Case 3 - Saving Customers at Vigil Home Security

Jonathan Ratschat / Franziska Bülck

03.11.2019

```
options(tinytex.verbose = TRUE)
```

Preparing dataset

Load dataset

```
#install.packages("readxl")
library(readxl)

Data <- read_excel("SavingCustomers.xlsx")
str(Data)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    45017 obs. of  4 variables:
##  $ Save ID    : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ Offer      : chr  "E" "C" "E" "H" ...
##  $ Save Month : num  5 5 6 3 1 2 3 3 5 5 ...
##  $ Disco Month: num  NA NA NA NA NA 3 NA NA NA 14 ...
```

```
summary(Data)
```

```
##     Save ID         Offer             Save Month    Disco Month
##  Min.   :    1   Length:45017       Min.   :1.00   Min.   : 2.000
##  1st Qu.:11255   Class :character   1st Qu.:2.00   1st Qu.: 7.000
##  Median :22509   Mode  :character   Median :4.00   Median : 9.000
##  Mean   :22509                      Mean   :3.58   Mean   : 8.733
##  3rd Qu.:33763                      3rd Qu.:5.00   3rd Qu.:11.000
##  Max.   :45017                      Max.   :6.00   Max.   :15.000
##                                                    NA's   :28401
```

Description of variables

Save ID: An identifier running from 1 to 45,017.

Offer: “A” through “O” for the 15 most-used offers.

Save Month: The month the save was made. (1=December, 2=January,. . . 6=May).

Disco Month: The month the customer discontinued VHS service. This data field is blank if the customer did not discontinue service during the nine-month period after the safe.

Transformation of variables

```
#Transform ID from integer to character
Data$`Save ID` <- as.character(Data$`Save ID`)

#Transform Offer from character to factor
Data$Offer <- as.factor(Data$Offer)
```

Creation and transformation of new variables for exploratory analysis and model

```
#Create variable DurationSaved
Data$DurationSaved <- Data$`Disco Month`- Data$`Save Month`

#Create binary variable: Was contract cancellated during nine-month period?
Data$Cancelled[is.na(Data$DurationSaved)] <- 0
Data$Cancelled[is.na(Data$Cancelled)] <- 1

#We do not know what happened to customers who did not cancel after the nine
#months. Since we have created a variable that indicated right-censoring
#(Data$Cancelled), we transform NAs to the highest possible duration.
Data$DurationSaved[is.na(Data$DurationSaved)] <- 9
```

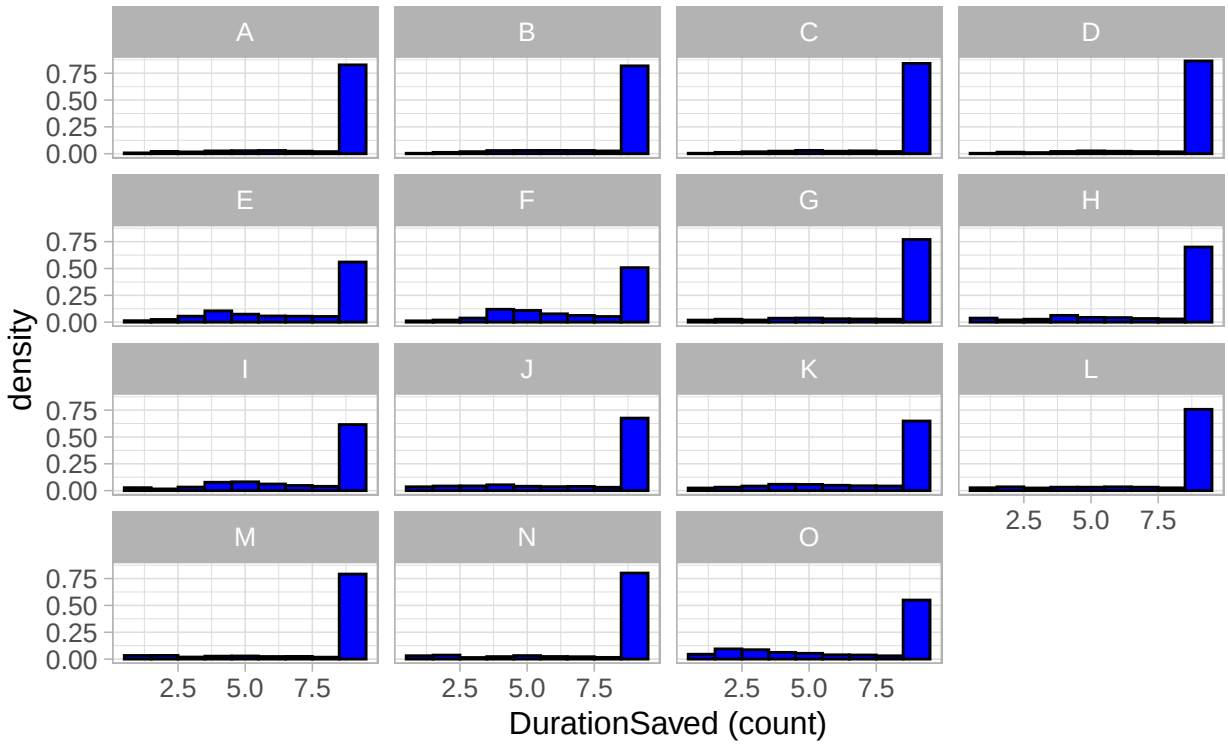
Exploratory analysis

```
#install.packages("ggplot2")
library(ggplot2)
```

