HW_03

Anahit Navoyan

2023-11-30

```r
library(ggplot2)
library(survival)
```

```
## Warning: package 'survival' was built under R version 4.2.3
```

```r
data = read.csv('telco.csv')
head(data)
```

```
##   ID region tenure age   marital address income                         ed
## 1  1 Zone 2     13  44   Married       9     64             College degree
## 2  2 Zone 3     11  33   Married       7    136    Post-undergraduate degree
## 3  3 Zone 3     68  52   Married      24    116 Did not complete high school
## 4  4 Zone 2     33  33 Unmarried      12     33         High school degree
## 5  5 Zone 2     23  30   Married       9     30 Did not complete high school
## 6  6 Zone 2     41  39 Unmarried      17     78         High school degree
##   retire gender voice internet forward       custcat churn
## 1     No   Male    No       No     Yes Basic service   Yes
## 2     No   Male   Yes       No     Yes Total service   Yes
## 3     No Female    No       No      No  Plus service    No
## 4     No Female    No       No      No Basic service   Yes
## 5     No   Male    No       No     Yes  Plus service    No
## 6     No Female    No       No      No  Plus service    No
```

```r
data$churn=ifelse(data$churn=='Yes',1,0)
y = data['churn']
valid_columns = colnames(data)[c(-1,-3,-15)]
x = data[valid_columns]
```

```r
# Print names of all available distributions
all_distributions <- survreg.distributions
distributions <- names(all_distributions)
print(distributions)
```

```
##  [1] "extreme"     "logistic"    "gaussian"    "weibull"     "exponential"
##  [6] "rayleigh"    "loggaussian" "lognormal"   "loglogistic" "t"
```

```r
surv_obj = Surv(time = data$tenure, event = data$churn)
```

```r
distribution_names = c()
loglikelihoods = c()
aics = c()
bics = c()
regression_models = c()

for (distribution in all_distributions) {
  reg_m = survreg(surv_obj~., dist = distribution, data = x)

  # Model fit information
  print('.....................')
  print(distribution$name)
  print(reg_m$loglik)
  print(extractAIC(reg_m)[2])
  print(BIC(reg_m))

  regression_models = c(regression_models, reg_m)
  distribution_names = c(distribution_names, distribution$name)
  aics = c(aics,extractAIC(reg_m)[2])
  bics = c(bics,BIC(reg_m))
  loglikelihoods = c(loglikelihoods,reg_m$loglik[2])
}
```

```
## [1] "....................."
## [1] "Extreme value"
## [1] -1747.194 -1571.191
## [1] 3182.381
## [1] 3280.536
## [1] "....................."
## [1] "Logistic"
## [1] -1734.223 -1554.948
## [1] 3149.896
## [1] 3248.051
## [1] "....................."
## [1] "Gaussian"
## [1] -1714.485 -1547.611
## [1] 3135.221
## [1] 3233.376
## [1] "....................."
## [1] "Weibull"
## [1] -1606.431 -1462.172
## [1] 2964.343
## [1] 3062.498
## [1] "....................."
## [1] "Exponential"
## [1] -1606.980 -1467.598
## [1] 2973.195
## [1] 3066.442
## [1] "....................."
## [1] "Rayleigh"
## [1] -1739.723 -1527.438
## [1] 3092.877
## [1] 3186.124
```

```
## [1] "...................."
## [1] "Log Normal"
## [1] -1602.518 -1457.012
## [1] 2954.024
## [1] 3052.179
## [1] "...................."
## [1] "Log Normal"
## [1] -1602.518 -1457.012
## [1] 2954.024
## [1] 3052.179
## [1] "...................."
## [1] "Log logistic"
## [1] -1605.208 -1458.103
## [1] 2956.206
## [1] 3054.361
## [1] "...................."
## [1] "Student-t"
## [1] -1748.062 -1562.957
## [1] 3165.914
## [1] 3264.069
```

```r
print(distribution_names[which.max(loglikelihoods)])
```

```
## [1] "Log Normal"
```

```r
print(distribution_names[which.min(aics)])
```

```
## [1] "Log Normal"
```

```r
print(distribution_names[which.min(bics)])
```

```
## [1] "Log Normal"
```

