marketing_analysis

October 17, 2018

```
In [1]: install.packages("corrplot")
        library(readr)
        library(dplyr)
        library(corrplot)
        library(ggplot2)
```

```
Updating HTML index of packages in '.Library'
Making 'packages.html' ... done

Attaching package: dplyr

The following objects are masked from package:stats:

    filter, lag

The following objects are masked from package:base:

    intersect, setdiff, setequal, union

corrplot 0.84 loaded
```

```
In [2]: install.packages("rms")
        library(rms)
```

```
Updating HTML index of packages in '.Library'
Making 'packages.html' ... done
Loading required package: Hmisc
Loading required package: lattice
Loading required package: survival
Loading required package: Formula

Attaching package: Hmisc

The following objects are masked from package:dplyr:

    src, summarize
```

```
The following objects are masked from package:base:

    format.pval, units

Loading required package: SparseM

Attaching package: SparseM

The following object is masked from package:base:

    backsolve
```

```
In [3]: library(MASS)
```

```
Attaching package: MASS

The following object is masked from package:dplyr:

    select
```

```
In [4]: install.packages("descr")
        library(descr)
```

```
Updating HTML index of packages in '.Library'
Making 'packages.html' ... done
```

```
In [5]: install.packages("SDMTools")
        library(SDMTools)
```

```
Updating HTML index of packages in '.Library'
Making 'packages.html' ... done
```

```
In [6]: library(boot)
```

```
Attaching package: boot

The following object is masked from package:survival:

    aml

The following object is masked from package:lattice:
```

```
melanoma
```

1 Customer Lifetime Value CLV

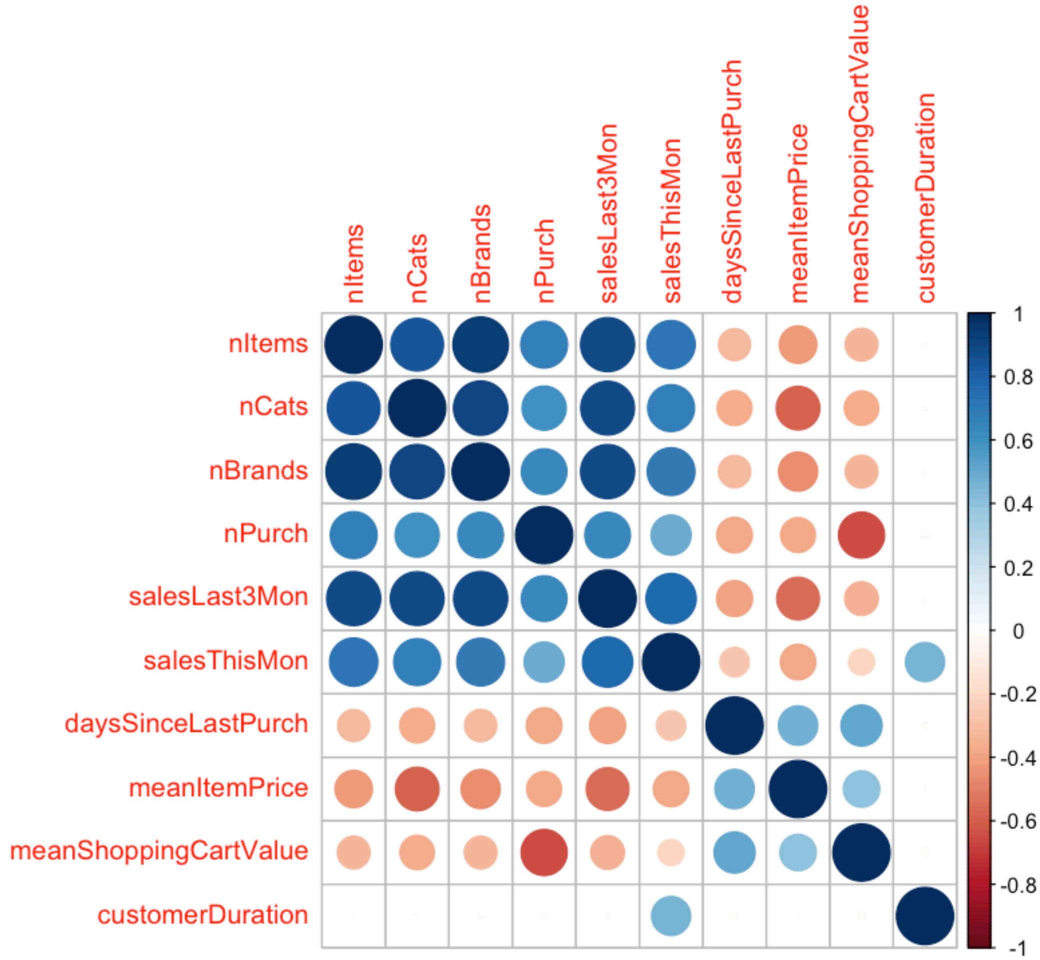
- predict future net-profit
- identity promising customers that drive margins
- prioritize customers according to future margins
- efficient organization of CRM
- no special segments

```
In [7]: sales_data <- read.csv("salesData.csv")
        str(sales_data)
```

```
'data.frame':       5122 obs. of  14 variables:
 $ id                  : int  1 2 3 4 5 6 7 8 9 10 ...
 $ nItems              : int  1469 1463 262 293 108 216 174 122 204 308 ...
 $ mostFreqStore       : Factor w/ 10 levels "Boston","Colorado Springs",..: 10 10 2 2 2 1 3
 $ mostFreqCat         : Factor w/ 10 levels "Alcohol","Baby",..: 1 1 10 3 4 1 8 10 3 1 ...
 $ nCats               : int  72 73 55 50 32 41 36 31 41 52 ...
 $ preferredBrand      : Factor w/ 10 levels "Akar","Alekto",..: 10 10 3 10 3 3 3 3 3 3 ...
 $ nBrands             : int  517 482 126 108 79 98 78 62 99 103 ...
 $ nPurch              : int  82 88 56 43 18 35 34 12 26 33 ...
 $ salesLast3Mon       : num  2742 2791 1530 1766 1180 ...
 $ salesThisMon        : num  1284 1243 683 730 553 ...
 $ daysSinceLastPurch  : int  1 1 1 1 12 2 2 4 14 1 ...
 $ meanItemPrice       : num  1.87 1.91 5.84 6.03 10.93 ...
 $ meanShoppingCartValue: num  33.4 31.7 27.3 41.1 65.6 ...
 $ customerDuration    : int  821 657 548 596 603 673 612 517 709 480 ...
```

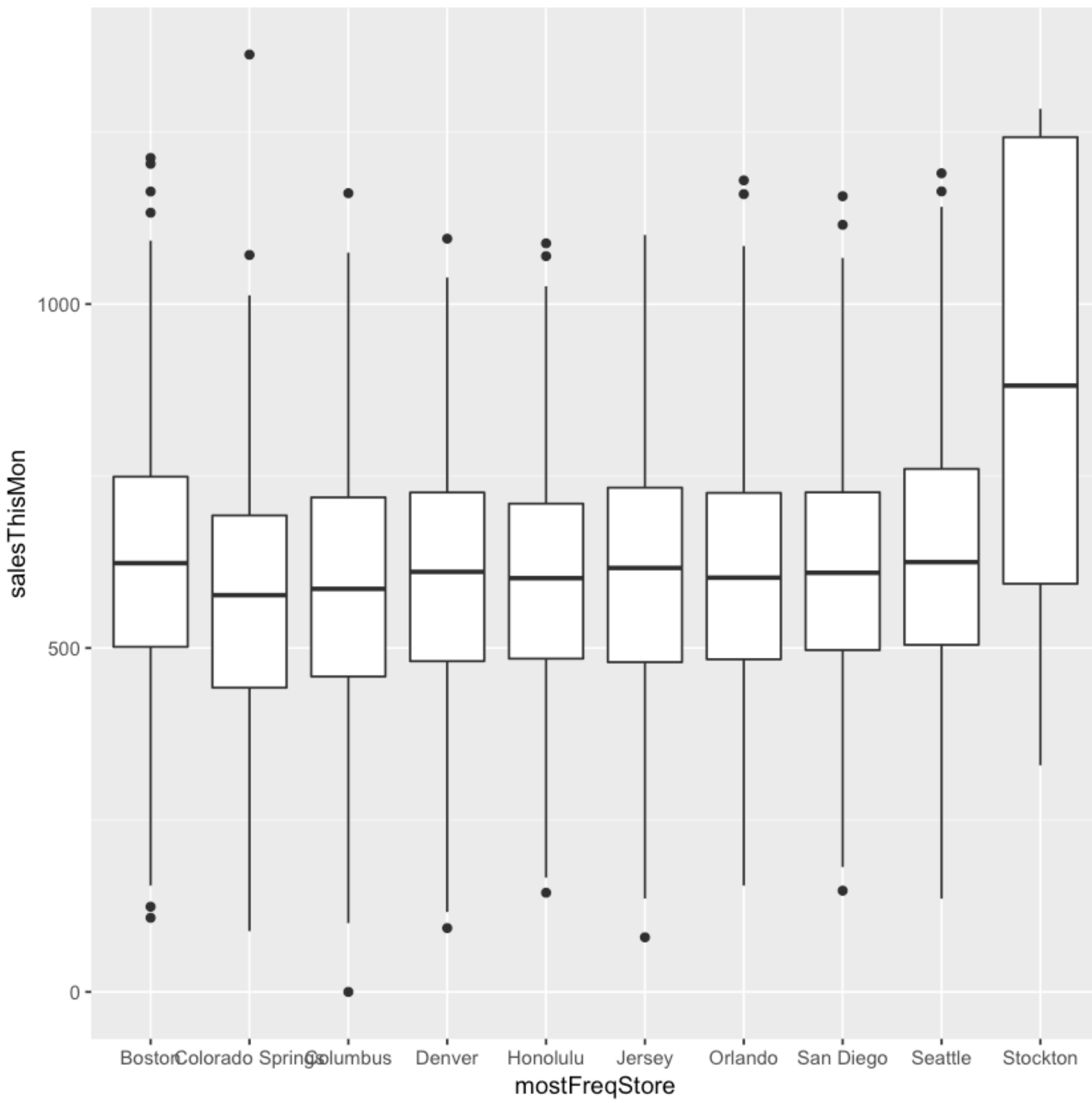
```
In [8]: # mostFreqStore: store person bought mostly from
        # mostFreCat: category person purchased mostly
        # nCats: number of different categories
        # preferredBrand: brand person purchased mostly
        # nBrands: number of different brands
```

```
In [13]: # Visualization of correlations
         sales_data %>% select_if(is.numeric) %>%
           dplyr:: select(-id) %>%
           cor() %>%
           corrplot()
```

