Survival And Hazard Models To Analyze And Improve Profitability Of Freemium Models.

What do Spotify, Todoist, Evernote and LinkedIn have in common? They all run under a freemium business model. The *freemium* model became an especially popular business model in the digital startup, newspaper and service scene since the emergence of *Software-as-a-Service (SaaS)*. The basic idea of the freemium model is that a service provider offers a less functional version of its product for free to encourage premium subscription because potential buyers can explore and test the free version and avoid a risky big jump. SaaS, on the other hand, was enabled by cloud computing that opened up the possibility to migrate the whole software on a cloud. Hence, the customer does not need to install the software and update it anymore. He just subscribes online and he can use it via the server like a service.

In this article, I will shortly show you how to analyze the freemium model using a survival and hazard model, what assumptions a freemium model rests on, and how you can use the information from the survival and hazard model to derive actions on how to improve the business model on the example of a fictive software company called “SeventhCloud”.

SeventhCloud initially developed a software that extracts procurement data from its customer’s ERP system and produces a dashboard to show saving potentials as well as possible compliance risks. The customers are mainly based in northern and western Europe and the company had faced problems attracting new customers with the software. After a review, it found out that new startups, which were growing at a fast pace, are offering SaaS-products rather than software. The board decided to follow the trend and kick-off the digital transformation as well as develop their own SaaS-solution. While developing the new product, the question emerged whether a freemium model would be more profitable than a general subscription model and how they could improve their subscription model.

The Problem: The hidden costs of the freemium model

SeventhCloud has launched a prototype of the SaaS solution as a free version and if interested, customers could pre-register for the premium version that is scheduled to come in three months. Due to time reasons, they have only followed their current subscribers for 30 days. The information that they have gathered can be expressed by the following in R simulated dataset:

```
################################## Create Dataset
# Duration & Censored
```

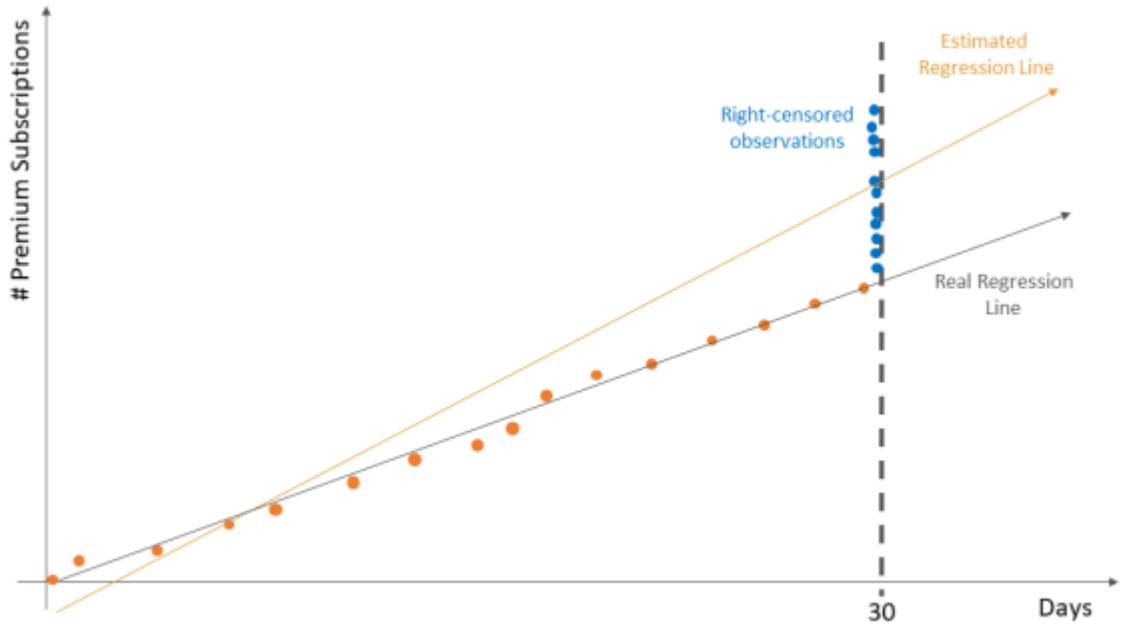
```
set.seed(1234)
duration <- round(abs(rnorm(n=89, mean=18, sd=12))*24) # in hours
duration[sample(1:89, 10)] <- 0 # some immeditally made decisions
censored <- ifelse(duration > 30*24, "yes", "no")
duration[duration > 30*24] <- 30*24 # Censoring

# ID
id <- 1:89

# subscription
subscription <- ifelse((sample(c(TRUE,FALSE, FALSE),89,replace = TRUE)
& (censored == "no")),"yes","no")

# Time spent on Application in days (out of subscription time)
appdays <- c()

for (k in 1:89) {
  if (subscription[k] == "yes") {
    appdays <- c(appdays, sample(1:(duration[k]/24),1))
  } else {
    appdays <- c(appdays, round(sample(1:(duration[k]/24),1)/4))
  }
}



# Industry
industries <- c("Manufacturing", "IT", "Telecommunication",
"Consulting", "Food", "Automotive","Health", "Finance")
prob_industries <- c(0.3,0.3,0.05, 0.01, 0.04,0.2,0.02, 0.08)
industry <- sample(industries,89,replace = TRUE, prob=prob_industries)

# Size
```

```
size <- sample(c("1-50","51-1000","1001+"), 89, replace = TRUE, prob =
c(0.6,0.35,0.5) )

# Previous customer
prevcustomer <- sample(c("yes", "no"),89,replace = TRUE, prob =
c(0.1,0.9))

# Creating Dataset
Subscription <- data.frame(id=id, duration=duration,
censored=censored, subscription=subscription, appdays=appdays,
industry=industry, prevcustomer=prevcustomer, size=size)
Subscription$id <- as.factor(Subscription$id)
rm(id, duration, censored, subscription, appdays, industry,
prevcustomer, size, industries, prob_industries, k)
```

As already indicated, while exploring the freemium model, SeventhCloud had two important question that had to be answered in order for it to be able to develop its strategical plan.

1. **What is the cost and revenue per 100 subscriptions (free and premium-version)?**
   The question is not trivial, because, for every 100 new users, some will unsubscribe, some will stay using the free version and others will sign up for the premium version. Furthermore, by understanding the costs, we will understand whether we will need to limit the free version. By understanding the benefits, we would be able to say, whether the costs are worth it.
2. **How can we improve the freemium model and increase the premium subscription rate?**
   This question aims at understanding the customer base and finding ways of improving the value of the SaaS solution.

So, the question that might arise now is, why do we need a more complex survival model to answer these questions? Wouldn't simply taking proportions, averaging or a simple linear regression suffice? Well, the answer is no due to the nature of the data we have. First, we have collected *spell data*, which describes durations. Spells cannot be negative, which is why we cannot use a simple linear regression. Second, we have *right-censored* data. Censoring generally describes the problem that we know that some values lie in a certain range, but we do not know the exact value. In our case, we are only able to follow all subscribers up to a 30-day period, but they might still subscribe at some point beyond the 30 days. Hence, where censored is "yes", we just know that the premium subscription has not happened until day 30, but might have happened any point after day 30. Using just averages or a linear regression will generally lead to a biased result because the censored observation is treated as if they occurred at the censoring time (see figure below).