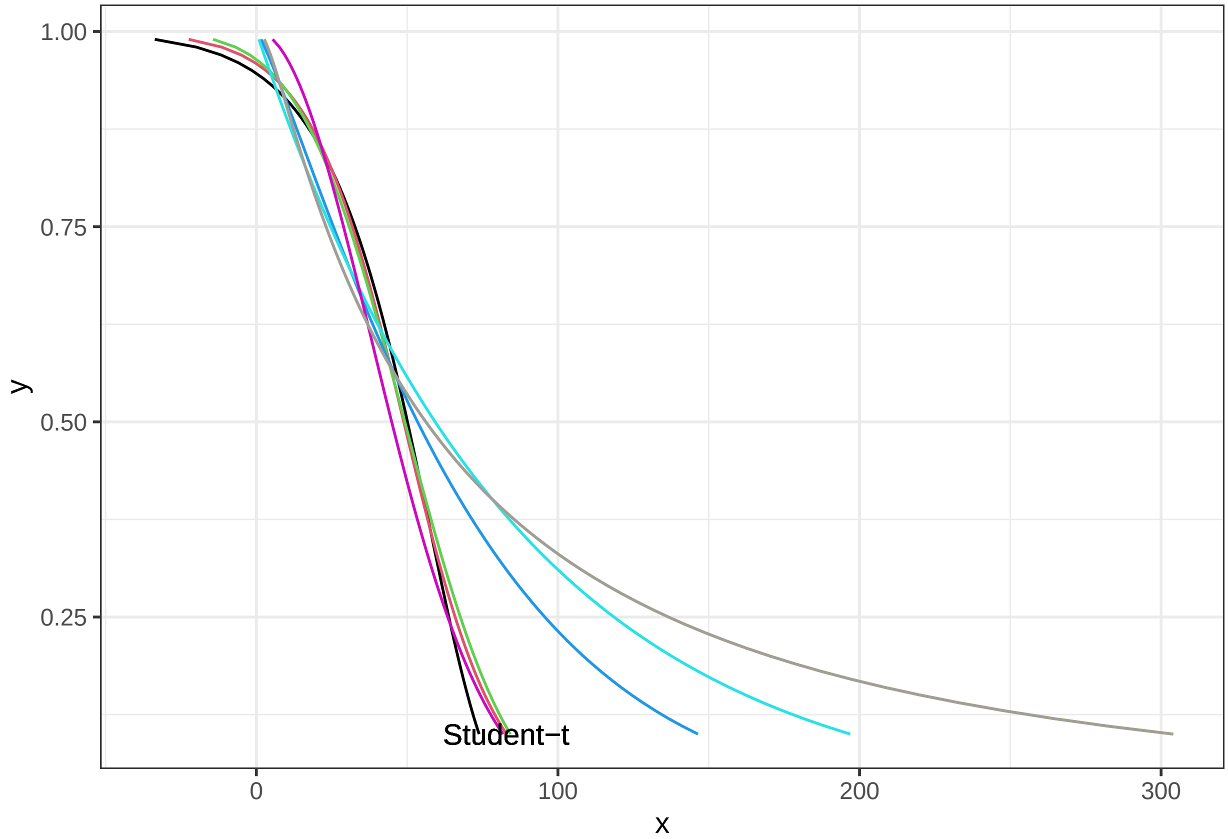
Taking the model with highest loglikelihood and lowest AIC and BIC score. The results show that the Log Normal model is a best fit