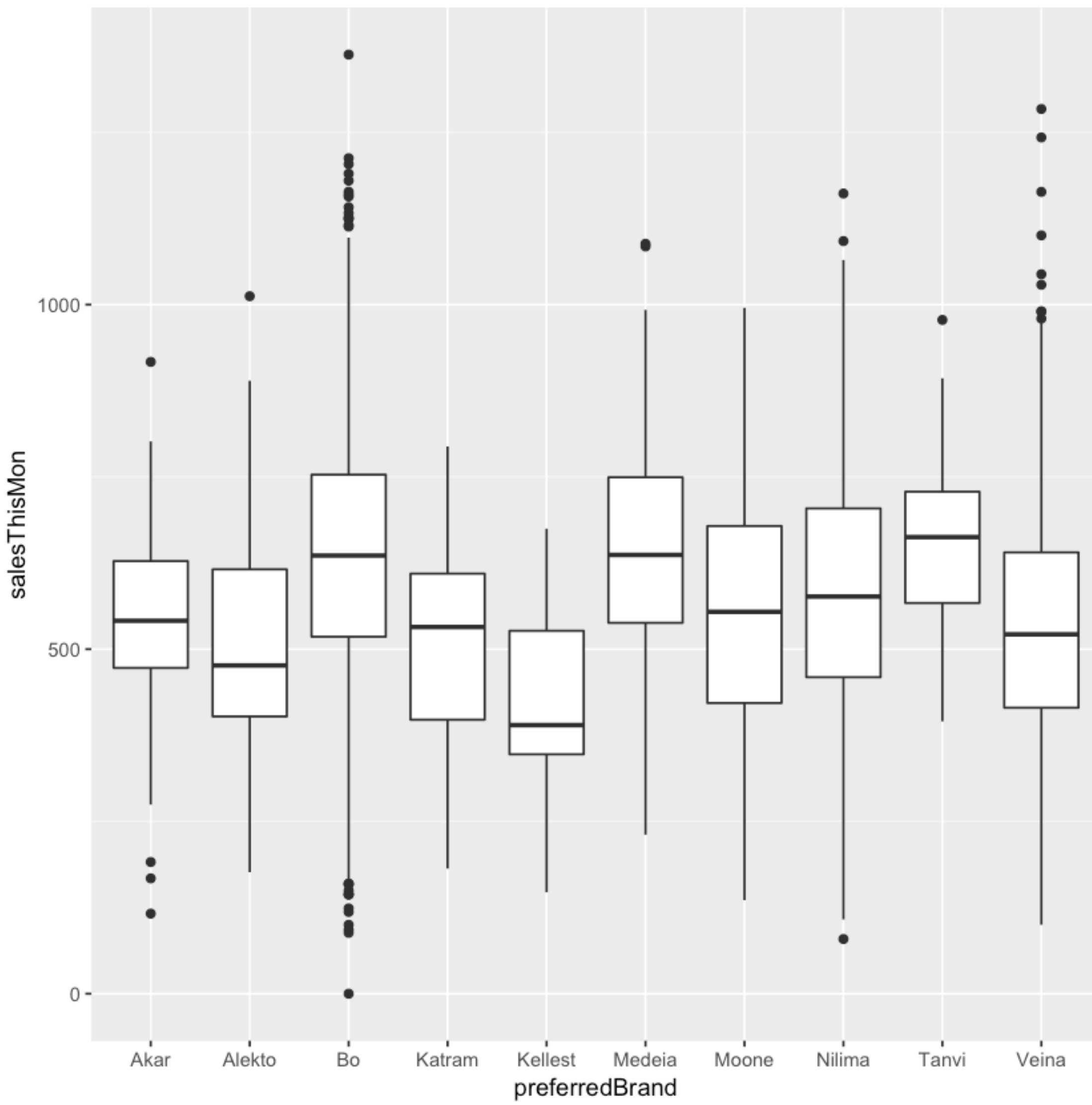


For salesThisMon, we can see nItems,nCats,nBrands,nPurch,salesLast3Mon,customerDuration have positive correlation, and daysSinceLastPurch, meanItemPrice, meanShoppingCartValue have negative correlation.

```
In [14]: # Frequent stores
         ggplot(sales_data) +
             geom_boxplot(aes(x = mostFreqStore, y = salesThisMon))
```



In [15]:
```
# Preferred brand
ggplot(sales_data) +
    geom_boxplot(aes(x = preferredBrand, y = salesThisMon))
```



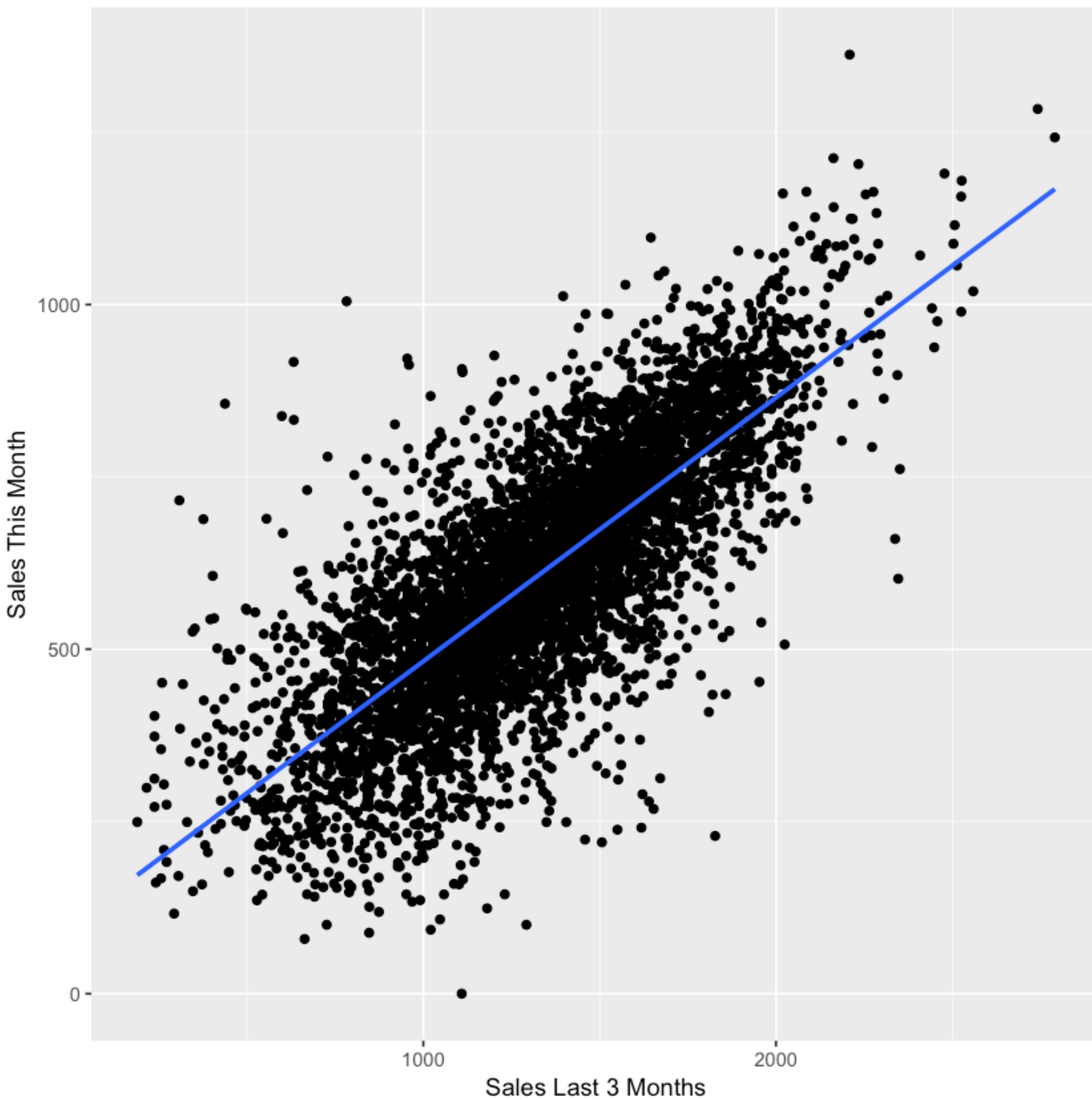

```
In [16]: # Model specification using lm
         salesSimpleModel <- lm(salesThisMon ~ salesLast3Mon,
                                 data = sales_data)
         summary(salesSimpleModel)
```

```
Call:
lm(formula = salesThisMon ~ salesLast3Mon, data = sales_data)

Residuals:
    Min      1Q  Median      3Q     Max
-570.18  -68.26    3.21   72.98  605.58
```

```
Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)   99.690501   6.083886   16.39   <2e-16 ***
salesLast3Mon  0.382696   0.004429   86.40   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 117.5 on 5120 degrees of freedom
Multiple R-squared:  0.5932,Adjusted R-squared:  0.5931
F-statistic:  7465 on 1 and 5120 DF,  p-value: < 2.2e-16
```

Since the regression coefficient is greater than 0, there exists a positive relationship between the explanatory variable salesLast3Mon and the dependent variable salesThisMon. It explains almost 60 percent of the variation in the sales of this month.

```
In [17]: ggplot(sales_data, aes(salesLast3Mon, salesThisMon)) +
           geom_point() +
           geom_smooth(method = lm, se = FALSE) +
           xlab("Sales Last 3 Months") +
           ylab("Sales This Month")
```