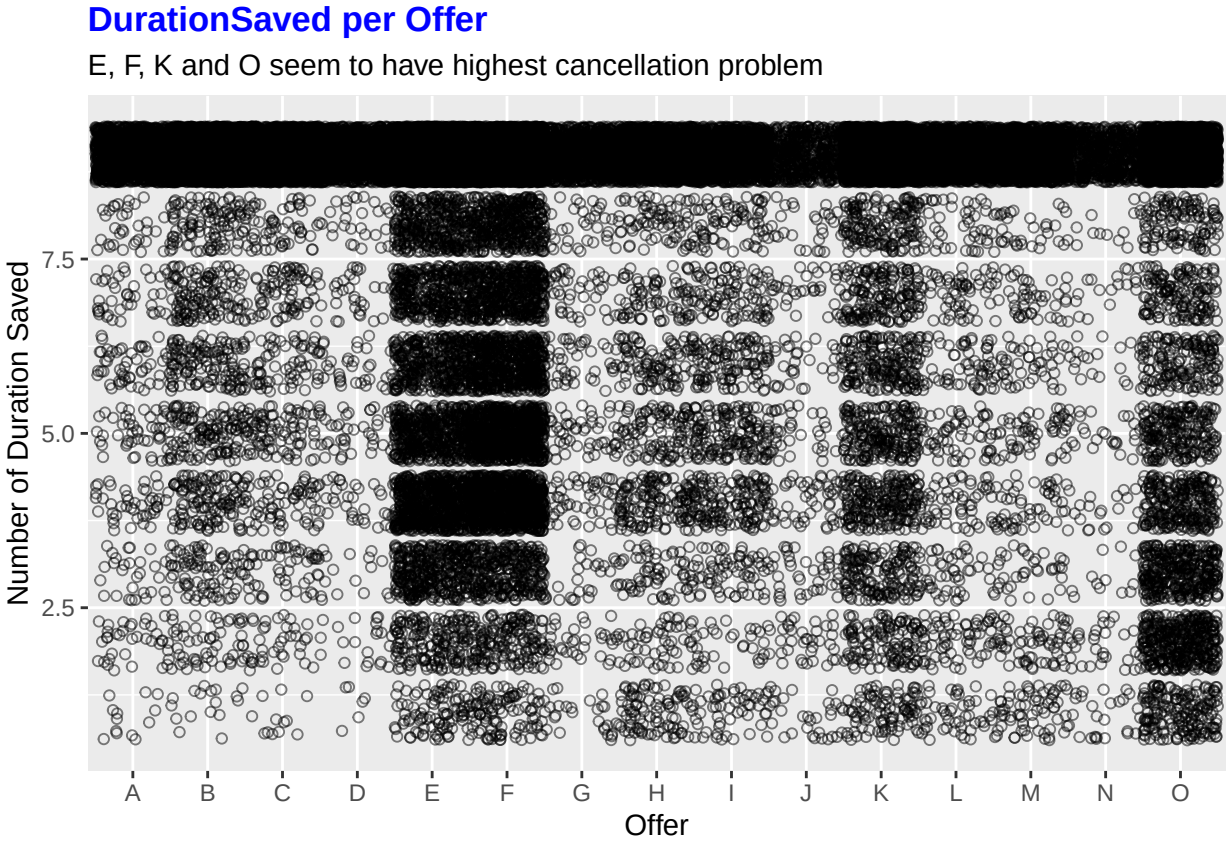
```
ggplot(Data, aes(x=DurationSaved)) +
  geom_histogram(aes(y=..density..),
                 binwidth = 1, color="black", fill = "blue") +
  facet_wrap(~Offer)+ theme_light(base_size=12) +
  ggtitle("Density of DurationSaved per Offer",
          subtitle = "There is a clear difference between the offers used") +
  xlab("DurationSaved (count)") +
  theme(plot.title = element_text(color = "blue", face = "bold"))
```

