Data Exploration Script

Paula McMahon

August 2019

```
knitr::opts_chunk$set(echo = TRUE)
library(readxl)
library(corrplot)
library(tree)
library(randomForest)
library(e1071)
library(MASS)
library(moments)
library(rpart)
library(rpart.plot)
library(class)
library(caret)
library(gbm)
library(lme4)
library(lubridate)
library(segmented)
```

Read in the full data

```
# Read in DORMANT customer data
dormant_customers <-
  read_excel("~/college/ST606_Project/data_files/dormant-transactions.xlsx")

# Read in CURRENT customer data
current_customers <-
  read_excel("~/college/ST606_Project/data_files/stayed-shopping-transactions.xlsx")
```

Check for missing values

```
sapply(dormant_customers, function(x) sum(is.na(x)))
sapply(current_customers, function(x) sum(is.na(x)))
```

Merge the current and dormant datasets.

```
# Add a variable "Churn" to the dormant dataset and set it equal to 1
# i.e. the customer has stopped shopping.
# Add a variable "Churn" to the current dataset and set it equal to 0
# i.e. the customer is a current shopper.
current_customers$Churn = 0
dormant_customers$Churn = 1

# Not all of the variables in the dataset will be used:
# OrderRef is a unique reference number and is not of any value.
```

```r
# ExpectedGoodsCharge is 98% correlated with ActualCharge, so drop
# ExpectedGoodsCharge from analysis.
# OverallSubstitutionPolicy is a categorical variable and not valuable
# for aggregated analysis.
# TotalOrderLines is an extra variable in current dataset but not dormant,
# so drop from analysis.
# TotalItemsApprovedPicks is 100% correlated with TotalPickedLines so
# drop TotalItemsApprovedPicks from analysis
# TotalQtyOrdered is 99% correlated with TotalOrderItems so drop
# TotalQtyOrdered from analysis.
# AvailabilityPostSubPercentage is 99.9% correlated with
# AvailabilityPreSubPercentage so drop AvailabilityPostSubPercentage from analysis.

merged_customers <-
  rbind(dormant_customers[c("sequenceID","StoreID", "ExpectedFulfillmentCharge",
                            "SlotStartDate","ActualCharge","TotalOrderItems",
                            "TotalPickedLines","TotalQtySubbed","TotalQtyOOS",
                            "TotalPickTimeSeconds","AvailabilityPreSubPercentage",
                            "PercentageOutOfStocks","PercentageSubstitutions",
                            "ValueOfSubstitutions","ValueOfOutOfStocks",
                            "RelatedCallsCount","Churn")],
        current_customers[c("sequenceID","StoreID","ExpectedFulfillmentCharge",
                            "SlotStartDate","ActualCharge","TotalOrderItems",
                            "TotalPickedLines","TotalQtySubbed","TotalQtyOOS",
                            "TotalPickTimeSeconds","AvailabilityPreSubPercentage",
                            "PercentageOutOfStocks","PercentageSubstitutions",
                            "ValueOfSubstitutions","ValueOfOutOfStocks",
                            "RelatedCallsCount","Churn")])
```

Explore each variable

```r
# Variable ExpectedFulfillmentCharge
boxplot(merged_customers$ExpectedFulfillmentCharge)

# Variable ActualCharge
boxplot(merged_customers$ActualCharge)

# Variable TotalOrderItems
boxplot(merged_customers$TotalOrderItems)

# Variable TotalPickedLines
boxplot(merged_customers$TotalPickedLines)

# Variable TotalQtySubbed
boxplot(merged_customers$TotalQtySubbed)

# Variable TotalQtyOOS
boxplot(merged_customers$TotalQtyOOS)

# Variable TotalPickTimeSeconds
boxplot(merged_customers$TotalPickTimeSeconds)
```

```
# Variable AvailabilityPreSubPercentage
boxplot(merged_customers$AvailabilityPreSubPercentage)

# Variable PercentageOutOfStocks
boxplot(merged_customers$PercentageOutOfStocks)

# Variable PercentageSubstitutions
boxplot(merged_customers$PercentageSubstitutions)

# Variable ValueOfSubstitutions
boxplot(merged_customers$ValueOfSubstitutions)

# Variable ValueOfOutOfStocks
boxplot(merged_customers$ValueOfOutOfStocks)

# Variable RelatedCallsCount
boxplot(merged_customers$RelatedCallsCount)
```

```
# Create factors
merged_customers$Churn <- as.factor(merged_customers$Churn)
merged_customers$StoreID <- as.factor(merged_customers$StoreID)
dim(merged_customers)
```

Create a variable that will allow exploration, in terms of time

Find the number of days between each transaction date and a "base" date. This variable will be called DaysSinceLastShop.

```
# The last shopping date in the "dormant" dataset is 10/05/2018.
# So use a date slightly after that as the "basedate"
basedate <- "01/06/18"
basedate <- as.Date(basedate, "%d/%m/%y")

# create a new variable, length = number of rows in dataset
merged_customers$DaysSinceLastShop = numeric(30977)

# subtract each SlotStartDate from the basedate and give the answer in days
for (i in 1:nrow(merged_customers)) merged_customers$DaysSinceLastShop[i] <-
as.vector(difftime(basedate, merged_customers$SlotStartDate[i], units="days"))
# We want days to be an integer value
merged_customers$DaysSinceLastShop <- ceiling(merged_customers$DaysSinceLastShop)
```

