MaxPurchNum+1);
master_QuantMAT = tmp_QuantMAT(:,1:MaxPurchNum+1);
master_SpendMAT = tmp_SpendMAT(:,1:MaxPurchNum+1);
```

We now create the 1/10th sample.

- We first compute the number of repeat purchases made by each customer in the first 39 weeks (273 days) of 1997 and the dollar value of these purchases:

```
x = sum((master_TransMAT(:,3:end) > 0 & ....
         master_TransMAT(:,3:end) <= 273),2);

RptSpend = zeros(NumCust,1);
for i = 1:NumCust
    RptSpend(i) = sum(master_SpendMAT(i,2:x(i)+2),2) ....
        - master_SpendMAT(i,2);
end
```

(Note that master_SpendMAT(i,2) is the value of the first (i.e., "trial") purchase.)

- For each trial week, we sort customers (in descending order) on repeat spend in the first 39 weeks and determine the ID of every 10th customer:

```
id = [];
for i = 1:12
   istrier = find(master_TransMAT(:,2) > 7*(i-1) ....
      & master_TransMAT(:,2) <= 7*i);
   [y,j] = sort(-RptSpend(istrier));
   id = [id istrier(j)'];
end
tmpindx = 10*[1:floor(length(id)/10)];
sampled = id(tmpindx)';
sampledID = master_TransMAT(sampled,1);
```

- We extract the rows of CustID, Date, Quant, and Spend corresponding to the sampled customers and save the records to the file `sample.txt`:

```
tmpindx = ismember(CustID,sampledID);
fid = fopen('D:\sample.txt','w');
fprintf(fid,' %05d %8.0f %2.0f %7.2f\r\n', ....
   [ CustID(tmpindx) Date(tmpindx) ....
     Quant(tmpindx) Spend(tmpindx) ]' );
fclose(fid);
```

  This is the same as the file `CDNOW_sample.txt`[3] (excluding the new Customer ID field) sorted by the first field (the customer's ID in the master dataset). (The file `CDNOW_sample.txt` is sorted by the new Customer ID numbers (ranging from 1 to 2357); the fact that this is different from the sorting by master dataset Customer ID simply reflects the way the new ID numbers (ranging from 1 to 2357) were created, and is immaterial.)

We now document the process of creating summaries of the 1/10th sample data given the xMAT data structure. We divide the 78 weeks in half: Period 1 is a 39-week calibration period while Period 2 is a 39-week longitudinal holdout used for model validation.

- We first create the xMAT matrices for the 1/10th sample:

```
% what is the maximum number of repeat transactions in the
% sampled set of customers?
i = 2;
while max(master_TransMAT(sampledID,i+1)) > 0,
```

[3] See `http://brucehardie.com/datasets/CDNOW_sample.zip`

```
    i = i + 1;
end

TransMAT = master_TransMAT(sampledID,1:i);
QuantMAT = master_QuantMAT(sampledID,1:i);
SpendMAT = master_SpendMAT(sampledID,1:i);
```

- The number of repeat transactions made by each customer in each period is computed in the following manner:

```
calwk = 273; % 39 week calibration period
NumHH = size(TransMAT,1);

p1x = sum((TransMAT(:,3:end) > 0 & ....
           TransMAT(:,3:end) <= calwk),2);
p2x = sum((TransMAT(:,3:end) > 0 & ....
           TransMAT(:,3:end) > calwk),2);
```

- The number of CDs purchased and total spend across these repeat transactions is computed in the following manner:

```
p1Quant = zeros(NumHH,1);
p2Quant = zeros(NumHH,1);
p1Spend = zeros(NumHH,1);
p2Spend = zeros(NumHH,1);
for i = 1:NumHH
   if p1x(i) == 0
      p1Quant(i) = 0;
      p1Spend(i) = 0;
   else
      p1Quant(i) = sum(QuantMAT(i,3:2+p1x(i)));
      p1Spend(i) = sum(SpendMAT(i,3:2+p1x(i)));
   end
   p2Quant(i) = sum(QuantMAT(i,3+p1x(i):end));
   p2Spend(i) = sum(SpendMAT(i,3+p1x(i):end));
end
```

- The average spend per repeat transaction is computed as follows:

```
mx = zeros(NumHH,1);
tmpindx = p1x>0;
mx(tmpindx) = p1Spend(tmpindx)./p1x(tmpindx);
```

- When fitting models such as the Pareto/NBD and BG/NBD to these data, we also want to know the "recency" information for each customer, as well as their effective calibration period:

```
% time of last calibration period repeat purchase (in weeks)
tx = [];
for i = 1:NumHH
   tx(i) = TransMAT(i,2+p1x(i)) - TransMAT(i,2);
end
tx = tx'/7;
% effective calibration period (in weeks)
T = (calwk - TransMAT(:,2))/7;
```

See Fader and Hardie (2008) for details of how some of these summary measures are computed in Excel.

3. Creating the Fader and Hardie (2001a) Summaries

Fader and Hardie (2001a) was the first paper to make use of the CDNOW dataset. To further illustrate use of the xMAT data structure, we show how to create the various summaries used in the analysis (Fader and Hardie 2001b).

- What is the total number of CDs purchased each week?