Density of DurationSaved per Offer

There is a clear difference between the offers used




```
ggplot(Data) +
  geom_jitter(aes(x=Offer,y=DurationSaved),
              shape = 1, width = 0.5, alpha = 0.5) +
  ggtitle("DurationSaved per Offer",
          subtitle = "E, F, K and O seem to have highest cancellation problem") +
  xlab("Offer") + ylab("Number of Duration Saved") +
  theme(plot.title = element_text(color = "blue", face = "bold"))
```



How do the offers affect the longevity of saved customers relationship?

Kaplan-Meier Method

The Kaplan-Meier estimator is a non-parametric statistic that allows us to estimate the survival function (in this case cancellation).

A non-parametric statistic is not based on the assumption of an underlying probability distribution, since survival data has a skewed distribution.

This statistic gives the probability that an individual customer will not cancel past a particular time t.

```
#install.packages("survival")
#install.packages("survminer")
#install.packages("dplyr")

library(survival)
library(survminer)
library(dplyr)

#Create a survival object (compiled version of DurationSaved and Survive)
surv_object <- Surv(time=Data$DurationSaved, Data$Cancelled)

#"+" behind survival times indicates censored data points
head(surv_object, n=50)
```

```
##  [1] 9+ 9+ 9+ 9+ 9+ 1  9+ 9+ 9+ 9  9+ 7  4  1  9+ 9+ 6  9+ 9+ 3  9+ 2  6
## [24] 9+ 7  9+ 1  2  9+ 9+ 9+ 3  7  9+ 3  9+ 9+ 9+ 9+ 9+ 9+ 5  9+ 7  7  9+
## [47] 9+ 9+ 9  9+
```

```
#Fit the Kaplan-Meier curves
fit1 <- survfit(surv_object ~ Offer, data = Data)
summary(fit1)
```