In [18]:
```
# Estimating the full model
salesMModel1 <- lm(salesThisMon ~ . - id,
                   data = sales_data)
summary(salesMModel1)
```

```
Call:
lm(formula = salesThisMon ~ . - id, data = sales_data)

Residuals:
    Min      1Q  Median      3Q     Max
-322.76  -50.76    0.78   50.90  398.79
```

```
Coefficients:
                                 Estimate Std. Error t value Pr(>|t|)    
(Intercept)                    -2.585e+02  1.762e+01 -14.673  < 2e-16 ***
nItems                          1.605e-01  2.709e-02   5.923 3.37e-09 ***
mostFreqStoreColorado Springs -7.167e+00  4.350e+00  -1.648 0.099503 .  
mostFreqStoreColumbus           9.579e-01  3.680e+00   0.260 0.794642    
mostFreqStoreDenver            -8.601e+00  5.130e+00  -1.676 0.093722 .  
mostFreqStoreHonolulu          -1.588e+01  4.916e+00  -3.231 0.001242 ** 
mostFreqStoreJersey            -2.169e+01  5.031e+00  -4.311 1.66e-05 ***
mostFreqStoreOrlando           -1.052e+01  4.492e+00  -2.342 0.019210 *  
mostFreqStoreSan Diego         -2.009e+01  5.717e+00  -3.514 0.000446 ***
mostFreqStoreSeattle           -9.784e+00  3.539e+00  -2.765 0.005716 ** 
mostFreqStoreStockton          -1.176e+02  3.580e+01  -3.286 0.001022 ** 
mostFreqCatBaby                -3.413e+00  3.513e+00  -0.972 0.331249    
mostFreqCatBakery              -1.025e+01  5.456e+00  -1.879 0.060339 .  
mostFreqCatBeverages            3.351e-01  7.008e+00   0.048 0.961867    
mostFreqCatClothes             -8.527e+00  6.213e+00  -1.372 0.170010    
mostFreqCatFresh food          -6.372e+00  7.245e+00  -0.880 0.379164    
mostFreqCatFrozen food         -8.084e+00  3.840e+00  -2.105 0.035332 *  
mostFreqCatPackaged food       -8.346e-01  4.356e+00  -0.192 0.848063    
mostFreqCatPets                 8.508e+00  7.242e+00   1.175 0.240102    
mostFreqCatShoes                3.298e+00  3.286e+00   1.004 0.315452    
nCats                          -7.917e-01  2.345e-01  -3.375 0.000742 ***
preferredBrandAlekto           -5.590e+00  1.649e+01  -0.339 0.734645    
preferredBrandBo               -2.505e+01  1.438e+01  -1.742 0.081516 .  
preferredBrandKatram           -6.264e+01  2.334e+01  -2.684 0.007295 ** 
preferredBrandKellest          -5.349e+01  2.214e+01  -2.416 0.015713 *  
preferredBrandMedeia           -2.161e+01  1.556e+01  -1.389 0.164967    
preferredBrandMoone            -4.166e+01  1.627e+01  -2.561 0.010453 *  
preferredBrandNilima           -2.888e+01  1.454e+01  -1.986 0.047040 *  
preferredBrandTanvi             3.135e+01  2.129e+01   1.472 0.141076    
preferredBrandVeina            -1.861e+01  1.451e+01  -1.282 0.199837    
nBrands                        -4.804e-02  8.468e-02  -0.567 0.570533    
nPurch                          4.758e-01  1.513e-01   3.145 0.001669 ** 
salesLast3Mon                   3.753e-01  8.599e-03  43.652  < 2e-16 ***
daysSinceLastPurch              1.794e-01  1.524e-01   1.177 0.239322    
meanItemPrice                   1.793e-01  9.289e-02   1.930 0.053680 .  
meanShoppingCartValue           2.596e-01  2.618e-02   9.918  < 2e-16 ***
customerDuration                5.713e-01  7.148e-03  79.927  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 77.34 on 5085 degrees of freedom
Multiple R-squared:  0.8249,Adjusted R-squared:  0.8237 
F-statistic: 665.6 on 36 and 5085 DF,  p-value: < 2.2e-16
```

```
In [19]: #the increase in the variance of an estimated coefficient due to multicollinearity
         # > 5 problem, >10 poor regression
         vif(salesMModel1)
```

nItems 11.7726003720926 **mostFreqStoreColorado Springs** 1.47809838533718 **mostFreqStoreColumbus** 1.74610149073175 **mostFreqStoreDenver** 1.28920276768667 **mostFreqStoreHonolulu** 1.33832962151996 **mostFreqStoreJersey** 1.31715824958004 **mostFreqStoreOrlando** 1.4013960039566 **mostFreqStoreSan Diego** 1.21992166861234 **mostFreqStoreSeattle** 1.79489094356777 **mostFreqStoreStockton** 1.07025015046598 **mostFreqCatBaby** 1.456920943197 **mostFreqCatBakery** 1.24603518193696 **mostFreqCatBeverages** 1.07900674526415 **mostFreqCatClothes** 1.15684054610955 **mostFreqCatFresh food** 1.06998659948896 **mostFreqCatFrozen food** 1.29635832928504 **mostFreqCatPackaged food** 1.26800036636315 **mostFreqCatPets** 1.07748772779324 **mostFreqCatShoes** 1.41780662860384 **nCats** 8.40207292692662 **preferredBrandAlekto** 3.84417571543037 **preferredBrandBo** 41.0759302801738 **preferredBrandKatram** 1.6329780973379 **preferredBrandKellest** 1.7135101487428 **preferredBrandMedeia** 6.12038383652048 **preferredBrandMoone** 4.5915695339592 **preferredBrandNilima** 22.7143759343577 **preferredBrandTanvi** 1.88577658902921 **preferredBrandVeina** 20.7391135467044 **nBrands** 14.1508681292858 **nPurch** 3.08395248721893 **salesLast3Mon** 8.69766334202653 **daysSinceLastPurch** 1.58505716973954 **meanItemPrice** 1.98766522983 **meanShoppingCartValue** 2.24757929753948 **customerDuration** 1.00466438582497

```
In [20]: # Estimating new model by removing information on brand
         salesMModel2 <- lm(salesThisMon ~ . - id - preferredBrand -nBrands,
                            data = sales_data)

         # Checking variance inflation factors
         vif(salesMModel2)
         AIC(salesMModel2)
         summary(salesMModel2)
```

nItems 6.98745645936931 **mostFreqStoreColorado Springs** 1.47050847482464 **mostFreqStoreColumbus** 1.7377898432521 **mostFreqStoreDenver** 1.28322165547675 **mostFreqStoreHonolulu** 1.33545670344308 **mostFreqStoreJersey** 1.29988900889812 **mostFreqStoreOrlando** 1.3983175098929 **mostFreqStoreSan Diego** 1.21386453248199 **mostFreqStoreSeattle** 1.78877681084754 **mostFreqStoreStockton** 1.05206479237803 **mostFreqCatBaby** 1.41275506193802 **mostFreqCatBakery** 1.2369388370151 **mostFreqCatBeverages** 1.07790655637251 **mostFreqCatClothes** 1.10505423608417 **mostFreqCatFresh food** 1.06708875737944 **mostFreqCatFrozen food** 1.27095256165401 **mostFreqCatPackaged food** 1.23516450097361 **mostFreqCatPets** 1.07227791700981 **mostFreqCatShoes** 1.38486149146926 **nCats** 5.8134943280382 **nPurch** 3.06904648703954 **salesLast3Mon** 8.41252036727487 **daysSinceLastPurch** 1.57942647277196 **meanItemPrice** 1.92549399943984 **meanShoppingCartValue** 2.23841003275004 **customerDuration** 1.00298102097394

59136.6223868772

```
Call:
lm(formula = salesThisMon ~ . - id - preferredBrand - nBrands,
```

```
    data = sales_data)

Residuals:
    Min      1Q  Median      3Q     Max
-322.66  -51.26    0.60   51.28  399.10

Coefficients:
                                Estimate Std. Error t value Pr(>|t|)
(Intercept)                   -2.828e+02  1.007e+01 -28.079  < 2e-16 ***
nItems                         1.470e-01  2.093e-02   7.023 2.45e-12 ***
mostFreqStoreColorado Springs -7.829e+00  4.351e+00  -1.799 0.072047 .
mostFreqStoreColumbus          5.960e-01  3.682e+00   0.162 0.871391
mostFreqStoreDenver           -9.721e+00  5.133e+00  -1.894 0.058305 .
mostFreqStoreHonolulu         -1.604e+01  4.925e+00  -3.257 0.001134 **
mostFreqStoreJersey           -2.215e+01  5.011e+00  -4.420 1.01e-05 ***
mostFreqStoreOrlando          -1.104e+01  4.500e+00  -2.454 0.014154 *
mostFreqStoreSan Diego        -1.985e+01  5.718e+00  -3.472 0.000521 ***
mostFreqStoreSeattle          -9.573e+00  3.542e+00  -2.702 0.006906 **
mostFreqStoreStockton         -1.129e+02  3.559e+01  -3.171 0.001530 **
mostFreqCatBaby               -3.496e+00  3.469e+00  -1.008 0.313594
mostFreqCatBakery             -9.908e+00  5.451e+00  -1.818 0.069188 .
mostFreqCatBeverages           9.253e-02  7.024e+00   0.013 0.989489
mostFreqCatClothes            -3.828e+00  6.090e+00  -0.629 0.529674
mostFreqCatFresh food         -5.935e+00  7.255e+00  -0.818 0.413368
mostFreqCatFrozen food        -7.196e+00  3.813e+00  -1.887 0.059179 .
mostFreqCatPackaged food      -1.387e+00  4.311e+00  -0.322 0.747746
mostFreqCatPets                9.073e+00  7.245e+00   1.252 0.210467
mostFreqCatShoes               2.649e+00  3.256e+00   0.814 0.415917
nCats                         -9.585e-01  1.956e-01  -4.900 9.90e-07 ***
nPurch                         5.092e-01  1.513e-01   3.364 0.000773 ***
salesLast3Mon                  3.782e-01  8.480e-03  44.604  < 2e-16 ***
daysSinceLastPurch             1.712e-01  1.526e-01   1.122 0.262022
meanItemPrice                  2.253e-01  9.168e-02   2.457 0.014034 *
meanShoppingCartValue          2.584e-01  2.620e-02   9.861  < 2e-16 ***
customerDuration               5.708e-01  7.162e-03  79.707  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 77.56 on 5095 degrees of freedom
Multiple R-squared:  0.8236,Adjusted R-squared:  0.8227
F-statistic: 914.9 on 26 and 5095 DF,  p-value: < 2.2e-16
```