Histograms of customers vs frequency of shops

```
# this output shows how many times each customer shopped
table(merged_customers$sequenceID)
# this shows a histogram of it
hist(table(merged_customers$sequenceID),
     main="Histogram of how many times customers shopped",
     xlab="Number of online shopping transactions", col="deepskyblue")
# how many customers shopped over 50 times
sum(table(merged_customers$sequenceID)>=50)
```

```r
# precisely which customers shopped over 50 times
which(table(merged_customers$sequenceID)>=50)

# We know that of their customer number (sequenceID) <= 1000, they are dormant
# We know that of their customer number (sequenceID) > 1000, they are still shopping
most_freq_dorm_shoppers <-
  c(6,195,197,201,439,441,554,557,573,640,642,662,881,885,931,942)

most_freq_curr_shoppers <- c(1037,1061,1064,1072,1076,1128,1134,1140,1156,1165,1180,1181,
                             1188,1201,1205,1207,1266,1316,1317,1318,1334,1338,1349,1369,1397,
                             1403,1422,1443,1497,1512,1537,1552,1555,1577,1583,1606,1619,1633,
                             1717,1726,1748,1751,1794,1865,1868,1948,1829,1994)

par(mfrow=c(1,2))
for (i in 1:16) {
  with(merged_customers, hist(DaysSinceLastShop[sequenceID == most_freq_dorm_shoppers[i]],
                              breaks=seq(0,800,by=100),
                              main=paste("Dormant customer, seqID",
                                         most_freq_dorm_shoppers[i]),
                              xlab="Data span measured in days",
                              col="deepskyblue2"))
}

for (i in 1:48) {
  with(merged_customers, hist(DaysSinceLastShop[sequenceID == most_freq_curr_shoppers[i]],
                              breaks=seq(0,400,by=50),
                              main=paste("Current customer, seqID",
                                         most_freq_curr_shoppers[i]),
                              xlab="Data span measured in days",
                              col="deeppink3"))
}
```