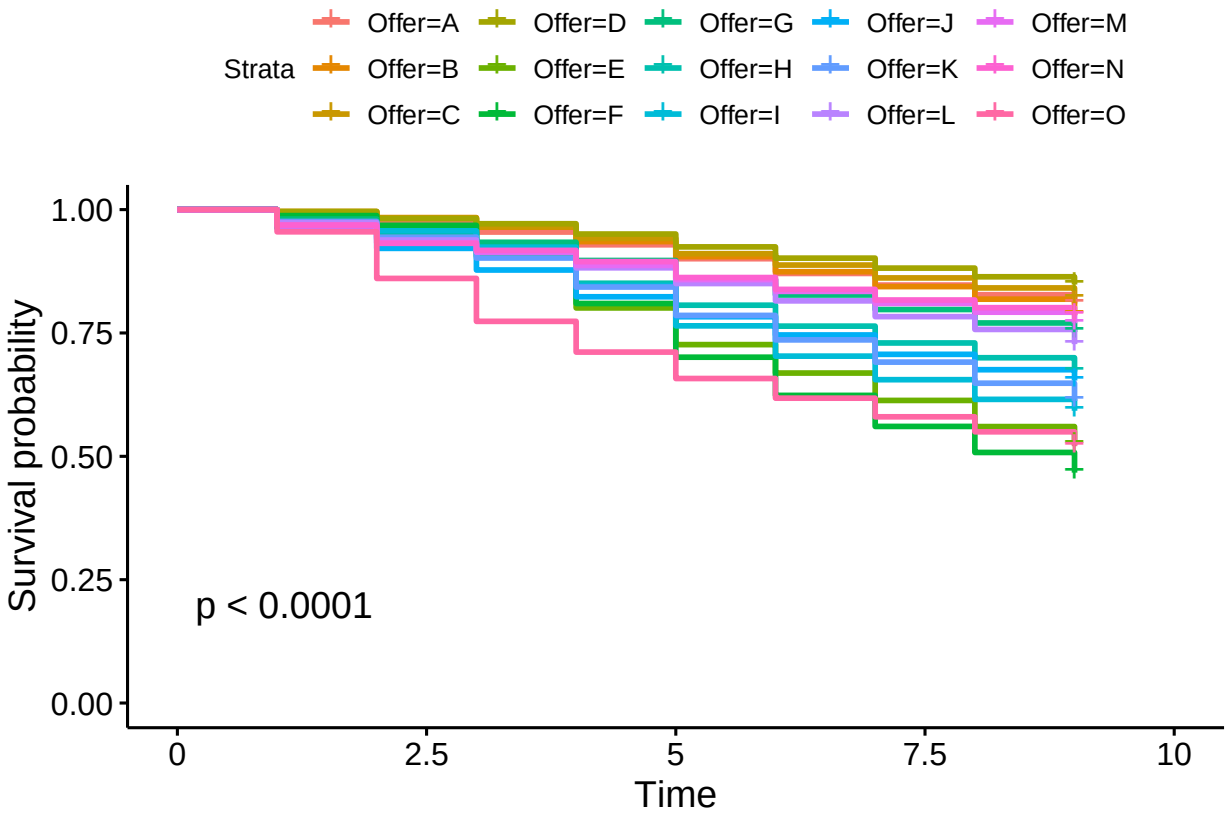
```
## Call: survfit(formula = surv_object ~ Offer, data = Data)
##
##                 Offer=A
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   1739      13    0.993 0.00207        0.988        0.997
##     2   1726      37    0.971 0.00401        0.963        0.979
##     3   1689      29    0.955 0.00499        0.945        0.964
##     4   1660      45    0.929 0.00617        0.917        0.941
##     5   1615      49    0.901 0.00718        0.887        0.915
##     6   1566      52    0.871 0.00805        0.855        0.887
##     7   1514      41    0.847 0.00863        0.830        0.864
##     8   1473      34    0.827 0.00906        0.810        0.845
##     9   1439      31    0.810 0.00941        0.791        0.828
##
##                 Offer=B
##  time n.risk n.event survival  std.err lower 95% CI upper 95% CI
##     1   4275      17    0.996 0.000963        0.994        0.998
##     2   4258      53    0.984 0.001941        0.980        0.987
##     3   4205      82    0.964 0.002832        0.959        0.970
##     4   4123     125    0.935 0.003765        0.928        0.943
##     5   3998     130    0.905 0.004489        0.896        0.914
##     6   3868     131    0.874 0.005073        0.864        0.884
##     7   3737     129    0.844 0.005550        0.833        0.855
##     8   3608     109    0.818 0.005895        0.807        0.830
##     9   3499     136    0.787 0.006266        0.774        0.799
##
##                 Offer=C
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   2911      12    0.996 0.00119        0.994        0.998
##     2   2899      36    0.984 0.00236        0.979        0.988
##     3   2863      53    0.965 0.00339        0.959        0.972
##     4   2810      69    0.942 0.00435        0.933        0.950
##     5   2741      90    0.911 0.00529        0.900        0.921
##     6   2651      67    0.888 0.00585        0.876        0.899
##     7   2584      76    0.862 0.00640        0.849        0.874
##     8   2508      59    0.841 0.00677        0.828        0.855
##     9   2449      62    0.820 0.00712        0.806        0.834
##
##                 Offer=D
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   1535       6    0.996 0.00159        0.993        0.999
##     2   1529      22    0.982 0.00342        0.975        0.988
##     3   1507      16    0.971 0.00426        0.963        0.980
##     4   1491      32    0.950 0.00554        0.940        0.961
##     5   1459      40    0.924 0.00675        0.911        0.938
##     6   1419      35    0.902 0.00760        0.887        0.917
##     7   1384      31    0.881 0.00825        0.865        0.898
```

```
##     8   1353      27    0.864 0.00875        0.847        0.881
##     9   1326      24    0.848 0.00916        0.830        0.866
##
##                Offer=E
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   6790      92    0.986 0.00140        0.984        0.989
##     2   6698     170    0.961 0.00234        0.957        0.966
##     3   6528     378    0.906 0.00355        0.899        0.913
##     4   6150     713    0.801 0.00485        0.791        0.810
##     5   5437     505    0.726 0.00541        0.716        0.737
##     6   4932     391    0.669 0.00571        0.658        0.680
##     7   4541     377    0.613 0.00591        0.602        0.625
##     8   4164     360    0.560 0.00602        0.549        0.572
##     9   3804     249    0.524 0.00606        0.512        0.536
##
##                Offer=F
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   8318     101    0.988 0.00120        0.986        0.990
##     2   8217     164    0.968 0.00193        0.964        0.972
##     3   8053     319    0.930 0.00280        0.924        0.935
##     4   7734     999    0.810 0.00430        0.801        0.818
##     5   6735     905    0.701 0.00502        0.691        0.711
##     6   5830     644    0.623 0.00531        0.613        0.634
##     7   5186     523    0.561 0.00544        0.550        0.571
##     8   4663     439    0.508 0.00548        0.497        0.519
##     9   4224     340    0.467 0.00547        0.456        0.478
##
##                Offer=G
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   1358      26    0.981 0.00372        0.974        0.988
##     2   1332      37    0.954 0.00571        0.942        0.965
##     3   1295      27    0.934 0.00675        0.921        0.947
##     4   1268      50    0.897 0.00825        0.881        0.913
##     5   1218      52    0.859 0.00945        0.840        0.877
##     6   1166      43    0.827 0.01027        0.807        0.847
##     7   1123      40    0.797 0.01091        0.776        0.819
##     8   1083      37    0.770 0.01142        0.748        0.793
##     9   1046      25    0.752 0.01172        0.729        0.775
##
##                Offer=H
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   2405      91    0.962 0.00389        0.955        0.970