```
TotQuant = zeros(78,1);
for i = 1:78
   weekQuant = zeros(NumCust,1);
   for k = 2:MaxPurchNum+1
      isbuyer = find(master_TransMAT(:,k) > 7*(i-1) ....
                  & master_TransMAT(:,k) <= 7*i);
      weekQuant(isbuyer) = weekQuant(isbuyer)  + ....
         master_QuantMAT(isbuyer,k);
   end
   TotQuant(i) = sum(weekQuant);
end
```

  These are the "Actual" data plotted in Figure 2 of Fader and Hardie (2001a).

- How many people made their first-ever ("trial") purchase each week?

```
NumTriers = zeros(12,1);
for i = 1:12
   istrier = find(master_TransMAT(:,2) > 7*(i-1) ....
               & master_TransMAT(:,2) <= 7*i);
   NumTriers(i) = length(istrier);
end
```

  This gives us the "Incremental triers" row in Table 1 of Fader and Hardie (2001a).

- What is the total number of CDs purchased by triers in their trial week?

```
TrialQuant = zeros(12,1);
for i = 1:12
   istrier = find(master_TransMAT(:,2) > 7*(i-1) ....
               & master_TransMAT(:,2) <= 7*i);
   for j = 1:length(istrier)
      not_done = true;
      k = 2;
      while not_done,
         TrialQuant(i) = TrialQuant(i) + ....
            master_QuantMAT(istrier(j),k);
         not_done = (master_TransMAT(istrier(j),k+1) > ....
            7*(i-1) & master_TransMAT(istrier(j),k+1) <= 7*i);
         k = k+1;
      end
   end
end
```

These are the "Actual" trial data plotted in Figure 1 of Fader and Hardie (2001a). (Note that in this paper the unit of time is week. As a result, any repeat purchasing by a customer in their trial week is added to that associated with their first-ever purchase from CDNOW.)

- What is the distribution of the number of units purchased in each of the first 12 weeks?

```
weekQuant = zeros(NumCust,12);
for i = 1:12
   for k = 2:MaxPurchNum+1
      isbuyer = find(master_TransMAT(:,k) > 7*(i-1) ....
                  & master_TransMAT(:,k) <= 7*i);
      weekQuant(isbuyer,i) = weekQuant(isbuyer,i)  + ....
         master_QuantMAT(isbuyer,k);
   end
end

MaxQuant = max(max(weekQuant));
QuantDist = zeros(MaxQuant,12);
for i = 1:12
   QuantDist(:,i) = histc(weekQuant(:,i),1:MaxQuant);
end
```

Given these week-by-week distributions, we can create Table 1 of Fader and Hardie (2001a):

```
TableOne = zeros(11,12);
TableOne(2:10,:) = QuantDist(1:9,:);
TableOne(11,:) = sum(QuantDist(10:end,:));
TableOne(1,:) = cumsum(NumTriers)' - sum(TableOne(2:end,:));
```

4. Comment on Fader et al. (2005)

In closing, we comment on the use of this dataset in Fader et al. (2005). The initial exploratory analysis presented in the paper uses the full dataset (23,570 customers) excluding the purchasing data for ten buyers who purchased more than $4,000 worth of CDs across the 78-week period. Having validated the model on the 1/10 sample, the final RFM-group analysis is based on the revised "full" dataset of 23,560 customers.

References

Fader, Peter S. and Bruce G. S. Hardie (2001a), "Forecasting Repeat Sales at CDNOW: A Case Study," *Interfaces*, **31** (May–June), Part 2 of 2, S94–S107.

Fader, Peter S. and Bruce G. S. Hardie (2001b), "A Note on Implementing the Fader and Hardie "CDNOW Model"." <http://brucehardie.com/notes/002/>

Fader, Peter S. and Bruce G. S. Hardie (2008), "Creating an RFM Summary Using Excel." <http://brucehardie.com/notes/022/>

Fader, Peter S., Bruce G. S. Hardie, and Ka Lok Lee (2005), "RFM and CLV: Using Iso-value Curves for Customer Base Analysis," *Journal of Marketing Research*, **42** (November), 415–430.

Appendix

```
function out = Datenump(DateNumber,OptionNumber)
% out = DATENUMP(DateNumber,OptionNumber)
%
% makes up for the deficiency in the standard
% DATENUM function to allow for a date input
% in the number format YYYYMMDD
% or string format "YYYYMMDD".
%
% 'out' is then a date of the format depending on
%   OptionNumber = 1 => out = [Year Month Day]
%   OptionNumber = 2 => out = Matlab date integer <-- DEFAULT
%   OptionNumber = 'x' or 'X' => out = Excel date integer
%
% see also:  DATENUM, DATESTR, M2XDATE, X2MDATE
%
% written by:  PNath@London.edu
%

if ischar(DateNumber)
   DateNumber = str2num(DateNumber);
end

YearNumber = floor(DateNumber/10000);
MonthNumber = floor((DateNumber-YearNumber*10000)/100);
DayNumber = floor((DateNumber - YearNumber*10000 - MonthNumber*100));

if nargin == 1
   out = datenum(YearNumber,MonthNumber,DayNumber);
else
   switch OptionNumber
   case 1
      out = [YearNumber MonthNumber DayNumber];
   case 2
      out = datenum(YearNumber,MonthNumber,DayNumber);
   case 'x'
      out = m2xdate(datenum(YearNumber,MonthNumber,DayNumber));
   case 'X'
      out = m2xdate(datenum(YearNumber,MonthNumber,DayNumber));
   otherwise
   end

end
```

Source: http://phd.london.edu/pnath.phd98/Matlab/Utilities/Datenump.m
Accessed: 2013-08-12