Plots of current customers vs the amount they spent over time

```r
# which customers have shopped more than 20 times
which(table(merged_customers$sequenceID)>=20)

# strip out the dormant customers and put the rest in a vector
curr <- c(1001, 1002, 1003, 1005, 1011, 1012, 1014, 1016, 1017, 1022, 1027, 1028, 1032,
          1033, 1034, 1035, 1037, 1038, 1039, 1040, 1041, 1045, 1046,  1051, 1054, 1055,
          1058, 1060, 1061, 1062, 1064, 1065, 1067, 1069, 1070, 1071, 1072, 1073, 1074,
          1076, 1077, 1079, 1080, 1081, 1082, 1085, 1087, 1088, 1089, 1095, 1096, 1098,
          1099, 1105, 1107, 1108, 1111, 1112, 1114, 1117, 1118, 1120, 1121, 1124, 1127,
          1128, 1129, 1131, 1133, 1134, 1135, 1137, 1140, 1141, 1142, 1143, 1144, 1145,
          1151, 1152, 1153, 1155, 1156, 1159, 1161, 1164, 1165, 1166, 1167, 1168, 1169,
          1171, 1172, 1173, 1174, 1176, 1177, 1179, 1180, 1181, 1182, 1183, 1184, 1185,
          1187, 1188, 1190, 1191, 1192, 1193, 1194, 1195, 1198, 1199, 1201, 1202, 1205,
          1207, 1208, 1209, 1212, 1214, 1216, 1220, 1221, 1222, 1224, 1230, 1231, 1232,
          1233, 1235, 1236, 1237, 1239, 1240, 1241, 1242, 1244, 1247, 1249, 1250, 1251,
          1252, 1253, 1257, 1258, 1259, 1266, 1270, 1271, 1272, 1273, 1274, 1276, 1278,
          1279, 1280, 1281, 1285, 1286, 1288, 1289, 1293, 1294, 1296, 1297, 1298, 1299,
          1301, 1302, 1303, 1306, 1307, 1309, 1311, 1312, 1316, 1317, 1318, 1320, 1321,
          1323, 1325, 1327, 1328, 1329, 1331, 1332, 1333, 1334, 1337, 1338, 1339, 1340,
          1341, 1344, 1345, 1346, 1347, 1348, 1349, 1350, 1352, 1353, 1354, 1356, 1357,
          1359, 1360, 1362, 1364, 1365, 1366, 1369, 1371, 1377, 1379, 1381, 1382, 1387,
          1388, 1390, 1392, 1393, 1394, 1395, 1396, 1397, 1401, 1402, 1403, 1404, 1405,
          1407, 1408, 1411, 1412, 1413, 1414, 1415, 1417, 1419, 1421, 1422, 1424, 1426,
          1427, 1429, 1430, 1432, 1433, 1436, 1438, 1439, 1440, 1441, 1442, 1443, 1445,
          1446, 1447, 1450, 1451, 1452, 1456, 1458, 1459, 1460, 1463, 1464, 1465, 1466,
          1467, 1468, 1469, 1470, 1472, 1474, 1475, 1476, 1478, 1479, 1481, 1483, 1490,
          1491, 1492, 1493, 1494, 1497, 1498, 1500, 1501, 1508, 1509, 1510, 1511, 1512,
          1513, 1514, 1515, 1516, 1517, 1518, 1519, 1520, 1529, 1530, 1531, 1532, 1536,
          1537, 1538, 1544, 1545, 1547, 1548, 1549, 1551, 1552, 1553, 1554, 1555, 1556,
          1558, 1563, 1566, 1567, 1568, 1569, 1570, 1573, 1574, 1576, 1577, 1578, 1582,
          1583, 1584, 1586, 1587, 1588, 1589, 1590, 1591, 1592, 1593, 1594, 1595, 1596,
          1597, 1598, 1599, 1600, 1601, 1602, 1606, 1607, 1608, 1609, 1610, 1611, 1612,
          1614, 1618, 1619, 1625, 1626, 1627, 1628, 1629, 1630, 1631, 1632, 1633, 1634,
          1635, 1636, 1637, 1639, 1641, 1642, 1644, 1645, 1647, 1652, 1653, 1655, 1661,
          1662, 1663, 1667, 1671, 1672, 1673, 1674, 1675, 1676, 1677, 1678, 1679, 1680,
          1686, 1687, 1689, 1690, 1692, 1701, 1703, 1704, 1705, 1706, 1707, 1708, 1709,
          1711, 1713, 1715, 1716, 1717, 1718, 1721, 1725, 1726, 1728, 1729, 1730, 1732,
          1734, 1735, 1737, 1738, 1739, 1744, 1748, 1749, 1750, 1751, 1753, 1756, 1758,
          1759, 1763, 1764, 1765, 1766, 1767, 1768, 1769, 1770, 1771, 1774, 1779, 1781,
          1782, 1783, 1784, 1786, 1787, 1793, 1794, 1795, 1800, 1801, 1803, 1804, 1805,
          1806, 1807, 1809, 1811, 1812, 1814, 1816, 1817, 1824, 1825, 1826, 1827, 1828,
          1829, 1830, 1832, 1833, 1835, 1836, 1837, 1838, 1839, 1840, 1846, 1847, 1848,
          1849, 1850, 1851, 1852, 1859,1860, 1861, 1862, 1863, 1864, 1865, 1866, 1867,
          1868, 1869, 1872, 1874, 1875, 1876, 1883, 1887, 1888, 1890, 1891, 1892, 1893,
          1894, 1895, 1907, 1909, 1911, 1912, 1913, 1914, 1915, 1916, 1917, 1922, 1923,
          1925, 1926, 1927, 1930, 1932, 1934, 1945, 1947, 1948, 1949, 1950, 1957, 1959,
          1961, 1966, 1967, 1976, 1978, 1979, 1980, 1981, 1982, 1983, 1989, 1990, 1992,
          1993, 1994, 1995, 1996, 1998, 1999, 2000)

length(curr)

for (i in 1:566) {
xxx <- qplot(merged_customers$DaysSinceLastShop[merged_customers$sequenceID==curr[i]],
      merged_customers$ActualCharge[merged_customers$sequenceID==curr[i]],
      xlab="Time Span of Transactions measured in Days",
      ylab="Transaction Charge in euro",
      main=paste("Spending pattern over time, for current customer, sequenceID", curr[i]))
print(xxx)
}
```

Finding a breakpoint where a customers shopping pattern starts to change

```
# Focus on one customer (sequenceID=1996). Extract the number of days since they
# last shopped and the amount they spent

# create the y-axis variable
charge <- merged_customers$ActualCharge[merged_customers$sequenceID==1996]
# create the x-axis variable
days <- merged_customers$DaysSinceLastShop[merged_customers$sequenceID==1996]
```

```r
# Use the segmented function to fit a regression model with a segmented relationship
# between response and explanatory variable
lin.mod <- lm(charge ~ days)
segmented.mod <- segmented(lin.mod, seg.Z= ~days)
summary(segmented.mod)
fit <- numeric(length(days)) * NA
fit[complete.cases(rowSums(cbind(charge, days)))] <- broken.line(segmented.mod)$fit

data1 <- data.frame(days = days, charge = charge, fit = fit)

ggplot(data1, aes(x = days, y = charge)) +
  geom_point() +
  geom_line(aes(x = days, y = fit), color = 'red') +
  ggtitle("Spending pattern over time for current customer, sequenceID 1996") +
  ylab("Transaction Charge in Euro") + xlab("Time span of Transactions measured in Days")
```