Since none of the variance inflation factors is greater than 10 we can certainly accept the second model. An expected sales in this month will be roughly given that every variables are 0 The effect of the mean item price on the sales this month is statistically significant. A one-unit increase in the mean item price leads to a 0.23 Euro increase in the sales of this month.

```
In [21]: sales_2to4=read.csv("salesDataMon2To4.csv")
         # predicting sales
         predSales5 <- predict(salesMModel2, newdata = sales_2to4)

         # calculating mean of future sales
         mean(predSales5, na.rm = FALSE)
```

625.143833363781

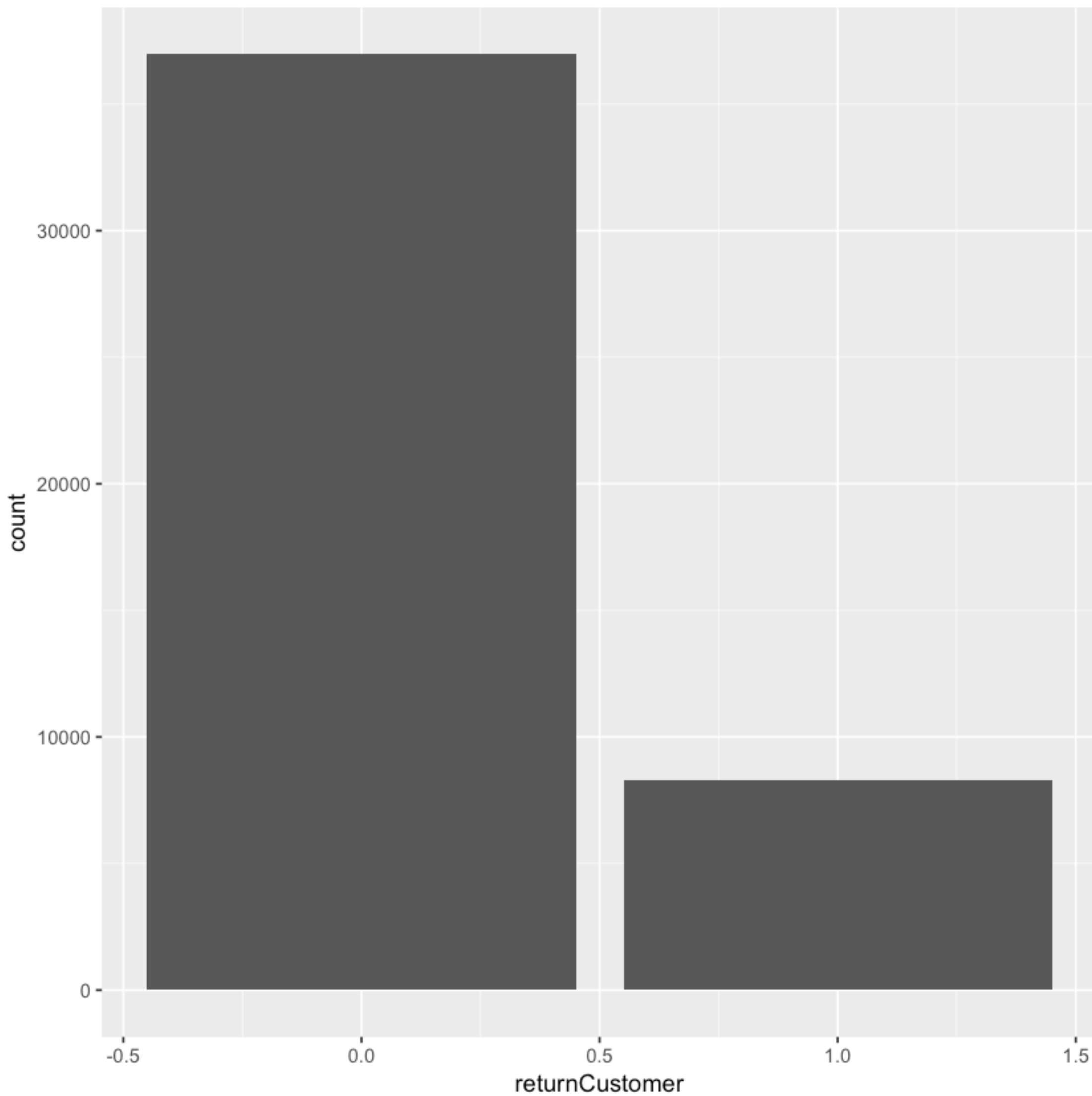
2 Churn Prevention

```
In [22]: churn_data = read.csv("churn_data.csv")
         str(churn_data)
```

```
'data.frame':        45236 obs. of  21 variables:
 $ ID               : int  1 3 5 7 8 9 10 11 12 13 ...
 $ orderDate        : Factor w/ 354 levels "1/1/15","1/10/15",..: 108 326 317 4 343 271 310 3
 $ title            : Factor w/ 4 levels "Company","Mr",..: 2 2 2 2 2 2 2 3 1 2 ...
 $ newsletter       : int  0 0 0 0 0 0 1 0 1 0 ...
 $ websiteDesign    : int  2 1 1 3 3 1 1 2 2 2 ...
 $ paymentMethod    : Factor w/ 4 levels "Cash","Credit Card",..: 3 4 1 1 1 2 4 4 3 1 ...
 $ couponDiscount   : int  1 0 0 0 0 1 0 0 1 0 ...
 $ purchaseValue    : int  2 1 4 4 4 4 4 5 1 3 ...
 $ giftwrapping     : int  0 0 0 0 0 0 0 0 0 0 ...
 $ throughAffiliate : int  1 0 0 1 1 0 0 1 1 1 ...
 $ shippingFees     : int  0 1 0 0 0 0 0 0 1 0 ...
 $ dvd              : int  0 0 0 1 2 4 0 3 0 0 ...
 $ blueray          : int  1 0 0 0 0 0 0 0 1 0 ...
 $ vinyl            : int  0 0 1 0 0 0 0 0 0 0 ...
 $ videogame        : int  0 0 0 0 0 0 0 0 0 0 ...
 $ videogameDownload: int  0 0 0 0 0 0 0 0 0 0 ...
 $ tvEquiment       : int  0 0 0 0 0 0 0 0 0 0 ...
 $ prodOthers       : int  0 0 0 0 0 0 0 0 0 0 ...
 $ prodRemitted     : int  0 0 0 0 0 0 0 0 0 0 ...
 $ prodSecondHand   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ returnCustomer   : int  0 0 0 0 0 0 0 0 0 0 ...
```

```
In [23]: # Analyze the balancedness of dependent variable
         ggplot(churn_data, aes(x = returnCustomer)) +
             geom_histogram(stat = "count")
```

```
Warning message:
Ignoring unknown parameters: binwidth, bins, pad
```




```
In [24]: logitModelFull <- glm(returnCustomer ~ title + newsletter + websiteDesign +
                                paymentMethod + couponDiscount + purchaseValue +
                                giftwrapping + throughAffiliate + shippingFees +
                                dvd + blueray + vinyl + videogame + videogameDownload +
                                tvEquiment + prodOthers + prodRemitted + prodSecondHand,
                                family = binomial, churn_data)
         summary(logitModelFull)
```

```
Call:
glm(formula = returnCustomer ~ title + newsletter + websiteDesign +
    paymentMethod + couponDiscount + purchaseValue + giftwrapping +
    throughAffiliate + shippingFees + dvd + blueray + vinyl +
```

```
    videogame + videogameDownload + tvEquiment + prodOthers +
    prodRemitted + prodSecondHand, family = binomial, data = churn_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5166  -0.6599  -0.5682  -0.4606   2.3674

Coefficients:
                             Estimate Std. Error z value Pr(>|z|)
(Intercept)                  -1.81151    0.07894 -22.949  < 2e-16 ***
titleMr                       0.21256    0.05286   4.021 5.79e-05 ***
titleMrs                      0.24188    0.05445   4.442 8.90e-06 ***
titleOthers                   0.78040    0.05766  13.535  < 2e-16 ***
newsletter                    0.51996    0.03028  17.169  < 2e-16 ***
websiteDesign                 0.10515    0.03430   3.066  0.00217 **
paymentMethodCredit Card     -0.25242    0.04834  -5.221 1.78e-07 ***
paymentMethodCurrent Account -0.27071    0.04141  -6.537 6.28e-11 ***
paymentMethodInvoice         -0.24603    0.03608  -6.818 9.21e-12 ***
couponDiscount               -0.22734    0.04174  -5.447 5.12e-08 ***
purchaseValue                -0.02693    0.01277  -2.108  0.03505 *
giftwrapping                  0.01193    0.19016   0.063  0.94998
throughAffiliate             -0.01585    0.05893  -0.269  0.78791
shippingFees                 -0.46821    0.04480 -10.451  < 2e-16 ***
dvd                           0.07159    0.01426   5.020 5.16e-07 ***
blueray                       0.12250    0.01761   6.954 3.54e-12 ***
vinyl                         0.05626    0.02276   2.472  0.01344 *
videogame                    -0.25059    0.11139  -2.250  0.02447 *
videogameDownload             0.27797    0.05257   5.288 1.24e-07 ***
tvEquiment                   -0.51552    1.08139  -0.477  0.63356
prodOthers                   -0.05989    0.07749  -0.773  0.43960
prodRemitted                  0.89450    0.07617  11.744  < 2e-16 ***
prodSecondHand                0.16179    0.09934   1.629  0.10339
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 43038  on 45235  degrees of freedom
Residual deviance: 41727  on 45213  degrees of freedom
AIC: 41773

Number of Fisher Scoring iterations: 4
```

```
In [25]: coefsExp <- coef(logitModelFull) %>% exp() %>% round(2)
         coefsExp
```