Censoring is a different than *truncation*. In truncation, an observation that would lie within a specific interval was completely left out and not observed. This is not applicable in our case. Overall, specific limited dependent models such as the Tobit model or Heckmann model have been developed to estimate models in case of censoring or truncation and we can differentiate between these general types of censoring and truncation:

- **Left-censoring**: The event of interested (for instance subscription) has been recorded at the border of the censoring value x, but it is known that it occurred before this specific point x.
- **Interval-censoring**: The event of interested has been recorded, but it is only known that it occurred between two values x_1 and x_2.
- **Right-censoring:** The event of interested has been recorded, but it is known that it has occurred beyond a specific point x.
- **Left-truncation:** The event of interest has not been recorded, because it lies below a specific threshold.
- **Interval-truncation:** The event of interest has not been recorded, because it lies within a specific interval.
- **Right-truncation:** The event has not been recorded because it lies beyond a specific value x.

Luckily, survival and hazard models are also able to deal with censored data. In order to address the two questions, we will apply survival and hazard models.

The Method & Premises: Does the customer survive, unsubscribe or subscribe?

The main purpose of survival and hazard models is to investigate the time until a certain event occurs. Therefore, the starting event (free subscription) and the end event (premium subscription or unsubscription) need to be clearly defined. The models allow someone to assess not only the risk at a specific time for an event to occur or the probability to survive until a certain point but also what factors influence these probabilities and whether groups differ from each other.

We apply these models to *spell data*, e.g. data about durations. However, there are two types of spell data:

1. **Single-spell data:** each individual studied can only have the event once.
2. **Mutliple-spell data:** each individual studied can experience the event of interested several times.

The models we introduce here, work only for single spell data. If we have multiple-spell data, we can only use the single spell models if the independent variables have a constant effects regardless of the period or episode (1), if the duration distribution of an individual only depends on the time since entry into the present state (2), and if successive episodes of individuals are independent (3). These assumptions, however, rarely hold true. If we study for instance the probability for an individual to get a heart attack, then the older a person gets, the stronger the effects of other health-related variables get (the first assumption violated), the more likely a heart attack is (the second assumption violated). And people, that already experienced a heart attack, are at higher risk to suffer another heart attack (the third assumption violated). In our case, we have single spell data.

First, we will compute a survival model for the probability that a user stays a free user and does not unsubscribe. This survival model will help us assess the cost of all subscribers. Then we will calculate a second hazard model to assess the probability that a free user will subscribe to the premium offer. This will help us estimate the revenue from premium users. In order to make these estimations, we have got the following information from the SeventhCloud managers:

- The fix costs are around 100.000€ per month and include the whole IT-infrastructure, the people staffed in that project and further overhead costs.
- The variable costs are 20€ per subscription per day regardless of whether the user is

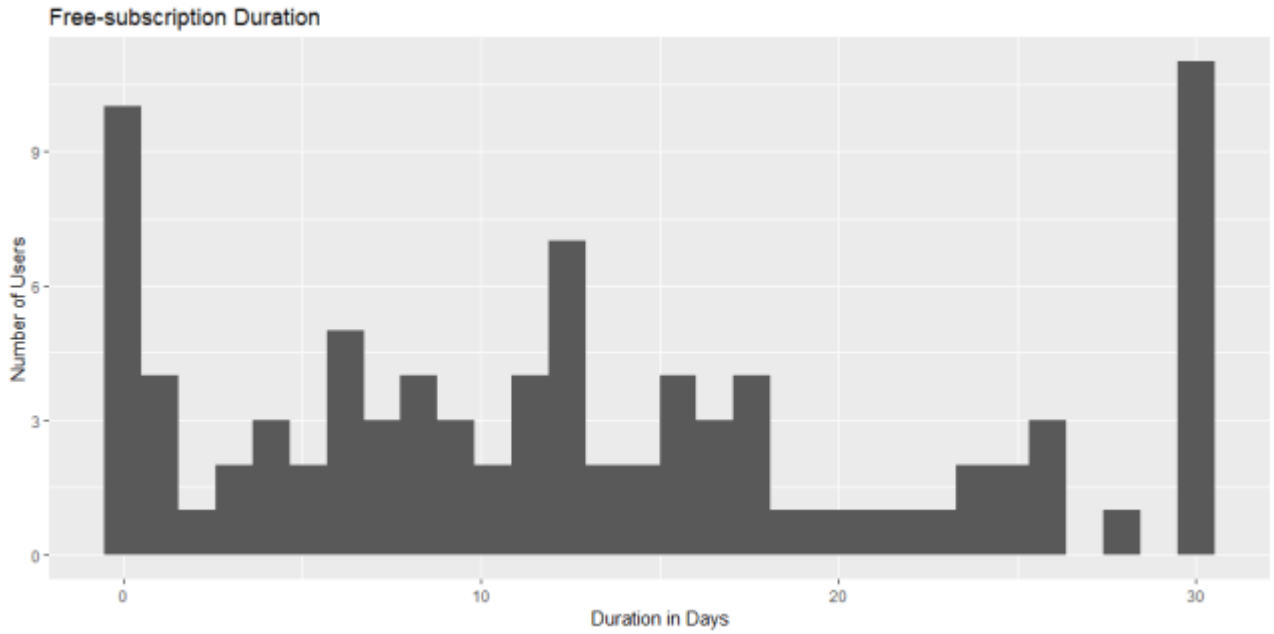
a premium or free user.
- The premium model generates a revenue of 100€ per company per day.

In the next section, we will solve the three problems of SeventhCloud using the information given and implement survival and hazard models using R. More concretely I will implement a non-parametric Kaplan Meier Model and a Cox proportional hazards regression model. Since the Cox proportional hazards regression model is only semi-parametric, both models applied are not parametric models because we do not assume any underlying distribution. This makes them suitable to any kind of data and more powerful when the underlying relationship really does not follow a certain shape, but less powerful if the underlying relationship does follow a certain shape.

Solution: Survival and hazard models

We have defined the moment that people register, for a free subscription or premium subscription directly, as the starting event for the spell data. Since SeventhCloud was only able to track the people for 30 days, there will be no subscription duration longer than 30 days. If the ending event has not occurred until the end of the 30^{th} day, the observation is right censored. However, there is a small trick about the definition of the ending event, which we will exploit for the cost, revenue, and profit estimation. In our case, we have two ending events. The first ending event is a premium subscription. In that case, we recorded the hours counted since free subscription until the premium subscription. The second ending event is subscription. In that case, we recorded the hours counted since free subscription until premium subscription. This is unusual for survival analysis, which usually requires only one clearly defined ending event. However, before we go into the estimation of the models, we should first have a look at the data that we have.

```
################################ Descriptive Statistics
# install.packages("ggplot2")
summary(Subscription)
library(ggplot2)

# duration
ggplot(Subscription, aes(x=duration/24)) + geom_histogram() +
  ggtitle("Free-subscription Duration") +
  xlab("Duration in Days") +
  ylab("Number of Users")
```