##     2   2314      50    0.941 0.00479        0.932        0.951
##     3   2264      66    0.914 0.00572        0.903        0.925
##     4   2198     152    0.851 0.00727        0.837        0.865
##     5   2046     107    0.806 0.00806        0.791        0.822
##     6   1939     102    0.764 0.00866        0.747        0.781
##     7   1837      82    0.730 0.00906        0.712        0.748
##     8   1755      72    0.700 0.00935        0.682        0.718
##     9   1683      67    0.672 0.00957        0.653        0.691
##
##                Offer=I
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   1906      52    0.973 0.00373        0.965        0.980
```

```
##     2   1854      30    0.957 0.00465        0.948        0.966
##     3   1824      63    0.924 0.00607        0.912        0.936
##     4   1761     148    0.846 0.00826        0.830        0.863
##     5   1613     156    0.764 0.00972        0.746        0.784
##     6   1457     117    0.703 0.01047        0.683        0.724
##     7   1340      91    0.655 0.01089        0.634        0.677
##     8   1249      76    0.615 0.01114        0.594        0.638
##     9   1173      44    0.592 0.01126        0.571        0.615
##
##                Offer=J
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1    866      31    0.964 0.00631        0.952        0.977
##     2    835      37    0.921 0.00914        0.904        0.940
##     3    798      38    0.878 0.01114        0.856        0.900
##     4    760      47    0.823 0.01296        0.798        0.849
##     5    713      35    0.783 0.01401        0.756        0.811
##     6    678      32    0.746 0.01479        0.718        0.776
##     7    646      34    0.707 0.01547        0.677        0.738
##     8    612      27    0.676 0.01591        0.645        0.707
##     9    585      19    0.654 0.01617        0.623        0.686
##
##                Offer=K
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   3996      96    0.976 0.00242        0.971        0.981
##     2   3900     127    0.944 0.00363        0.937        0.951
##     3   3773     169    0.902 0.00471        0.893        0.911
##     4   3604     235    0.843 0.00575        0.832        0.854
##     5   3369     230    0.786 0.00649        0.773        0.798
##     6   3139     198    0.736 0.00697        0.722        0.750
##     7   2941     180    0.691 0.00731        0.677        0.705
##     8   2761     171    0.648 0.00755        0.634        0.663
##     9   2590     144    0.612 0.00771        0.597        0.627
##
##                Offer=L
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   1849      49    0.973 0.00374        0.966        0.981
##     2   1800      65    0.938 0.00559        0.927        0.949
##     3   1735      45    0.914 0.00652        0.901        0.927
##     4   1690      59    0.882 0.00750        0.868        0.897
##     5   1631      59    0.850 0.00830        0.834        0.867
##     6   1572      65    0.815 0.00903        0.798        0.833
##     7   1507      59    0.783 0.00958        0.765        0.802
##     8   1448      48    0.757 0.00997        0.738        0.777
##     9   1400      55    0.727 0.01036        0.707        0.748
##
##                Offer=M
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   1919      64    0.967 0.00410        0.959        0.975
##     2   1855      65    0.933 0.00572        0.922        0.944
##     3   1790      36    0.914 0.00640        0.902        0.927
##     4   1754      52    0.887 0.00723        0.873        0.901
##     5   1702      55    0.858 0.00796        0.843        0.874
##     6   1647      46    0.834 0.00849        0.818        0.851
##     7   1601      47    0.810 0.00896        0.792        0.828
```

```
##     8   1554      34    0.792 0.00926        0.774        0.810
##     9   1520      43    0.770 0.00961        0.751        0.789
##
##                 Offer=N
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1    835      26    0.969 0.00601        0.957        0.981
##     2    809      31    0.932 0.00873        0.915        0.949
##     3    778      12    0.917 0.00953        0.899        0.936
##     4    766      19    0.895 0.01063        0.874        0.916
##     5    747      27    0.862 0.01193        0.839        0.886
##     6    720      20    0.838 0.01274        0.814        0.864
##     7    700      18    0.817 0.01339        0.791        0.843
##     8    682      13    0.801 0.01381        0.775        0.829
##     9    669      13    0.786 0.01420        0.758        0.814
##
##                 Offer=O
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   4315     194    0.955 0.00315        0.949        0.961
##     2   4121     408    0.860 0.00527        0.850        0.871
##     3   3713     375    0.774 0.00637        0.761        0.786
##     4   3338     269    0.711 0.00690        0.698        0.725
##     5   3069     231    0.658 0.00722        0.644        0.672
##     6   2838     172    0.618 0.00740        0.604        0.633
##     7   2666     163    0.580 0.00751        0.566        0.595
##     8   2503     131    0.550 0.00757        0.535        0.565
##     9   2372     126    0.521 0.00761        0.506        0.536
```

```
ggsurvplot(fit1, data = Data, pval = TRUE)
```