(Intercept) 0.16 **titleMr** 1.24 **titleMrs** 1.27 **titleOthers** 2.18 **newsletter** 1.68 **websiteDesign** 1.11

paymentMethodCredit Card 0.78 **paymentMethodCurrent Account** 0.76 **paymentMethodInvoice** 0.78 **couponDiscount** 0.8 **purchaseValue** 0.97 **giftwrapping** 1.01 **throughAffiliate** 0.98 **shippingFees** 0.63 **dvd** 1.07 **blueray** 1.13 **vinyl** 1.06 **videogame** 0.78 **videogameDownload** 1.32 **tvEquiment** 0.6 **prodOthers** 0.94 **prodRemitted** 2.45 **prodSecondHand** 1.18

```
In [26]: logitModelNew <- stepAIC(logitModelFull, trace = 0)
         summary(logitModelNew)
```

```
Call:
glm(formula = returnCustomer ~ title + newsletter + websiteDesign +
    paymentMethod + couponDiscount + purchaseValue + shippingFees +
    dvd + blueray + vinyl + videogame + videogameDownload + prodRemitted +
    prodSecondHand, family = binomial, data = churn_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5147  -0.6603  -0.5679  -0.4606   2.3692

Coefficients:
                             Estimate Std. Error z value Pr(>|z|)
(Intercept)                  -1.80497    0.07512 -24.027  < 2e-16 ***
titleMr                       0.21140    0.05283   4.001 6.30e-05 ***
titleMrs                      0.24169    0.05445   4.439 9.04e-06 ***
titleOthers                   0.77973    0.05765  13.526  < 2e-16 ***
newsletter                    0.51938    0.03027  17.157  < 2e-16 ***
websiteDesign                 0.09725    0.01624   5.989 2.11e-09 ***
paymentMethodCredit Card     -0.25426    0.04820  -5.275 1.33e-07 ***
paymentMethodCurrent Account -0.26979    0.04136  -6.524 6.87e-11 ***
paymentMethodInvoice         -0.24497    0.03602  -6.801 1.04e-11 ***
couponDiscount               -0.23073    0.04058  -5.686 1.30e-08 ***
purchaseValue                -0.02761    0.01271  -2.173   0.0298 *
shippingFees                 -0.46860    0.04478 -10.465  < 2e-16 ***
dvd                           0.07241    0.01419   5.102 3.36e-07 ***
blueray                       0.12283    0.01756   6.997 2.62e-12 ***
vinyl                         0.05722    0.02270   2.521   0.0117 *
videogame                    -0.24845    0.11136  -2.231   0.0257 *
videogameDownload             0.28187    0.05200   5.421 5.94e-08 ***
prodRemitted                  0.89434    0.07617  11.742  < 2e-16 ***
prodSecondHand                0.16387    0.09924   1.651   0.0987 .
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 43038  on 45235  degrees of freedom
Residual deviance: 41728  on 45217  degrees of freedom
```

```
AIC: 41766

Number of Fisher Scoring iterations: 4
```

giftwrapping, throughAffiliate, tvEquiment, prodOthers are removed for better model (lower AIC)

```
In [27]: # Save the formula of the new model (it will be needed for the out-of-sample part)
         formulaLogit <- as.formula(summary(logitModelNew)$call)
         formulaLogit
```

```
returnCustomer ~ title + newsletter + websiteDesign + paymentMethod +
    couponDiscount + purchaseValue + shippingFees + dvd + blueray +
    vinyl + videogame + videogameDownload + prodRemitted + prodSecondHand
```

```
In [28]: LogRegR2(logitModelNew)
```

```
Chi2                 1310.648
Df                   18
Sig.                 0
Cox and Snell Index  0.02855785
Nagelkerke Index     0.04652587
McFadden's R2        0.03045313
```

```
In [29]: churn_data$predNew <- predict(logitModelNew, type = "response", na.action = na.exclude

In [30]: confMatrixNew <- confusion.matrix(churn_data$returnCustomer,
                                  churn_data$predNew, threshold = 0.5)
         confMatrixNew
```

```
    obs
pred     0    1
   0 36921 8243
   1    43   29
attr(,"class")
[1] "confusion.matrix"
```

```
In [31]: # Calculate the accuracy for the full Model
         accuracy <- sum(diag(confMatrixNew)) / sum(confMatrixNew)
         accuracy
```

0.816827305685737

```
In [32]: # Prepare data frame with threshold values and empty payoff column
         payoffMatrix <- data.frame(threshold = seq(from = 0.1, to = 0.5, by = 0.1),
                                    payoff = NA)
         for(i in 1:length(payoffMatrix$threshold)) {
           # Calculate confusion matrix with varying threshold
           confMatrix <- confusion.matrix(churn_data$returnCustomer,
                         churn_data$predNew,
                         threshold = payoffMatrix$threshold[i])
           # Calculate payoff and save it to the corresponding row
           payoffMatrix$payoff[i] <- confMatrix[1,1]*250 + confMatrix[1,2]*(-1000)
         }
         payoffMatrix
```

threshold	payoff
0.1	453750
0.2	2163500
0.3	1470750
0.4	1087000
0.5	987250

optimal threshold 0.2 maximizes payoff

```
In [33]: # Generating random index for training and test set
         # set.seed ensures reproducibility of random components
         set.seed(534381)

         churn_data$isTrain <- rbinom(nrow(churn_data), 1, 0.66)
         train <- subset(churn_data, churn_data$isTrain == 1)
         test <- subset(churn_data, churn_data$isTrain == 0)

         # Modeling logitTrainNew
         logitTrainNew <- glm( returnCustomer ~ title + newsletter + websiteDesign +
                 paymentMethod + couponDiscount + purchaseValue + throughAffiliate +
                 shippingFees + dvd + blueray + vinyl + videogameDownload +
                 prodOthers + prodRemitted, family = binomial, data = train)
         # Out-of-sample prediction for logitTrainNew
         test$predNew <- predict(logitTrainNew, type = "response", newdata = test)

         #calculating the confusion matrix
         confMatrixTest <- confusion.matrix(test$returnCustomer, test$predNew,
                          threshold = 0.2)
         confMatrixTest
         #calculating the accuracy
         accuracyTest <- sum(diag(confMatrixTest)) / sum(confMatrixTest)
         accuracyTest
```

```
    obs
pred    0    1
   0 9025 1501
```

```
   1 3630 1298
attr(,"class")
[1] "confusion.matrix"
```

0.667982399378802

In [34]:
```
# Accuracy function with threshold = 0.2
Acc <- function(r, pi = 0) {
  cm <- confusion.matrix(r, pi, threshold = 0.2)
  acc <- sum(diag(cm)) / sum(cm)
  return(acc)
}

# Accuracy
set.seed(534381)
cv.glm(churn_data, logitModelNew, cost = Acc, K = 6)$delta[1]
```

0.664957113803166

3 Survival Analysis

Survival analysis is suited for situations where for some observations an event has not yet happened, but may happen at some point in time.

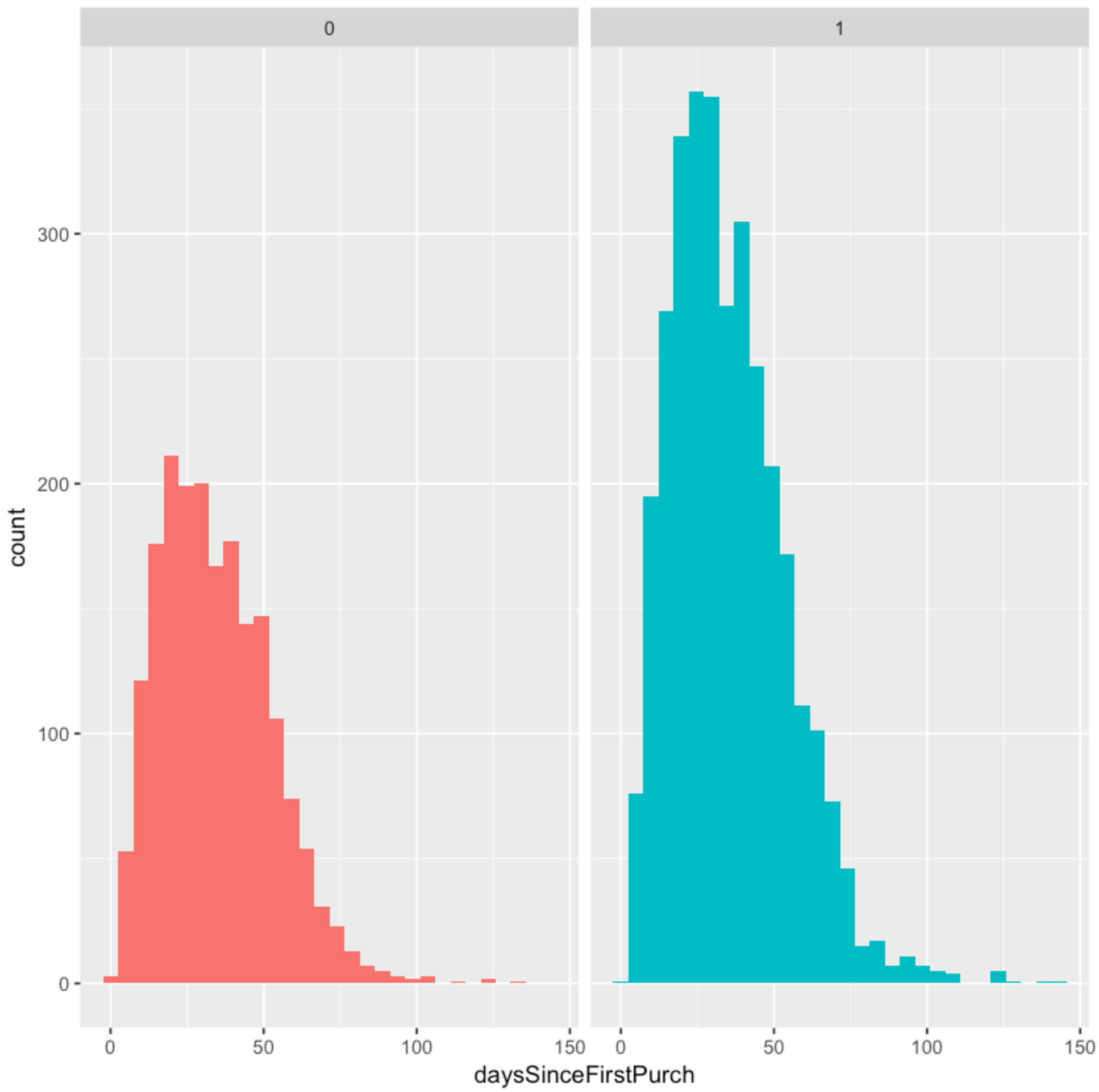
Survival function gives the probability that a customer will not churn in the period leading up to the time point t.