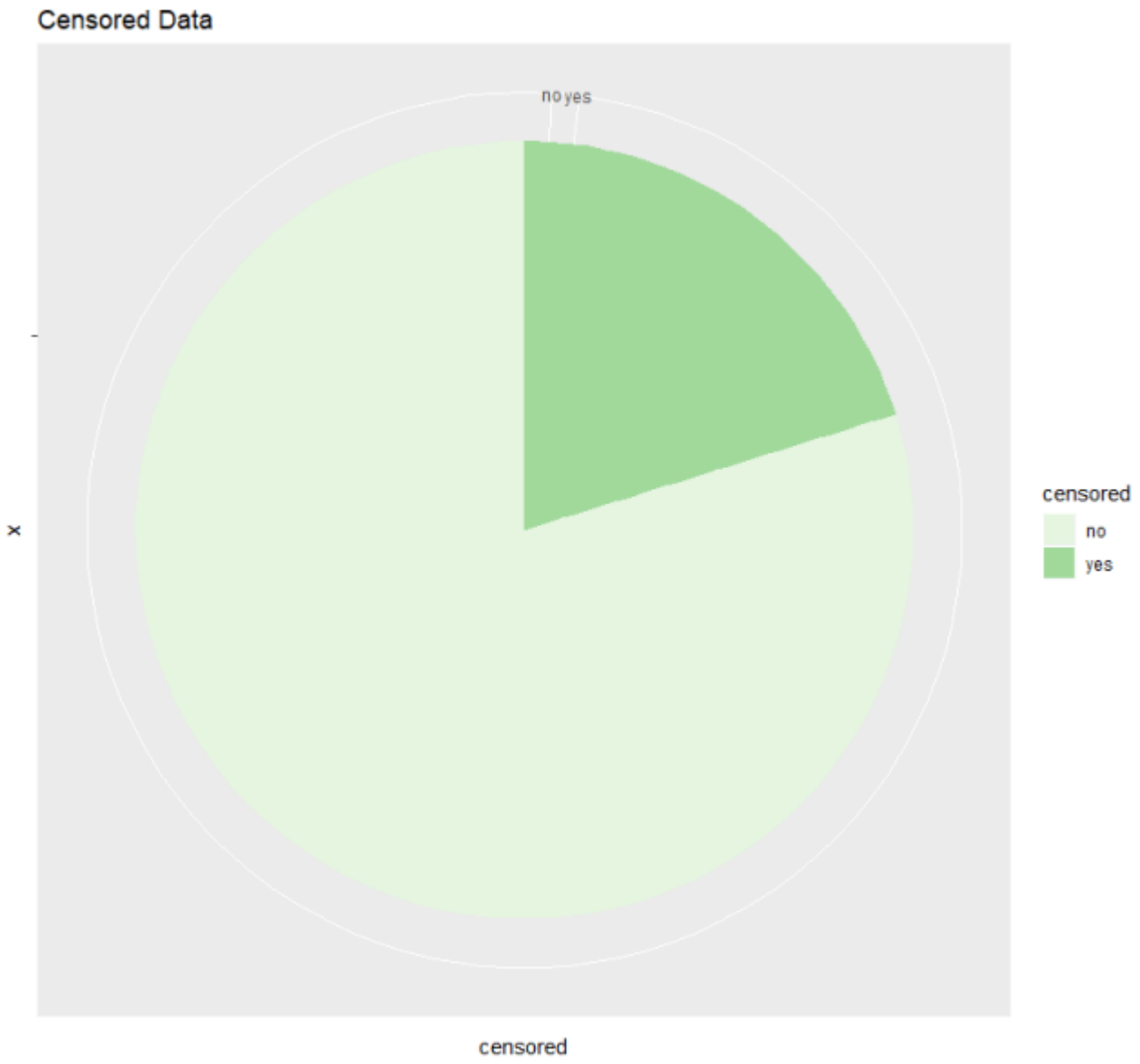
We can clearly see the right peack at 30 days which confirms that our data is right-censored. Luckily, we have a variable indicating which observations have been censored. Let’s have a closer look at that.

```
# censored
ggplot(Subscription, aes(x="", y=censored, fill=censored))+
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y", start=0) +
  scale_fill_brewer(palette="Dark3") +
```

```
ggtitle("Censored Data")
```

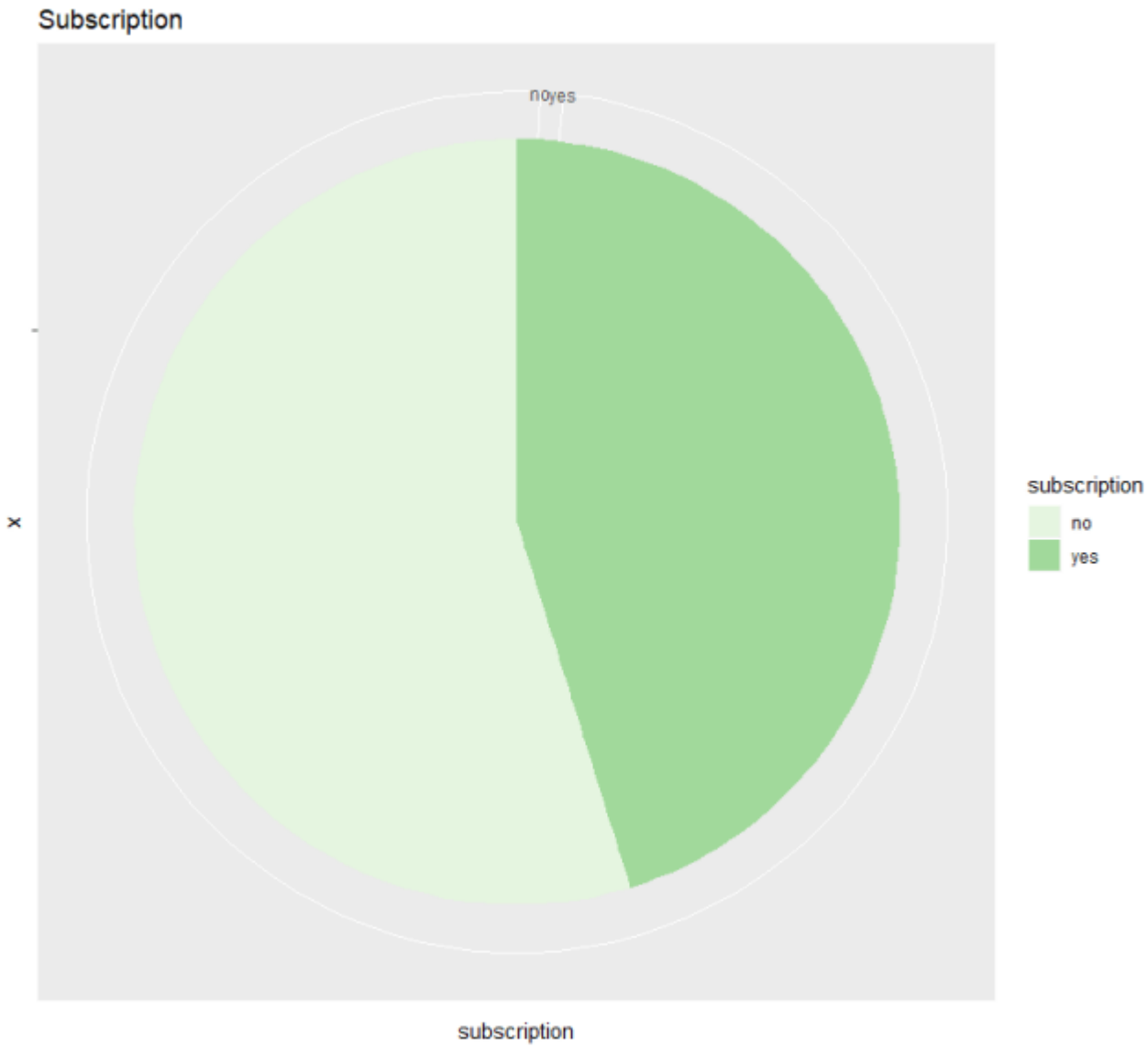
```
[code language="r"]
# censored
ggplot(Subscription, aes(x="", y=censored, fill=censored))+
 geom_bar(width = 1, stat = "identity") +
 coord_polar("y", start=0) +
 scale_fill_brewer(palette="Dark3") +
 ggtitle("Censored Data")
```

```
[/code]
```



Around a fifth of the observations seem to have been censored. That’s quite some.