Visualize all the curves: one plot for all

```r
pct <- 1:90/100
all_predictions = matrix(ncol = length(pct))

for (distribution in distributions){
  reg_m = survreg(surv_obj~., dist = distribution, data = x)
  ptime <- predict(reg_m, type='quantile', p = pct)
  all_predictions = rbind(all_predictions, x = ptime[1, ])
}

all_predictions = all_predictions[2:11,1:90]
pal = palette(rainbow(n = 10))
```

```r
p <- ggplot()

for (i in c(1:length(distributions))){
  p <- p + geom_line(aes_string(x = all_predictions[i,1:90], y = 1-pct), color = pal[i], group = distri
    geom_text(aes(x = all_predictions[[i,c(90)]], y = (1-pct)[90], label = paste(distribution_names[[i]
}

print(p)
```

```
## Warning: Removed 90 row(s) containing missing values (geom_path).
## Removed 90 row(s) containing missing values (geom_path).
```



The model with the Log Normal distribution is the best model because it has the lowest AIC and BIC and it has the highest log-likelihood. Therefore, it has a better fit on the original data.

Keep significant features

```r
best_model <- survreg(surv_obj ~ ., dist = "lognormal", data = x)

significant_features <- rownames(summary(best_model)$table)[summary(best_model)$table[, 4] < 0.05]

significant_features <- c("age", "address", "voice", "custcat", "marital", "internet")
```

```r
final_model <- survreg(surv_obj ~ ., dist = "lognormal", data = x[significant_features])

summary(final_model)
```

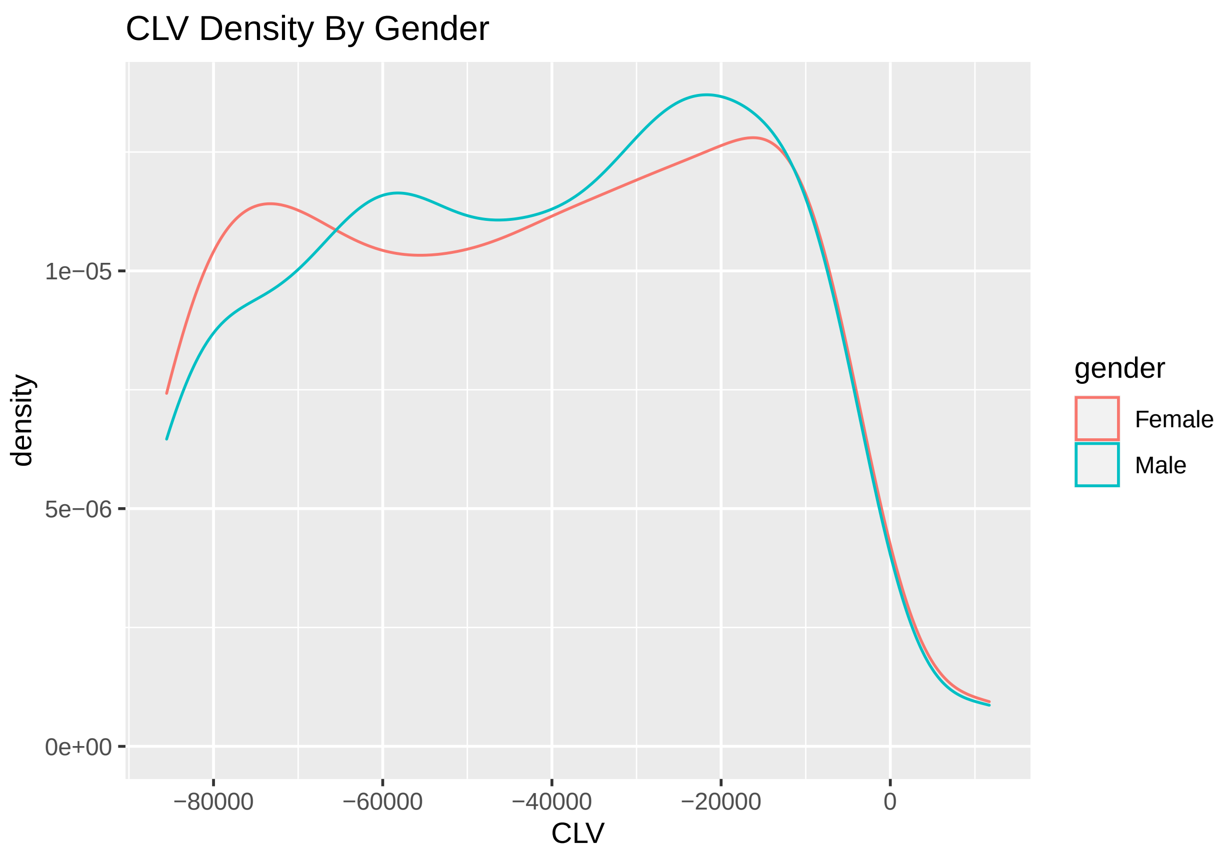
```
##
## Call:
## survreg(formula = surv_obj ~ ., data = x[significant_features],
##     dist = "lognormal")
##                          Value Std. Error     z       p
## (Intercept)            2.53488    0.24261 10.45 < 2e-16
## age                    0.03683    0.00640  5.75 8.7e-09
## address                0.04282    0.00885  4.84 1.3e-06
## voiceYes              -0.46350    0.16677 -2.78  0.0054
## custcatE-service       1.02582    0.16905  6.07 1.3e-09
## custcatPlus service    0.82250    0.16942  4.85 1.2e-06
## custcatTotal service   1.01326    0.20958  4.83 1.3e-06
## maritalUnmarried      -0.44732    0.11447 -3.91 9.3e-05
## internetYes           -0.84054    0.13826 -6.08 1.2e-09
## Log(scale)             0.28303    0.04602  6.15 7.7e-10
##
## Scale= 1.33
##
## Log Normal distribution
## Loglik(model)= -1462.1   Loglik(intercept only)= -1602.5
##  Chisq= 280.83 on 8 degrees of freedom, p= 4.9e-56
## Number of Newton-Raphson Iterations: 5
## n= 1000
```