In [35]:
```
dataNextOrder = read.csv("survivalData.csv")
str(dataNextOrder)
```

```
'data.frame':        5122 obs. of  6 variables:
 $ daysSinceFirstPurch: int  37 63 48 17 53 11 22 16 74 44 ...
 $ shoppingCartValue  : num  33.4 31.7 27.3 41.1 65.6 ...
 $ gender             : Factor w/ 2 levels "female","male": 2 2 1 2 1 1 1 1 1 1 ...
 $ voucher            : int  0 1 0 0 0 0 0 1 0 0 ...
 $ returned           : int  0 0 0 0 0 0 0 0 0 0 ...
 $ boughtAgain        : int  0 1 0 1 0 1 1 1 0 1 ...
```

In [36]:
```
# Plot a histogram
ggplot(dataNextOrder) +
  geom_histogram(aes(x = daysSinceFirstPurch,
                     fill = factor(boughtAgain))) + # Different colours
  facet_grid( ~ boughtAgain) + # Separate plots for boughtAgain = 1 vs. 0
  theme(legend.position = "none") # Don't show legend
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

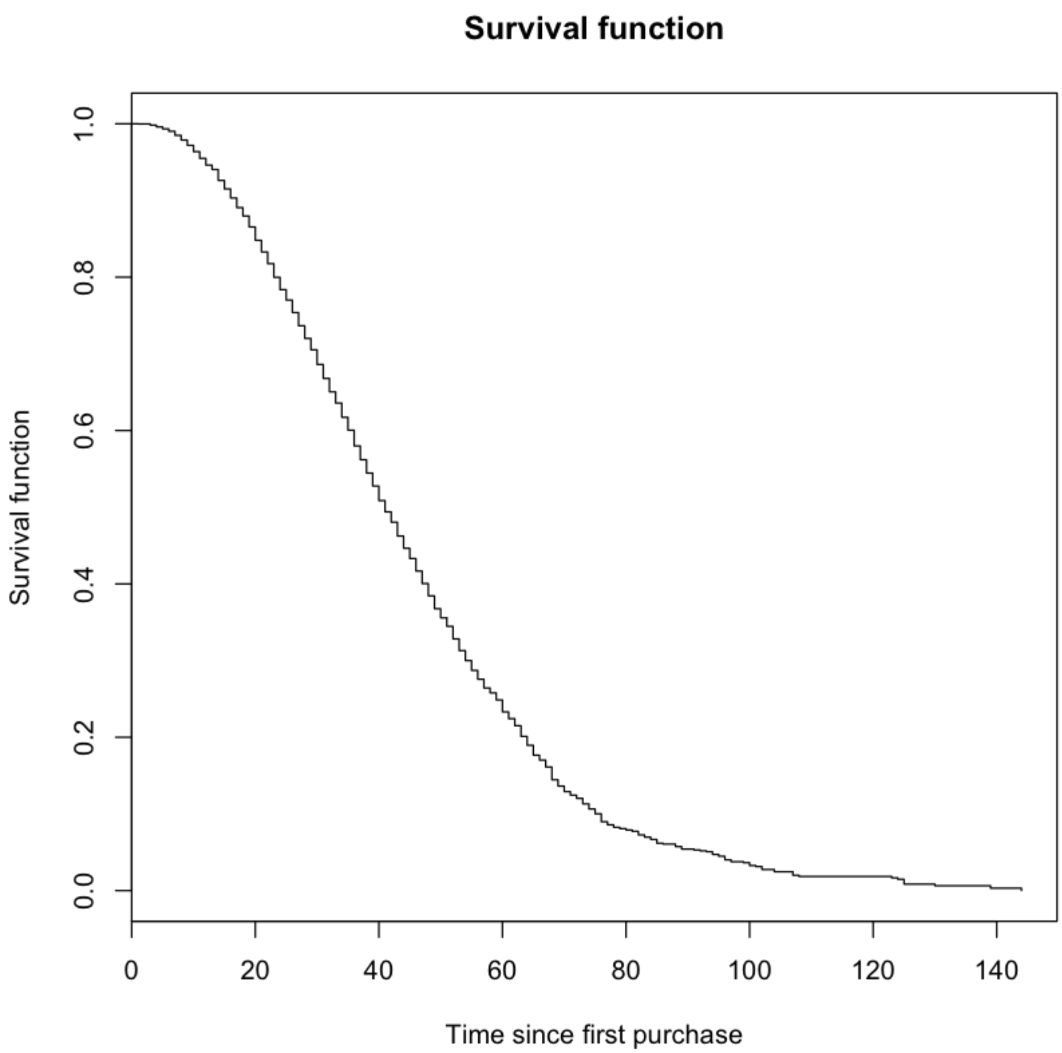



```
In [37]: # Create survival object
         survObj <- Surv(dataNextOrder$daysSinceFirstPurch, dataNextOrder$boughtAgain)

         # Look at structure
         str(survObj)
```

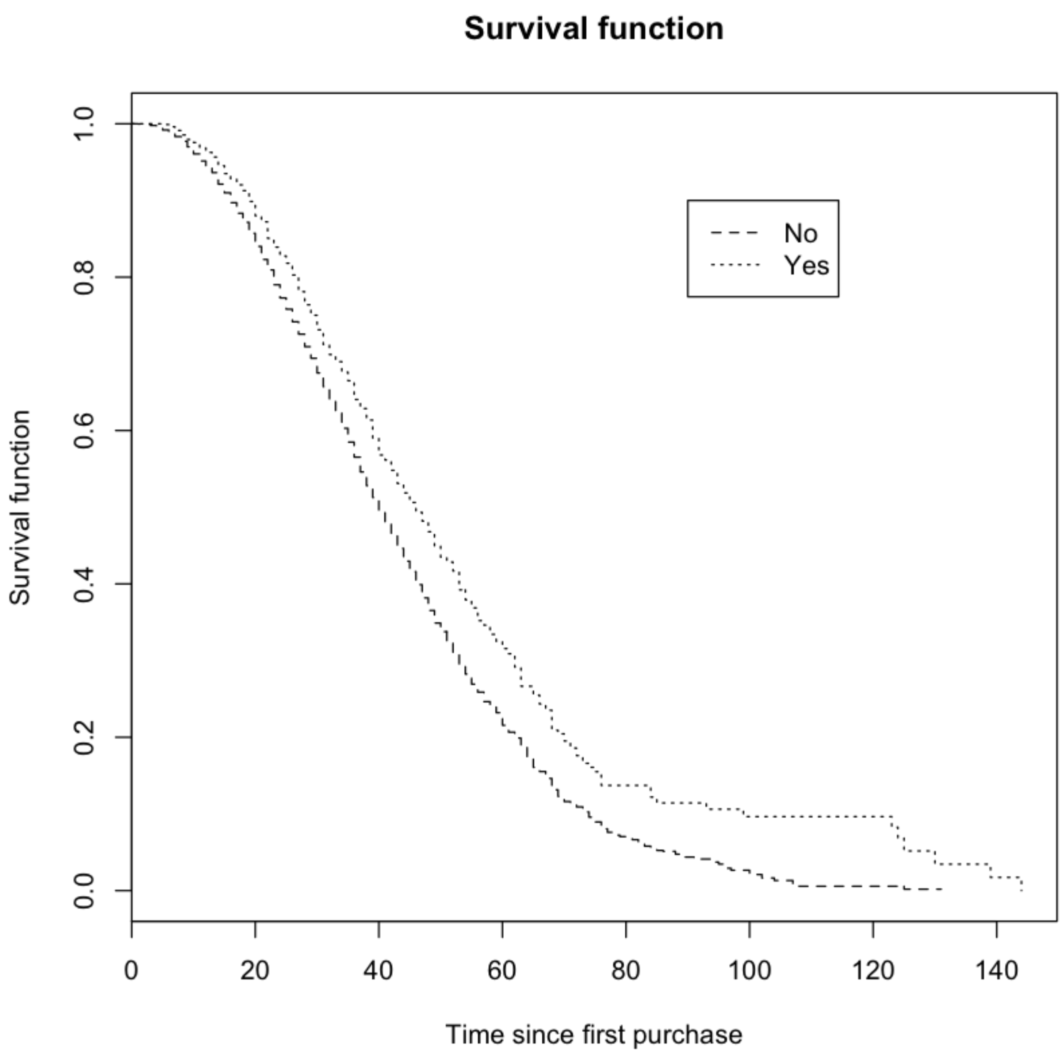
```
 'Surv' num [1:5122, 1:2]  37+  63   48+  17   53+  11   22   16   74+  44  ...
 - attr(*, "dimnames")=List of 2
  ..$ : NULL
  ..$ : chr [1:2] "time" "status"
 - attr(*, "type")= chr "right"
```

```
In [38]: # Compute and print fit
         fitKMSimple <- survfit(survObj ~ 1,type = "kaplan-meier") # independent of any covari
         print(fitKMSimple)

         # Plot fit
         plot(fitKMSimple, conf.int = FALSE,
              xlab = "Time since first purchase", ylab = "Survival function", main = "Survival
```

```
Call: survfit(formula = survObj ~ 1, type = "kaplan-meier")

      n  events  median 0.95LCL 0.95UCL
   5122    3199      41      40      42
```



In [39]:
```
# Compute fit with covariate
fitKMCov <- survfit(survObj ~ voucher, data = dataNextOrder)