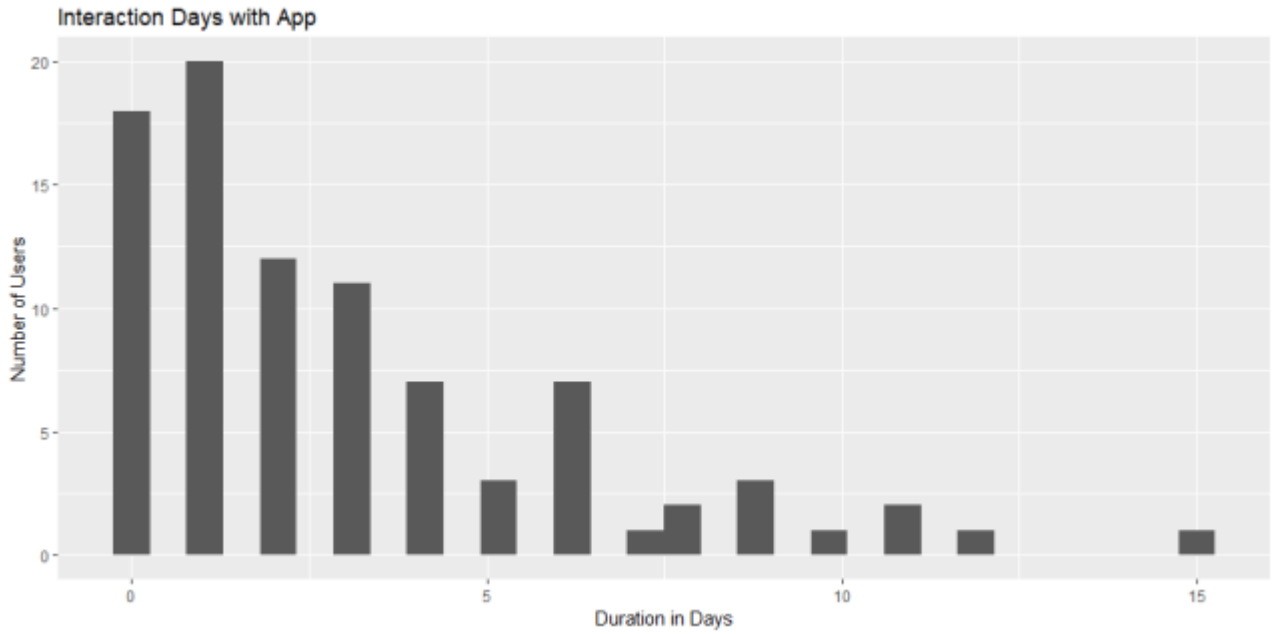
```
# subscription
ggplot(Subscription, aes(x="", y=subscription, fill=subscription))+
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y", start=0) +
  scale_fill_brewer(palette="Dark3") +
  ggtitle("Subscription")
```



Out of the many people that have signed up, around 45% chose a premium subscription after 30 Days. That is quite surprisingly good deal, however, 45% is a “biased” conversion rate as I have already illustrated above since we have right censored data. In fact, the real conversion rate will be different than 45%.

```
# appdays
ggplot(Subscription, aes(x=appdays)) + geom_histogram() +
  ggtitle("Interaction Days with App") +
  xlab("Duration in Days") +
```

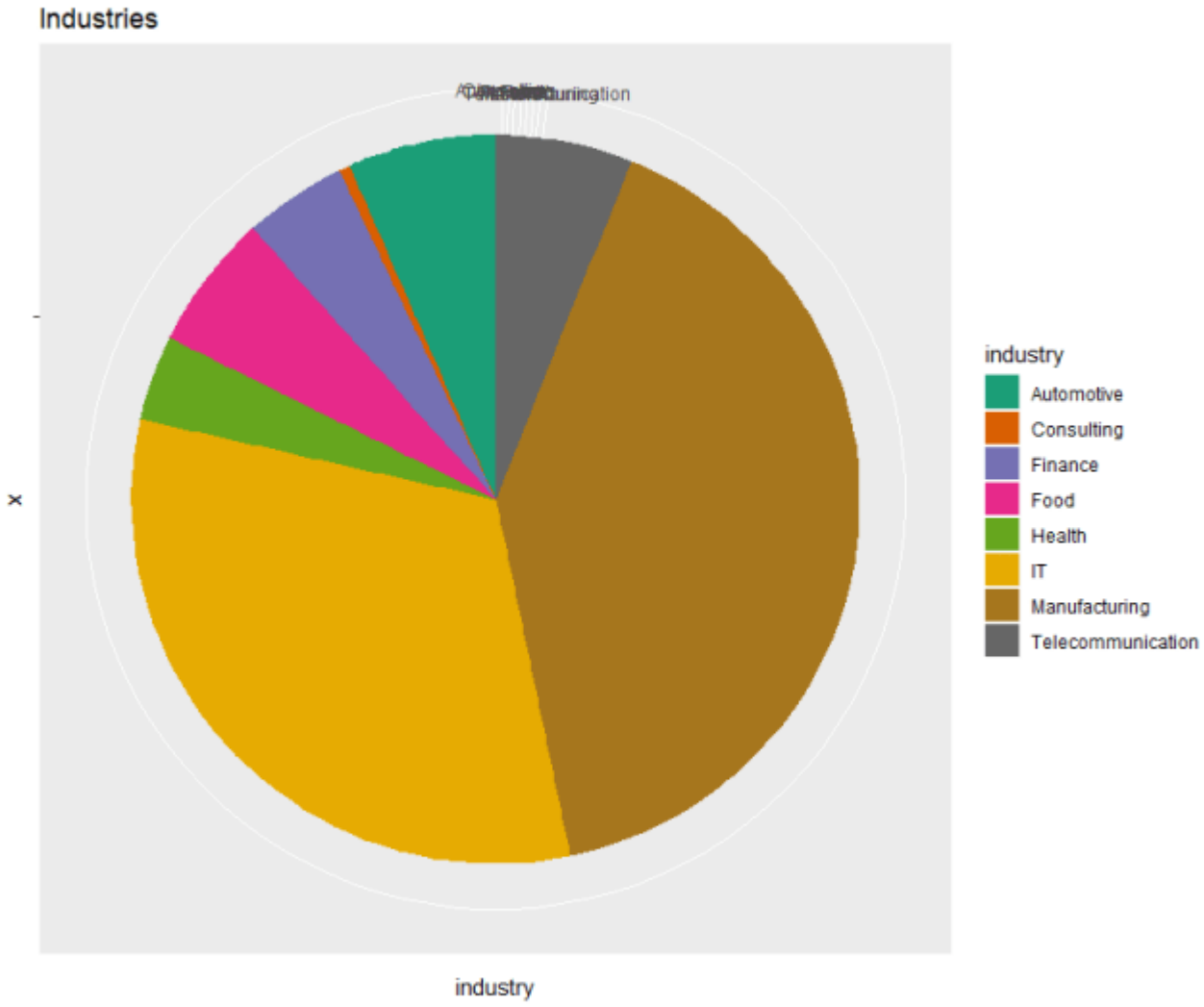
```
ylab("Number of Users")
```



Apparently we have a wide variations how many days SeventhCloud's application was used.

```
# industry
ggplot(Subscription, aes(x="", y=industry, fill=industry))+
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y", start=0) +
  scale_fill_brewer(palette="Dark2") +
```