CLV

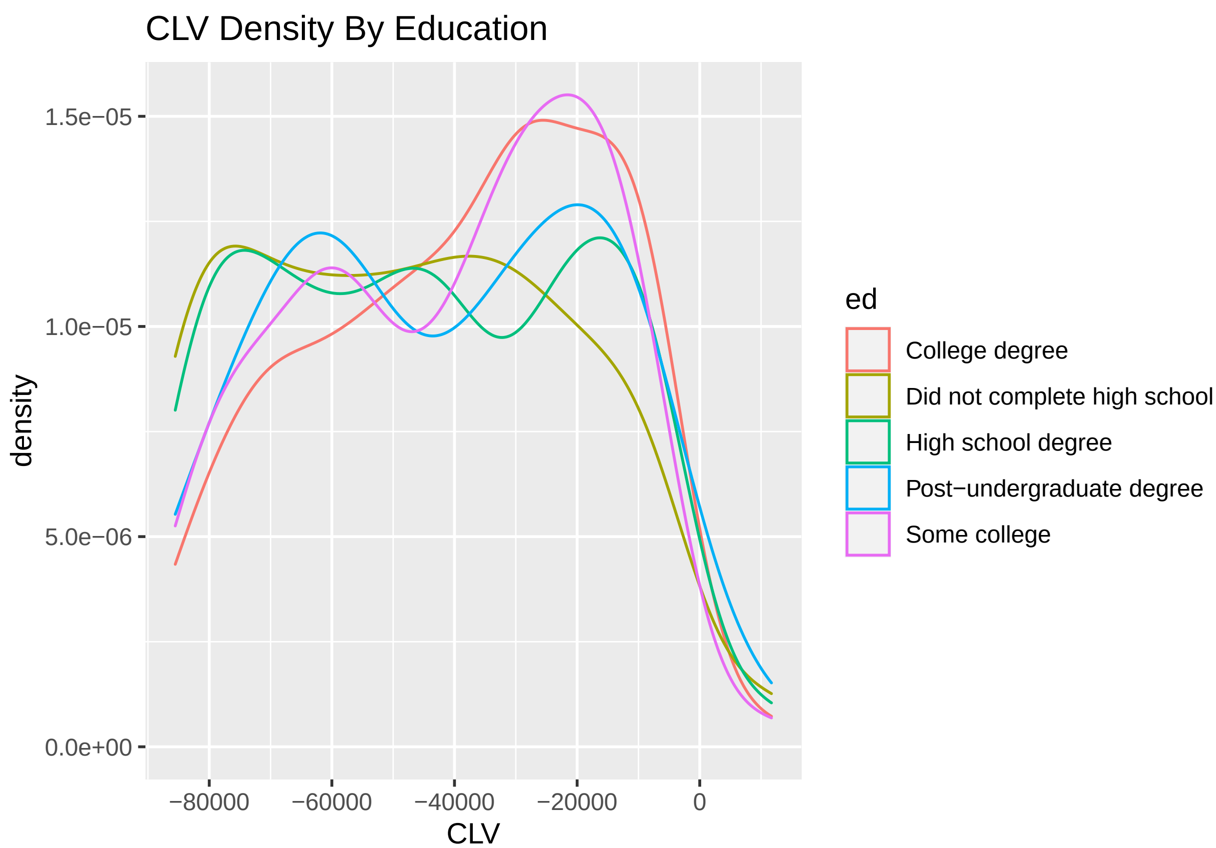
```r
pred <- predict(final_model, newdata = data, type = "response")
# list.tree(pred)

pred_data <- data.frame(surv = pred)
```

```r
average_margin_MM <- 1300
discount_rate_r <- 0.1
retention_rate <-
tenure <- data$tenure

# Calculate CLV
data$CLV <- (average_margin_MM * (1 - discount_rate_r) * retention_rate) / (1 + discount_rate_r - reten