# Plot fit with covariate and add labels
plot(fitKMCov, lty = 2:3,
     xlab = "Time since first purchase", ylab = "Survival function", main = "Survival
legend(90, .9, c("No", "Yes"), lty = 2:3)
```



Among 5122 people, 3199 customers have purchased again. The median 41 shows that 50% customers will not place the second order within 41 days.

Customers using a voucher seem to take longer to place their second order.

In [40]:
```
# Determine distributions of predictor variables
units(dataNextOrder$daysSinceFirstPurch) <- "Day"
```

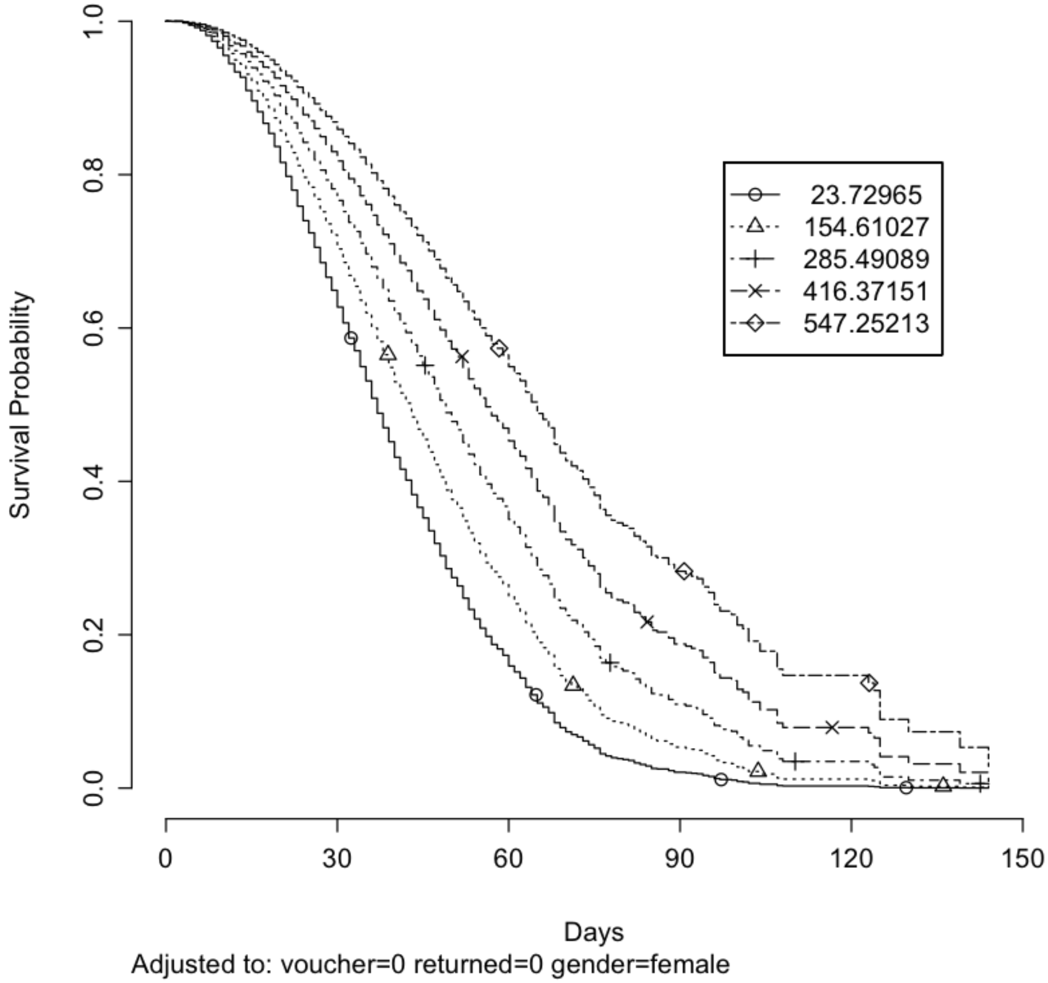
```
dd <- datadist(dataNextOrder)
options(datadist = "dd")

# Compute Cox PH Model and print results
fitCPH1 <- cph(Surv(daysSinceFirstPurch, boughtAgain) ~ shoppingCartValue + voucher +
              data = dataNextOrder,
              x = TRUE, y = TRUE, surv = TRUE)
print(fitCPH1)
```

```
Cox Proportional Hazards Model

 cph(formula = Surv(daysSinceFirstPurch, boughtAgain) ~ shoppingCartValue +
     voucher + returned + gender, data = dataNextOrder, x = TRUE,
     y = TRUE, surv = TRUE)

                      Model Tests       Discrimination
                                           Indexes
 Obs        5122    LR chi2     155.68    R2       0.030
 Events     3199    d.f.             4    Dxy      0.116
 Center -0.2808     Pr(> chi2) 0.0000     g        0.238
                    Score chi2 140.57     gr       1.269
                    Pr(> chi2) 0.0000

                   Coef    S.E.   Wald Z Pr(>|Z|)
 shoppingCartValue -0.0021 0.0003 -7.56  <0.0001
 voucher           -0.2945 0.0480 -6.14  <0.0001
 returned          -0.3145 0.0495 -6.36  <0.0001
 gender=male        0.1080 0.0363  2.97  0.0029
```

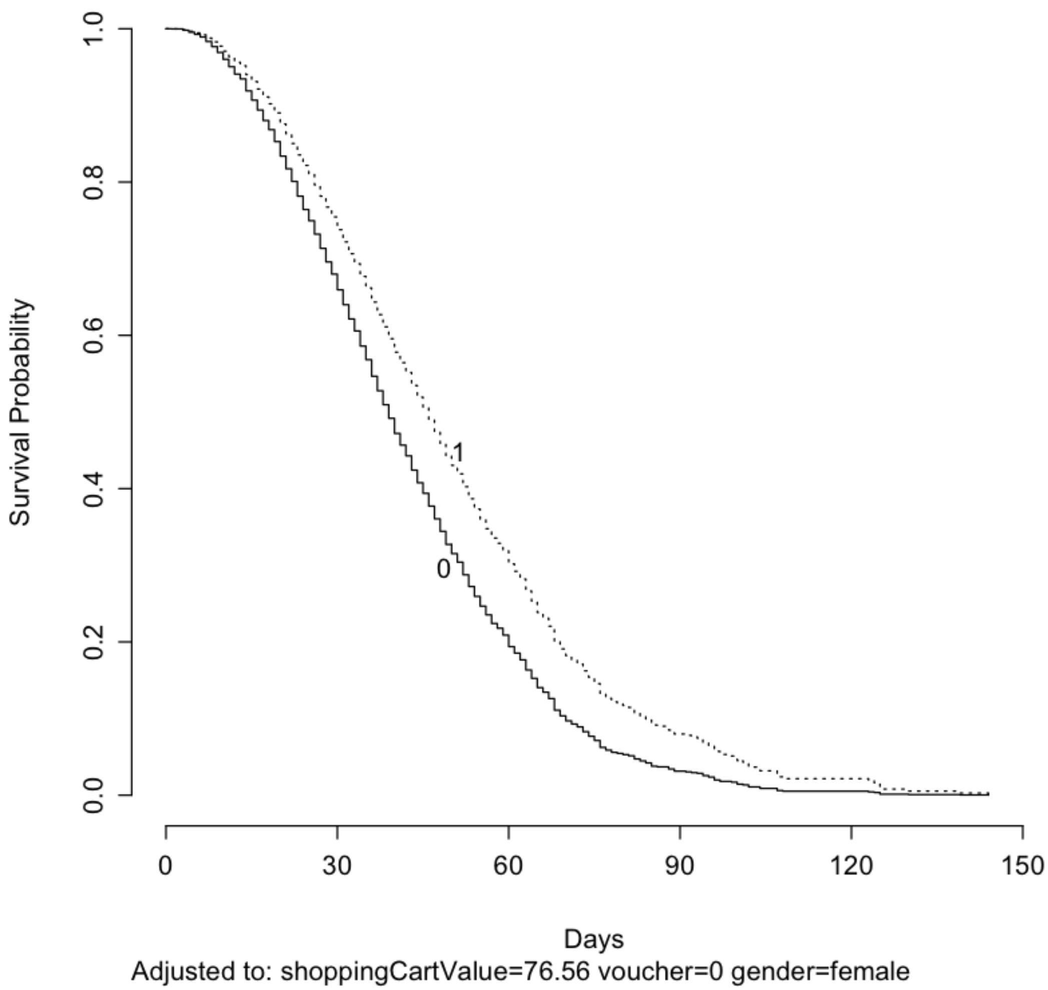
```
In [41]: # Interpret coefficients
         exp(fitCPH1$coefficients)
```

shoppingCartValue 0.997860099822716 **voucher** 0.744936184973824 **returned** 0.730166724008206 **gender=male** 1.11408908245914

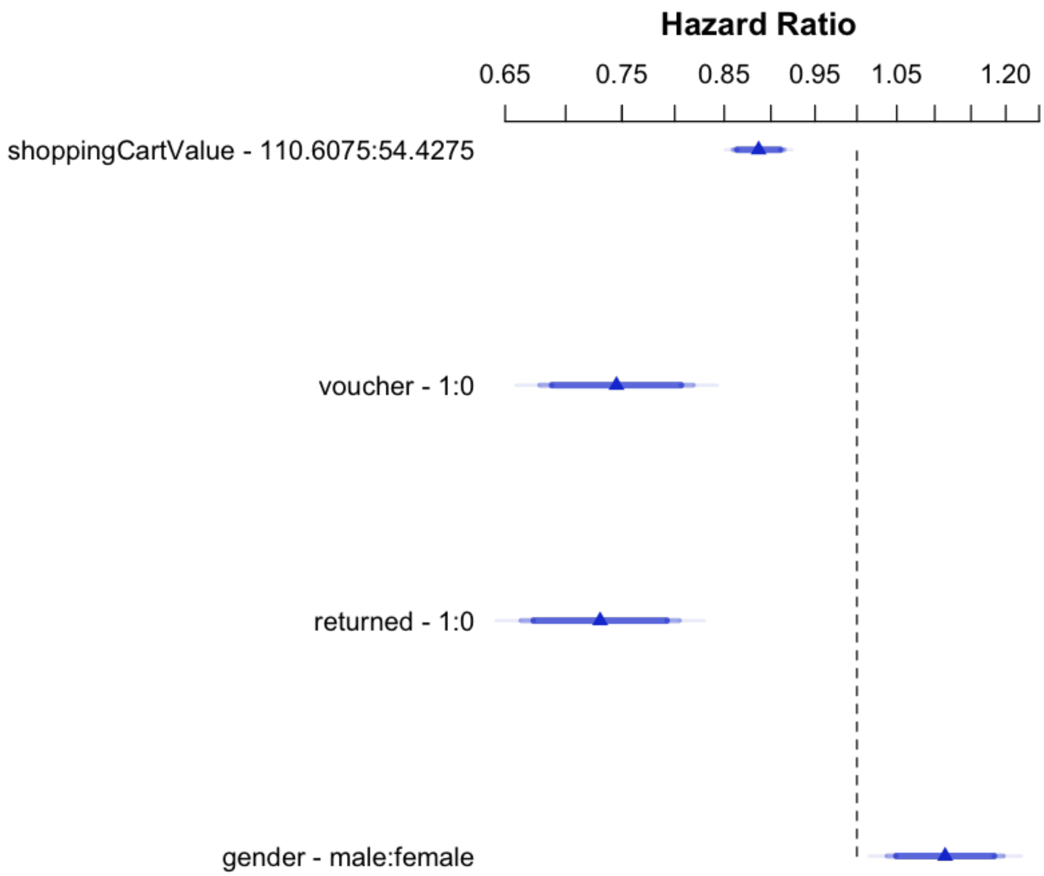
```
In [42]: #survival probability plot
         survplot(fitCPH1, shoppingCartValue, label.curves = list(keys = 1:5))
         survplot(fitCPH1, returned)
```



1.0
0.8
0.6
0.4
0.2
0.0
Survival Probability
23.72965
154.61027
285.49089
416.37151
547.25213
0
30
60
90
120
150
Days
Adjusted to: voucher=0 returned=0 gender=female

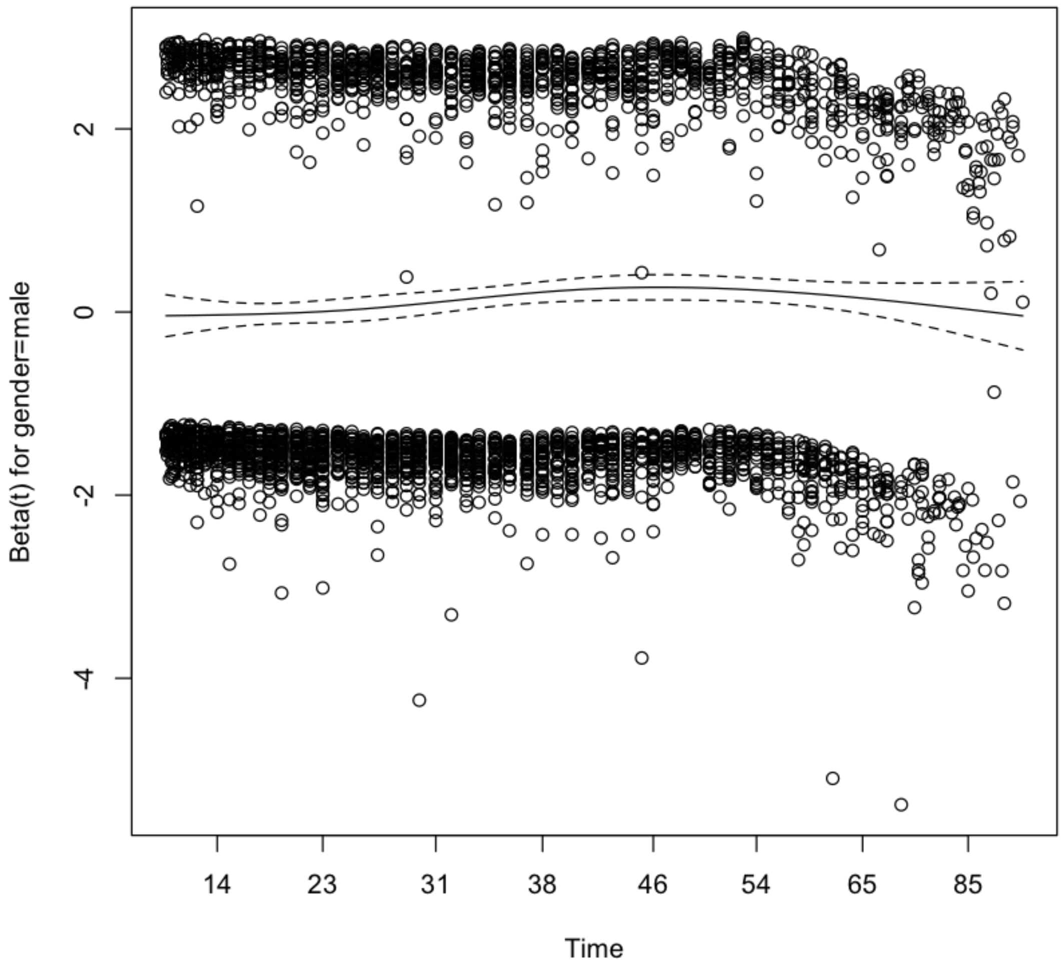


In [43]:
```
# Plot results
plot(summary(fitCPH1), log = TRUE)
```