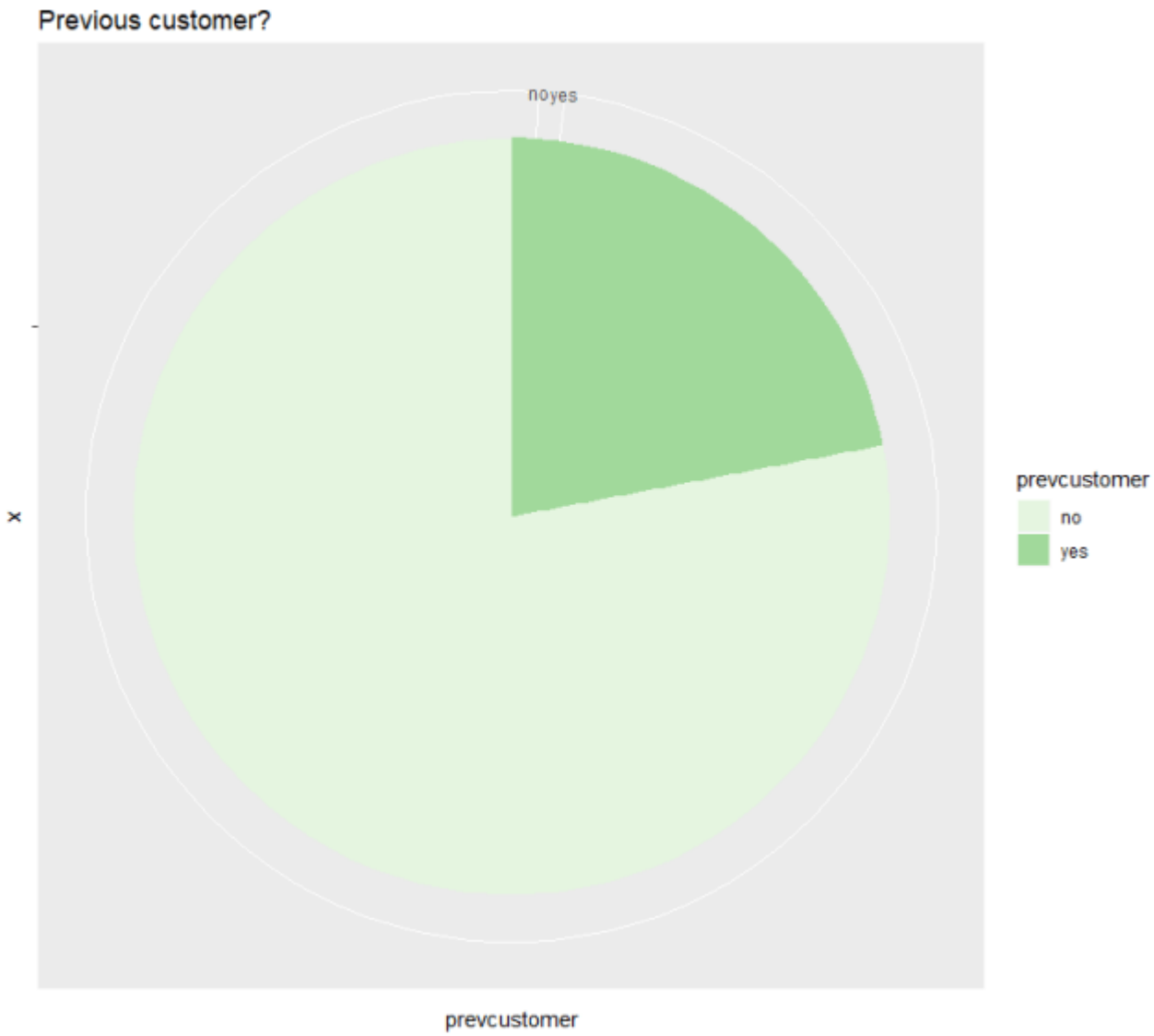
```
ggtitle("Industries")
```



Interestingly, the SaaS-product seems to appeal especially to the IT and manufacturing industry. That's something worth noting and investigating further.

```
# prevcustomer
ggplot(Subscription, aes(x="", y=prevcustomer, fill=prevcustomer))+
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y", start=0) +
  scale_fill_brewer(palette="Dark3") +
```

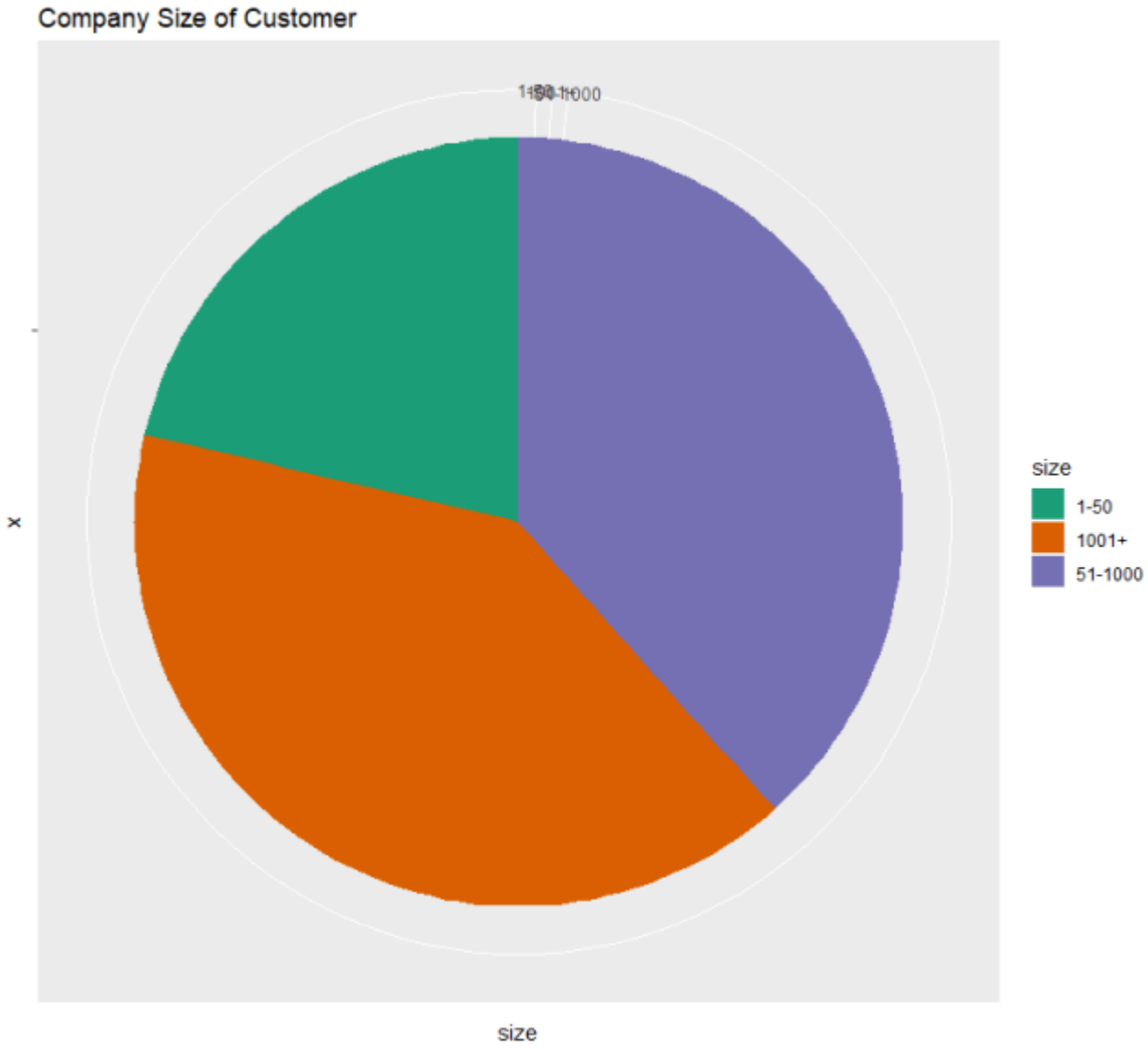
```
ggtitle("Previous customer?")
```



This plot also reveals something unexpected. Apparently only a small fraction of the subscribers have been previous customers. The SaaS-product that is being developed is able to attract a new customer segment that is different from SeventhClouds current customers.

```
# size
ggplot(Subscription, aes(x="", y=size, fill=size))+
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y", start=0) +
  scale_fill_brewer(palette="Dark2") +
```

```
ggtitle("Company Size of Customer")
```