```

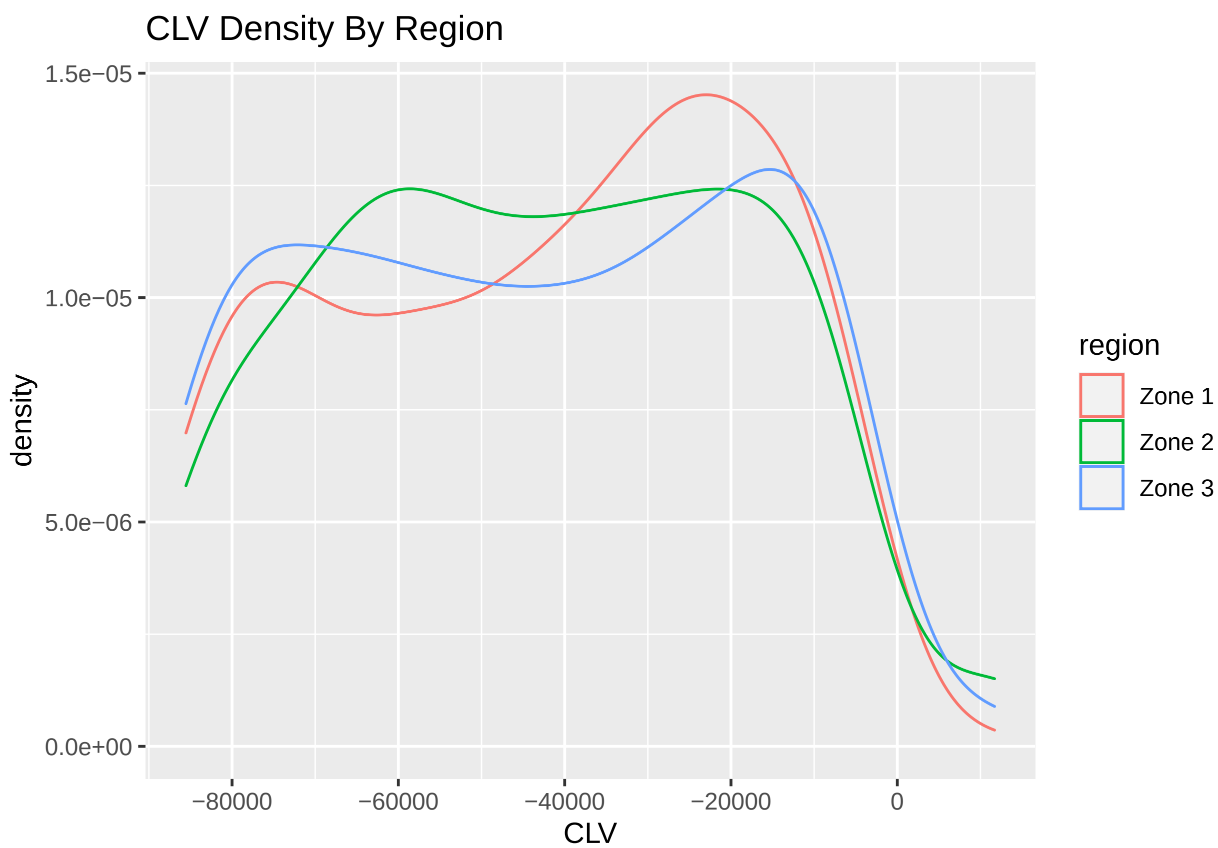
```r
#CLV Density By Gender

ggplot(data, aes(x=CLV, color=gender))+
labs(title = "CLV Density By Gender")+
geom_density()
```




```
#CLV Density By Education

ggplot(data,aes(x=CLV, color=ed))+
labs(title = "CLV Density By Education")+
geom_density()
```