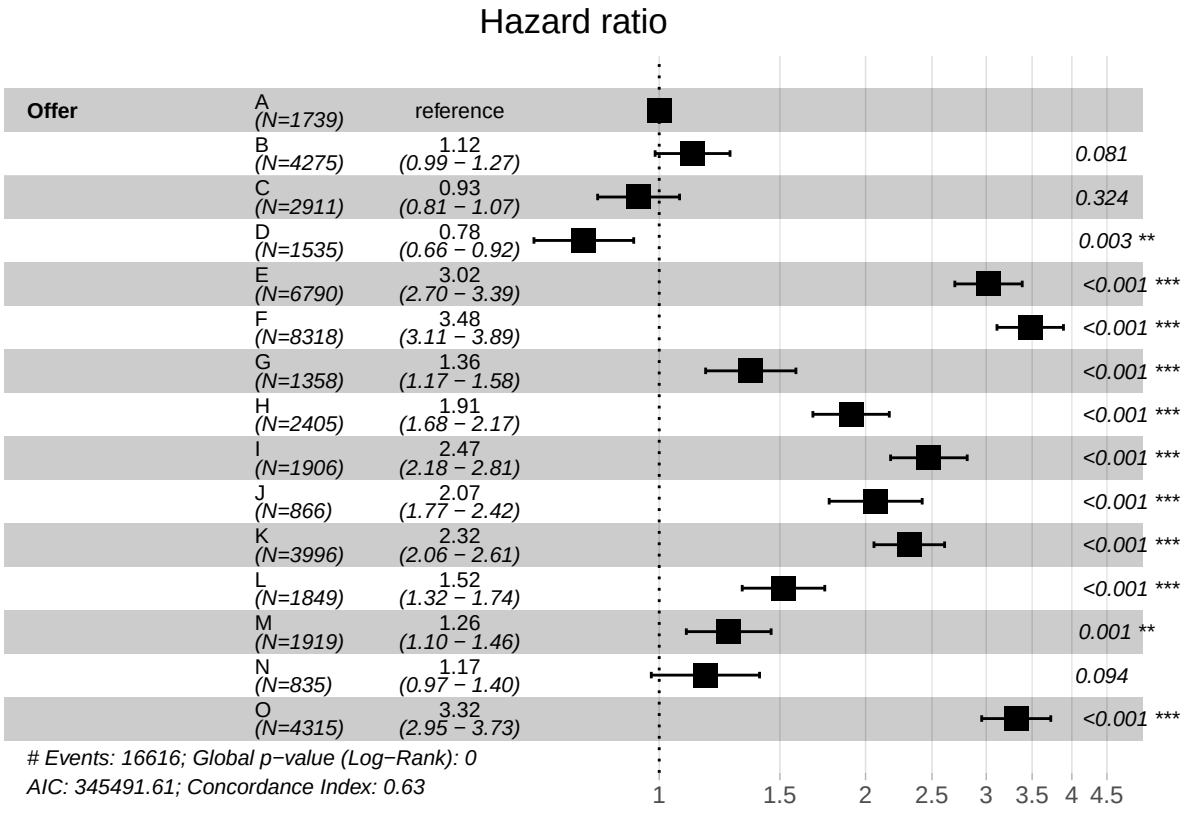
We can use the log-rank test to compare survival curves of two groups. The log-rank test is a statistical hypothesis test that tests the null hypothesis that survival curves of two populations do not differ. The log-rank p-value of 0.0001 indicates a significant result, therefore, the survival curves differ.