The company size of the customers seems to matter less, even though bigger companies seem to be more likely to subscribe. This is understandable at the end, because our solution appeals only to companies that need and have implemented a good ERP system.
We will also need to prepare some packages and functions that we will need later.

```
################################## Survival Analysis
# Adding Variable unsubscribed
Subscription$unsub <- ifelse((Subscription$subscription == "no")
```

```
                              , "yes", "no")

# install.packages(survival)
install.packages("survival")
library(survival) # for survival & hazard models

install.packages("survminer")
library(survminer) # for ggsurvplot

# function that interpolates mathematical step-functions
step_approx <- function(x,y) {
  new_x <- c()
  new_y <- c()
  for (i_x in x[1]:x[length(x)]) {
    if (i_x %in% c(x)) {
      pos <- which(x == i_x)
      i_y <- y[pos]
    }
    new_x <- c(new_x, i_x)
    new_y <- c(new_y, i_y)
  }
  df <- data.frame(x=new_x, y=new_y)
  return(df)
}
```

Now everything is prepared and we understand the data sufficiently. We will start to answer the two questions that SeventhCloud had.

Question 1: What is the cost and revenue per 100 free and premium-version subscription?

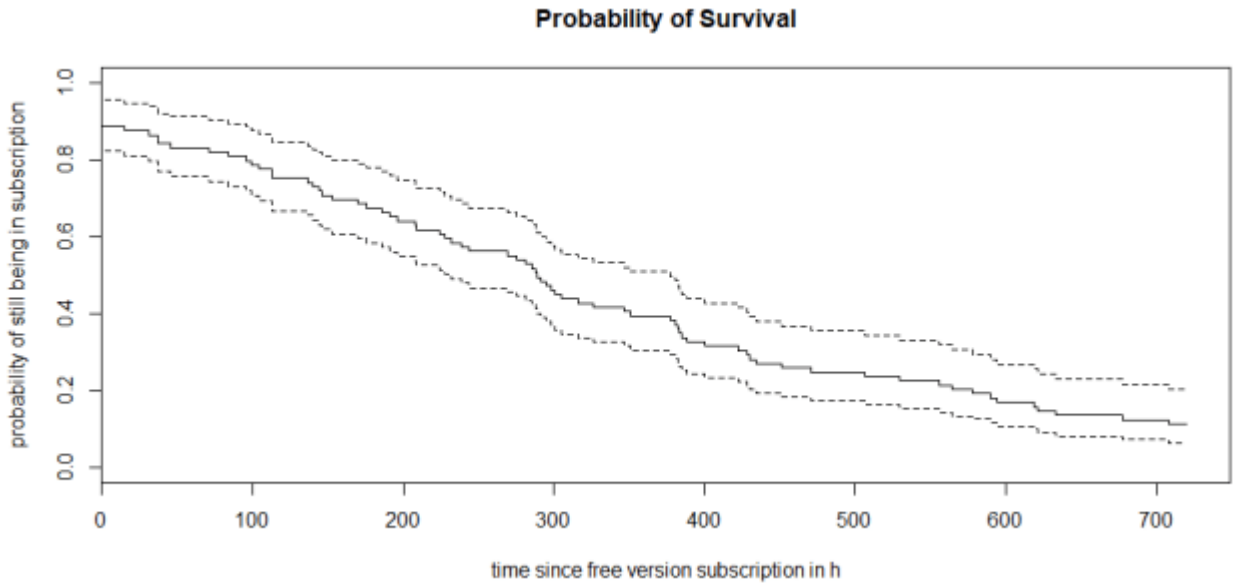
We will answer the first question in 5 steps.

1. Estimate the Kaplan Meier curve for the costs scenario
2. Calculate total monthly costs per 100 free subscriptions based on Kaplan Meier curve
3. Estimate the Kaplan Meier curve for the revenue scenario
4. Calculate total monthly revenue per 100 free subscriptions based on Kaplan Meier curve
5. Calculate monthly profit per 100 free subscriptions

Step 1: Estimate the Kaplan Meier curve for the costs scenario

For the first step, we first have to create a survival-object using the survival package. Based on the survival-object, we can easily compute the Kaplan Meier curve.

```
####### Question 1: What is the cost and revenue per 100 free and
premium-version subscription?

### Step 1
# Creating a survival object for right-censored data
SurvSub <- Surv(Subscription$duration, Subscription$censored=="no")
SurvSub # plus indicates right censoring



KaplanMeier <- survfit(SurvSub ~ 1)
KaplanMeier

summary(KaplanMeier)
plot(SurvSub,
     main ="Probability of Survival",
     xlab="time since free version subscription in h",
     ylab="probability of still being in subscription")
```



The Kaplan Meier curve is a step-function showing us the estimated probability of survival for a given point in time. The "survival" in our case means the probability that the person has not unsubscribed because we will need to pay the same cost for people regardless of whether they are free subscribers or premium subscribers.

Step 2: Calculate total monthly costs per 100 free subscriptions based on Kaplan Meier curve

If we multiply the probability for a given point in time that a person is still subscribed with 100, then we get the expected number of survivors of 100 initial subscriptions. If we multiply this number with the variable costs per hour and add these up, we would get the expected variable costs per 100 subscriptions. Now we only need to add the fixed costs.

```
### Step 2
str(KaplanMeier)
y <- KaplanMeier$surv
x <- KaplanMeier$time
cost_surv <- step_approx(x,y)


tcost_per_h <- (20/24)*100
fixcost <- 100000
tcost100 <- sum(cost_surv$y) * tcost_per_h + fixcost # because
stepwise function
```

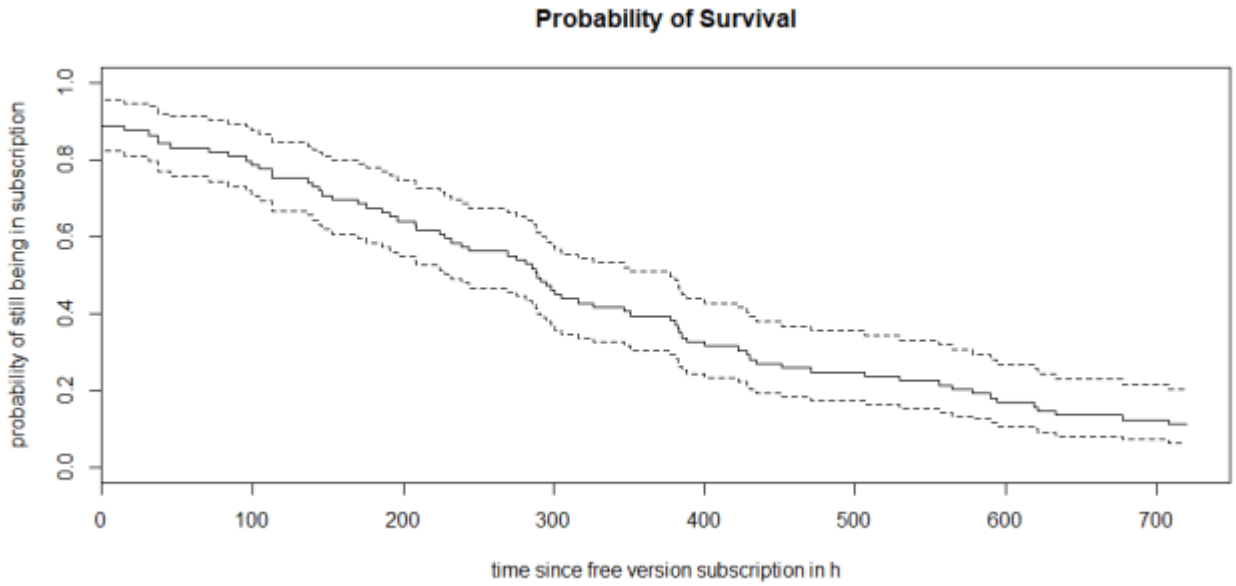
```
tcost100
```

According calculation at this step, the result is 126,500.90 € for the total cost for every 100 subscriptions.