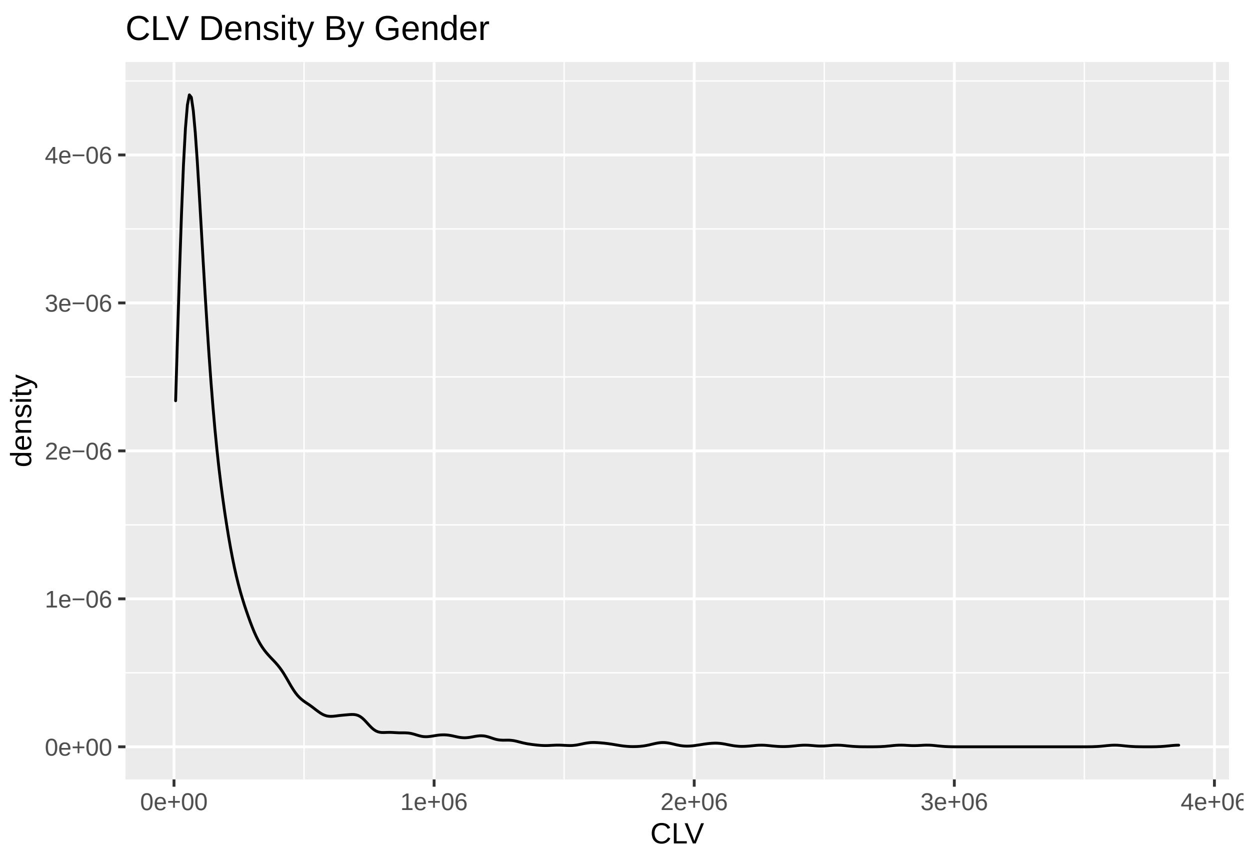

```r
#CLV Density By Region

ggplot(data,aes(x=CLV, color=region))+
labs(title = "CLV Density By Region")+
geom_density()
```