Offer A to D have the highest probabilities that an individual customer will not cancel past 9 months (>80%), while offers E, F and O have the lowest probabilities (<55%).

Cox Proportional Hazards Models

It describes the probability of an event or its hazard (cancellation in this case) if the customer survived up to that particular time point t. It is a bit more difficult to illustrate than the Kaplan-Meier estimator because it measures the instantaneous risk of cancellation. Nevertheless, we need the hazard function to consider covariates when we compare survival of patient groups. Covariates, also called explanatory or independent variables in regression analysis, are variables that are possibly predictive of an outcome or that we might want to adjust for to account for interactions between variables.

```
# Fit a Cox proportional hazards model
fit.coxph <- coxph(surv_object ~ Offer, data = Data)
ggforest(fit.coxph, data = Data)
```



Offer A was used as a reference to calculate the hazard ratio.

An hazard ratio > 1 indicates an increased risk of cancellation if a specific offer is given to a customer. An hazard ratio < 1, on the other hand, indicates a decreased risk.

Therefore, the Cox proporational hazard model indicates that offers E, F and O have a relatively high risk of cancellation. These results are significant.

On the other hand, it indicates that offer D has a relatively low risk of cancellation. There are other offers with low hazard ratios (e.g. A and C), but these results are not significant.

Conclusion

The offers affect the longevity of saved customers relationship as shown with the two models. We recommend using offer B since it performs well in both models. For further analysis, we would need to analyse if offer B also leads to the highest feasible profits.