Step 3: Estimate the Kaplan Meier curve for the revenue scenario

Now we repeat the first step with the difference, that we do it for the revenue scenario. This time, we only get the revenue from the ones that have a premium subscription and here comes the trick that I have mentioned earlier for why we can use two ending events.

```
### Step 3
NotCensored <- subset(Subscription, censored == "no")
SurvSub2 <- Surv(NotCensored$duration,
NotCensored$subscription=="yes")
SurvSub2 # plus indicates right censoring


KaplanMeier2 <- survfit(SurvSub2 ~ 1)
KaplanMeier2

summary(KaplanMeier2)
plot(SurvSub,
     main ="Probability of Survival",
     xlab="time since free version subscription in h",
     ylab="probability of still being in subscription")
```



Step 4: Calculate the total monthly revenue per 100 free subscriptions based on Kaplan Meier curve

Using the Kaplan Meier curve from step 3, we can now calculate the expected monthly revenue per 100 free subscriptions.

```
### Step 4
str(KaplanMeier2)
y2 <- KaplanMeier2$surv
x2 <- KaplanMeier2$time
rev_surv <- step_approx(x2,y2)

trev_per_h <- (100/24)*100
trev100 <- sum(rev_surv$y) * trev_per_h # because stepwise function
```

```
trev100
```

The code returns me an overall revenue of 198,954.80 €. Now we can compute the profit.
Step 5: Calculate monthly profit per 100 free subscriptions
Finally, we just calculate the profit using the results from step 2 and step 3.

```
### Step 5
profit <- trev100 - tcost100
```

```
profit
```

Now we can summarize the result for SeventhCloud and say, that the expected costs per 100 free subscriptions are 126,500.90 €, the expected revenue is 198,954.80 €, and the profit is 72,453.82 €. Based on this information, they can conclude that the freemium model is working on them. Of course, the calculation I have done was simplified. However, it is the basis for more advanced and complex estimation. It would be for instance possible, to use probability density functions instead of a step function and calculate the total cost, revenue and profit per 100 subscriptions for an unlimited time. It would also be possible to include tax considerations, use more complex cost structures than this simplified procedure. You can also calculate the net present value based on the very same procedure I just showed you.

Question 2: How can we improve the freemium model and increase the premium subscription rate?

The second question can be answered in two steps. First, we will use the Cox hazard regression model to identify differences between groups, possible causal factors influencing the time that a person will take to decide for premium subscription or unsubscription and to derive hypothesis for the future development. In the second step, we will calculate other possible scenarios based on the first question.

Step 1: Choose different variables and compute Cox regression

Besides the necessary variables, we have further variables which we can use to investigate their relationship and the duration until subscription or unsubscription. We will do it here for the case until the premium subscription to find possible relationships and formulate hypotheses on how to make people sign up faster for the premium offer.

```
####### Question 2: How can we improve the freemium model and increase
premium subscription rate?
## Step 1: How to reduce increase premium subscription?
coxfit2 <- coxph(SurvSub2 ~ NotCensored$prevcustomer
                 + NotCensored$appdays
                 + NotCensored$industry
                 + NotCensored$size, method = "breslow")
```

```
coxfit2
```

This gives us the following output:

Call:
coxph(formula = SurvSub2 ~ NotCensored$prevcustomer + NotCensored$appdays +
NotCensored$industry + NotCensored$size, method = "breslow")

	coef	exp(coef)	se(coef)	z	p
NotCensored$prevcustomeryes	0.0859	1.0897	0.6661	0.13	0.90
NotCensored$appdays	0.0738	1.0766	0.0525	1.41	0.16
NotCensored$industryConsulting	-0.6299	0.5326	1.1414	-0.55	0.58
NotCensored$industryFinance	-0.4750	0.6219	1.0779	-0.44	0.66
NotCensored$industryFood	-1.0413	0.3530	1.0734	-0.97	0.33
NotCensored$industryHealth	-1.3304	0.2644	1.1265	-1.18	0.24
NotCensored$industryIT	-0.1096	0.8962	0.5094	-0.22	0.83
NotCensored$industryManufacturing	-1.4925	0.2248	0.6873	-2.17	0.03
NotCensored$industryTelecommunication	-0.9828	0.3743	1.0980	-0.90	0.37
NotCensored$size1001+	-0.2467	0.7814	0.5421	-0.46	0.65
NotCensored$size51-1000	0.4136	1.5122	0.5390	0.77	0.44

Likelihood ratio test=11.97 on 11 df, p=0.4
n= 79, number of events= 26

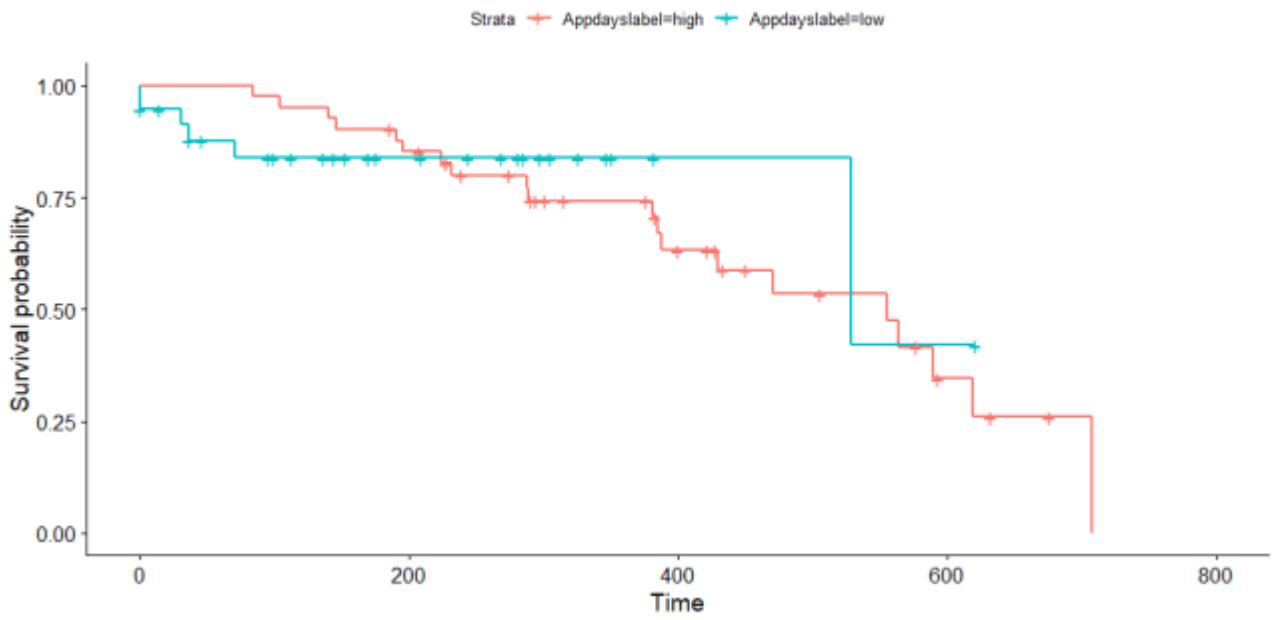
Now that is a typical output of the Cox regression model, but before we start to interpret it, we need to remember that it is not a survival but a hazard function. That means that the output of the model is relative risk with the form $h(t;x) = h_o(t)e^{\beta x}$, where $h_o(t)$ is the baseline hazard, x is a covariate and β its parameter. As we can see the natural exponent $e^{\beta x}$, it means that we cannot just take estimate coefficients and interpret them directly. By each unit increase of x, the baseline hazard increases by $e^{\beta \Delta x}$. Furthermore, we have the p-values which quantify the uncertainty or precision of our estimates. The general rule of thumb is that if the p-value is below 0.05, then we have a significant effect and we can assume that there is a relationship.