```
In [45]: # Check proportional hazard assumption and print result
         testCPH <- cox.zph(fitCPH1)
         print(testCPH)

         # Plot time-dependent beta
         plot(testCPH, var = "gender=male")
```

```
                      rho chisq      p
shoppingCartValue -0.0168 0.907 0.3409
voucher           -0.0155 0.770 0.3803
returned           0.0261 2.182 0.1397
gender=male        0.0390 4.922 0.0265
GLOBAL                 NA 8.528 0.0740
```



p-value <0.05 shows that we reject null hypothesis that given variable meets proportional hazarad assumption (predictor has effect does not change over time). Gender's Beta(t) line shows sign changes over time.

```
In [47]: # Validate model, make sure not overfitting
         validate(fitCPH1, method = "crossvalidation",
                  B = 10, dxy = TRUE, pr = FALSE)# no results printed after each cv step
```

```
      index.orig training   test optimism index.corrected  n
Dxy       0.1159   0.1159 0.1122   0.0037          0.1121 10
R2        0.0299   0.0300 0.0284   0.0016          0.0283 10
Slope     1.0000   1.0000 0.9676   0.0324          0.9676 10
D         0.0032   0.0033 0.0041  -0.0008          0.0040 10
U         0.0000   0.0000 0.0003  -0.0003          0.0003 10
```

```
Q         0.0032   0.0033 0.0038  -0.0005          0.0038  10
g         0.2380   0.2383 0.2299   0.0084          0.2296  10
```

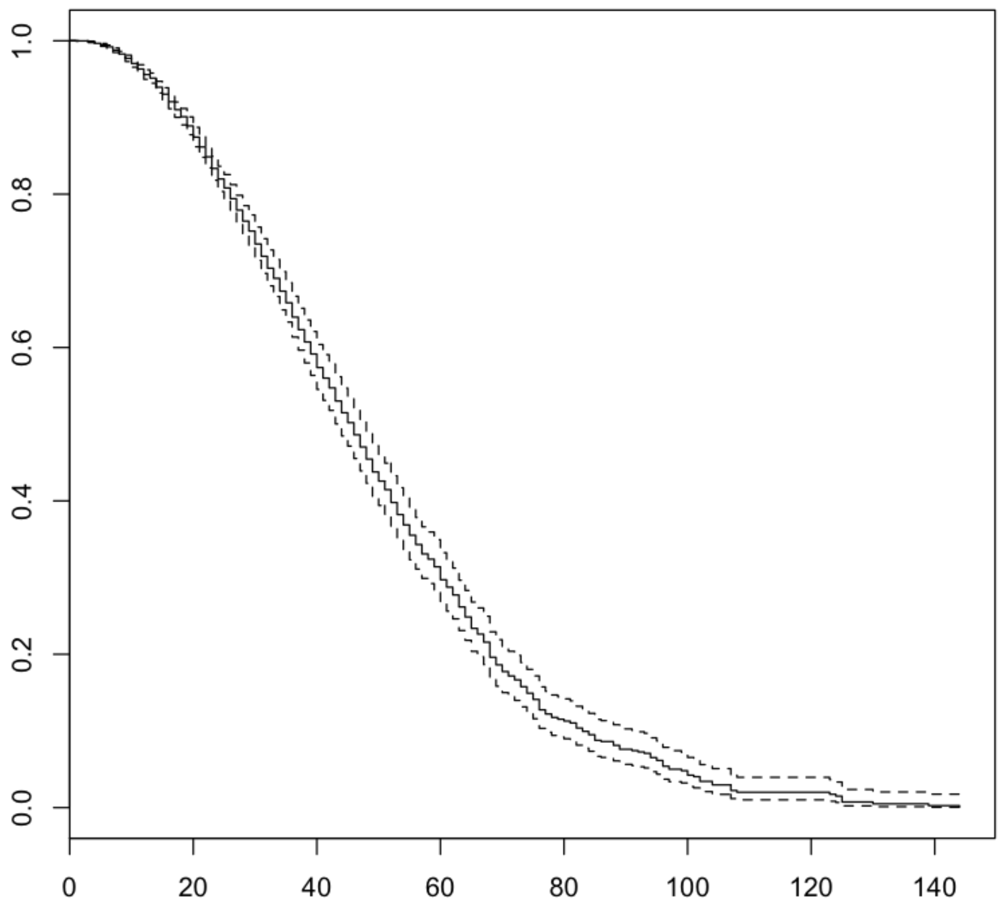
```
In [50]: #stratified analysis
         fitCPH2 <- cph(Surv(daysSinceFirstPurch, boughtAgain) ~ shoppingCartValue + voucher +
                        stratum = "gender = Male",
                        data = dataNextOrder,x = TRUE, y = TRUE, surv = TRUE)
         print(fitCPH2)
```

```
Cox Proportional Hazards Model

 cph(formula = Surv(daysSinceFirstPurch, boughtAgain) ~ shoppingCartValue +
     voucher + returned, data = dataNextOrder, x = TRUE, y = TRUE,
     surv = TRUE, stratum = "gender = Male")

                     Model Tests       Discrimination
                                          Indexes
 Obs      5122    LR chi2     146.85    R2       0.028
 Events   3199    d.f.             3    Dxy      0.108
 Center -0.31     Pr(> chi2) 0.0000     g        0.229
                  Score chi2 132.75     gr       1.257
                  Pr(> chi2) 0.0000

                   Coef    S.E.   Wald Z Pr(>|Z|)
 shoppingCartValue -0.0020 0.0003 -7.29  <0.0001
 voucher           -0.2942 0.0480 -6.13  <0.0001
 returned          -0.3106 0.0494 -6.28  <0.0001
```

```
In [51]: # Create data with new customer
         newCustomer <- data.frame(daysSinceFirstPurch = 21, shoppingCartValue = 99.90,
                                   gender = "female", voucher = 1, returned = 0, stringsAsFact

         # Make predictions
         pred <- survfit(fitCPH2, newdata = newCustomer)
         print(pred)
         plot(pred)

         # Correct the customer's gender
         newCustomer2 <- newCustomer
         newCustomer2$gender <- "male"

         # Redo prediction
         pred2 <- survfit(fitCPH2, newdata = newCustomer2)
         print(pred2)
```

```
Call: survfit(formula = fitCPH2, newdata = newCustomer)
```

```
      n  events  median 0.95LCL 0.95UCL
   5122    3199      46      44      48
Call: survfit(formula = fitCPH2, newdata = newCustomer2)

      n  events  median 0.95LCL 0.95UCL
   5122    3199      46      44      48
```




```
In [ ]:
```