Retention

```
sequence = seq(1,length(colnames(pred_data)),1)
MM = 1300
r = 0.1
for (num in sequence) {
pred_data[,num]=pred_data[,num]/(1+r/12)^(sequence[num]-1)
}
pred_data$CLV=MM*rowSums(pred_data)
data$CLV = pred_data$CLV
```

```
#CLV Density By Gender

ggplot(data, aes(x=CLV)) + labs(title = "CLV Density By Gender")+
  geom_density()
```



Report

```r
summary(final_model)
```

```
##
## Call:
## survreg(formula = surv_obj ~ ., data = x[significant_features],
##     dist = "lognormal")
##                         Value Std. Error     z       p
## (Intercept)           2.53488    0.24261 10.45 < 2e-16
## age                   0.03683    0.00640  5.75 8.7e-09
## address               0.04282    0.00885  4.84 1.3e-06
## voiceYes             -0.46350    0.16677 -2.78  0.0054
## custcatE-service      1.02582    0.16905  6.07 1.3e-09
## custcatPlus service   0.82250    0.16942  4.85 1.2e-06
## custcatTotal service  1.01326    0.20958  4.83 1.3e-06
## maritalUnmarried     -0.44732    0.11447 -3.91 9.3e-05
## internetYes          -0.84054    0.13826 -6.08 1.2e-09
## Log(scale)            0.28303    0.04602  6.15 7.7e-10
##
## Scale= 1.33
##
## Log Normal distribution
```

```
## Loglik(model)= -1462.1   Loglik(intercept only)= -1602.5
##  Chisq= 280.83 on 8 degrees of freedom, p= 4.9e-56
## Number of Newton-Raphson Iterations: 5
## n= 1000
```

The distribution lognormal was chosen as it has the highest loglikelyhood and the lowest AIC and BIC score. Overall p-value of the model indicates that the model is statistically significant and is a good fit.

The positive coefficients for age, address, custcatE-service, custcatPlus service, and custcatTotal service suggest that older individuals are less prone to churn. Customers who have not chosen the basic service are also less likely to churn. On the contrary, the negative coefficients for maritalUnmarried, VoiceYes, and internetYes imply that customers with internet and voice services show a lower survival rate. Furthermore, being unmarried increases the likelihood of churn among customers.

Important segments are the segments with higher CLV than the other groups. For example from the visualizations we can conlude that males with some college education and from zone 1 has the highest CLV.