It is also important here to differentiate between statistical significance and economic significance. Statistical significance only helps us to identify whether there might be any relationship between the y-variable and x-variables. However, it does not tell us whether the variable is relevant. If the effect of the variable or its influence on the y variable is very little despite the statistical significance then the economic significance is low. Imagine we would have a significant variable that increases the baseline by 0.00001% or another one that

would increase the baseline hazard by 30%. Then the former one would be far less economically significant.
Now in our case, we detect two interesting things. First, we can formulate the hypothesis that the industry matters, as one level differs significantly from the reference level. The more interesting question that should be addressed in the future is why it differs? Does it differ because our products cover the need of certain industries better or because there is a lack of competition? The second interesting thing we can observe is the variable appdays, which is not significant. Nevertheless, we put it on the agenda, because there might be another interesting relationship. We formulate the hypothesis that the high-group is more likely to sign up. For further investigation, we create another plot differentiating between the people that have used the app only little (appdays low) vs. the people that have used the app a lot (appdays high).

```
# Create two KapplanMeier Models for both subsgroups
NotCensored$Appdayslabel <- ifelse(NotCensored$appdays >=
median(NotCensored$appdays), "high", "low")
appdaysSurv <- Surv(NotCensored$duration,
NotCensored$subscription=="yes")

appdaysFit <- survfit(appdaysSurv ~ Appdayslabel, data=NotCensored)

ggsurvplot(appdaysFit, data=NotCensored)
```

```
table(NotCensored$Appdayslabel)
```



high low
41 38

The shape seems to be very interesting and the groups seem to be balanced, which is very important in order to to be able to make adequate conclusions and generalizations. The interesting thing we observe first is that there is a sudden drop at days XX for the low-group, while the survival chance of the high-group decreases rather continuously. The question that might come up here is also why is there a sudden drop? A possible hypothesis is that the low-group and high-group might have different needs.

The rough analysis steps just shown can also be done in greater depth with further methods and with greater care when picking possible hypothesis. You can also do the same procedure to investigate the hazard function for unsubscription to find out depending on your strategy how to a) either increase free subscriptions or b) decrease free subscriptions depending on the strategy.

Step 2: Calculate different scenarios (profit margin and comparison)

Given the hypothesis and conclusions from the first step, you could now do different simulations to calculate a business case for this scenario using the procedure presented to you in question 1. In fact, we can simplify everything into a simple function:

```
### Step 2
freemium_profit <- function(df) {
  SurvSub <- Surv(df$duration, df$censored=="no")
  KaplanMeier <- survfit(SurvSub ~ 1)

  ### Step 2 from Q1
  y <- KaplanMeier$surv
  x <- KaplanMeier$time
  cost_surv <- step_approx(x,y)
  tcost_per_h <- (20/24)*100
  fixcost <- 100000
  tcost100 <- sum(cost_surv$y) * tcost_per_h + fixcost # because
stepwise function
  print(paste("Total cost is", round(tcost100,digits = 2),"EUR"))
  ### Step 3 from Q1
  NotCensored <- subset(Subscription, censored == "no")
  SurvSub2 <- Surv(NotCensored$duration,
NotCensored$subscription=="yes")
  KaplanMeier2 <- survfit(SurvSub2 ~ 1)
  ### Step 4 from Q1
  y2 <- KaplanMeier2$surv
  x2 <- KaplanMeier2$time
  rev_surv <- step_approx(x2,y2)
  trev_per_h <- (100/24)*100
  trev100 <- sum(rev_surv$y) * trev_per_h # because stepwise function
  print(paste("Total revnue is", round(trev100,digits = 2),"EUR"))

  ### Step 5 from Q1
  profit <- trev100 - tcost100
  print(paste("Total profit is", round(profit,digits = 2),"EUR"))
```

```
}
```

For demonstration purpose, we want to investigate what would be the expected potential profit if we had only the low-group from step 1 and only the high-group from step 2.

```
LowAppdays <- subset(Subscription, appdays < median(appdays))
HighAppdays <- subset(Subscription, appdays >= median(appdays))

freemium_profit(LowAppdays)
```

```
freemium_profit(HighAppdays)
```

freemium_profit(LowAppdays)
[1] "Total cost is 112905.7 EUR"
[1] "Total revnue is 198954.76 EUR"
[1] "Total profit is 86049.05 EUR"
> freemium_profit(HighAppdays)
[1] "Total cost is 129630.72 EUR"
[1] "Total revnue is 198954.76 EUR"
[1] "Total profit is 69324.04 EUR"

Now we can see that the profit for the low-scenario would be 86,039.05 € and for the high-scenario 69,324.04 €. Given the results and if the second hypothesis holds true, we might be able to increase the profit per month per 100 subscriptions by focusing on the low-appdays users. Of course there are more considerations to be made, but using this methodology it is possible to simulate different scenarios.

Conclusion: Survival & hazard models are powerful tools for the freemium model

There are still three general remarks to be made on the freemium model. First, the methodology shown fits regardless of the strategy. And a very important takeaway message from the story and methodology shown is that it helps solve a very important problem of the freemium model:

1. find the right balance between offering a rich attractive free subscription offer, that drives premium subscription and
2. not offering too much in the free subscription offer, because a comprehensive free version might keep customers from signing up for the premium version.

The methodology presented for answering the second question helps you to navigate to the "right" balance and further optimize the offers.

Second, of course, you could compute the conversion rate in the traditional way, but it will be biased. Biasedness means it will not be precise and it will not reflect the true state of the conversion rate because of the right censoring of our data. Therefore, you will derive a more precise conversion rate by using the results from a survival model. Sometimes, the classical conversion rate will not be far off, but there is no guarantee for that.

Third, a common problem that freemium model users face is that the subscriptions will start to flatten at some point. At this point, many companies pivot away to a limited freemium version with a 30-day free trial or completely abandon it. The methodology shown can tell you in advance when a pivot might be needed by simply tracking the profit for the first

month per 100 new subscriptions and the number of monthly subscriptions. If any of these two metrics starts dropping, then you should analyze the reason and take the right actions to sustain the business.

Now, what is the story? SeventhCloud has created a good service that seems to be appealing to new clients especially from the area of manufacturing. Their business will continue to grow and they will make a great profit. If SeventhCloud specialized in manufacturing and increases quality as well as the frequency with which the subscribers interact with the platform, they can even achieve greater growth with the freemium model.