

# DS223\_HW3

Alla Khojayan

2025-11-07

```
if (!requireNamespace("pacman", quietly = TRUE)) install.packages("pacman")
# a cool library i learnt about not too long ago, it manages installation and
# loading of packages automatically
pacman::p_load(survival, survminer, flexsurv, dplyr, ggplot2,
broom, tibble, stringr, wesanderson)
```

```
telco <- read.csv("telco.csv", stringsAsFactors = FALSE)
head(telco)
```

```
##   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
```

```
telco$churn <- ifelse(telco$churn %in% c(1,"Yes","yes","TRUE",TRUE), 1, 0)
```

Factorising the categories:

```
telco$region <- factor(telco$region)
telco$marital <- factor(telco$marital)
telco$ed <- factor(telco$ed)
telco$retire <- factor(telco$retire)
telco$gender <- factor(telco$gender)
telco$voice <- factor(telco$voice)
telco$internet <- factor(telco$internet)
telco$forward <- factor(telco$forward)
telco$custcat <- factor (telco$custcat)

surv_obj <- Surv(time = telco$tenure, event = telco$churn)
```

Finding available distributions

```
names(survreg.distributions)
```

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

```
aft_formula <- surv_obj ~ region + age + marital + address + income + ed +
  retire + gender + voice + internet + forward + custcat
```

```
reg_extreme      <- try(survreg(aft_formula, data = telco, dist = "extreme"))
reg_logistic     <- try(survreg(aft_formula, data = telco, dist = "logistic"))
reg_gaussian     <- try(survreg(aft_formula, data = telco, dist = "gaussian"))
reg_weibull      <- try(survreg(aft_formula, data = telco, dist = "weibull"))
reg_exponential  <- try(survreg(aft_formula, data = telco, dist = "exponential"))
reg_rayleigh     <- try(survreg(aft_formula, data = telco, dist = "rayleigh"))
reg_loggaussian  <- try(survreg(aft_formula, data = telco, dist = "loggaussian"))
reg_lognormal    <- try(survreg(aft_formula, data = telco, dist = "lognormal"))
reg_loglogistic  <- try(survreg(aft_formula, data = telco, dist = "loglogistic"))
reg_t            <- try(survreg(aft_formula, data = telco, dist = "t"))
```

```
dists_vec <- c("extreme", "logistic", "gaussian", "weibull", "exponential",
  "rayleigh", "loggaussian", "lognormal", "loglogistic", "t")
```

```
fits_vec <- list(reg_extreme, reg_logistic, reg_gaussian, reg_weibull, reg_exponential,
  reg_rayleigh, reg_loggaussian, reg_lognormal, reg_loglogistic, reg_t)
```

```
AIC_vec <- rep(NA_real_, length(fits_vec))
BIC_vec <- rep(NA_real_, length(fits_vec))
LL_vec  <- rep(NA_real_, length(fits_vec))
n_sig   <- rep(NA_integer_, length(fits_vec))
ok_vec  <- rep(FALSE, length(fits_vec))
```

```
for (i in seq_along(fits_vec)) {
  f <- fits_vec[[i]]
  if (!inherits(f, "try-error")) {
    ok_vec[i] <- TRUE
    AIC_vec[i] <- AIC(f)
    BIC_vec[i] <- BIC(f)
    LL_vec[i] <- as.numeric(logLik(f))
    st <- summary(f)$table
    if (!is.null(st) && "p" %in% colnames(st)) {
      # counting how many p < 0.05 excluding the intercept
      pvals <- st[, "p"]
      if (length(pvals) > 1) n_sig[i] <- sum(pvals[-1] < 0.05, na.rm = TRUE)
    }
  }
}
```

```
comparison <- data.frame(
  dist = dists_vec, ok = ok_vec,
  AIC = AIC_vec, BIC = BIC_vec, logLik = LL_vec, n_sig = n_sig,
  row.names = NULL)
```

```
comparison[order(comparison$AIC), ]
```

	dist	ok	AIC	BIC	logLik	n_sig
## 7	loggaussian	TRUE	2954.024	3052.179	-1457.012	9
## 8	lognormal	TRUE	2954.024	3052.179	-1457.012	9
## 9	loglogistic	TRUE	2956.206	3054.361	-1458.103	11
## 4	weibull	TRUE	2964.343	3062.498	-1462.172	11
## 5	exponential	TRUE	2973.195	3066.442	-1467.598	10
## 6	rayleigh	TRUE	3092.877	3186.124	-1527.438	12
## 3	gaussian	TRUE	3135.221	3233.376	-1547.611	12
## 2	logistic	TRUE	3149.896	3248.051	-1554.948	10
## 10	t	TRUE	3165.914	3264.069	-1562.957	10
## 1	extreme	TRUE	3182.381	3280.536	-1571.191	10

```
ref <- telco[1, , drop = FALSE]
for (nm in names(ref)) {
  if (is.numeric(telco[[nm]])) {
    ref[[nm]] <- median(telco[[nm]], na.rm = TRUE)
  } else if (is.factor(telco[[nm]])) {
    ref[[nm]] <- names(sort(table(telco[[nm]]), decreasing = TRUE))[1]}
}

surv_levels <- seq(0.9, 0.1, by = -0.1)
curve_df <- data.frame()
```

Weibull

```
if (!inherits(reg_weibull, "try-error")) {
  qt_weib <- predict(reg_weibull, type = "quantile", p = 1 - surv_levels, newdata = ref)
  df_weib <- data.frame(dist = "weibull", Time = as.numeric(qt_weib), Survival = surv_levels)
  curve_df <- rbind(curve_df, df_weib)}
}
```

Lognormal

```
if (!inherits(reg_lognormal, "try-error")) {
  qt_lnorm <- predict(reg_lognormal, type = "quantile", p = 1 - surv_levels, newdata = ref)
  df_lnorm <- data.frame(dist = "lognormal", Time = as.numeric(qt_lnorm), Survival = surv_levels)
  curve_df <- rbind(curve_df, df_lnorm)}
}
```

LogLogistic

```
if (!inherits(reg_loglogistic, "try-error")) {
  qt_llog <- predict(reg_loglogistic, type = "quantile", p = 1 - surv_levels, newdata = ref)
  df_llog <- data.frame(dist = "loglogistic", Time = as.numeric(qt_llog), Survival = surv_levels)
  curve_df <- rbind(curve_df, df_llog)}
}
```

Extreme

```
if (!inherits(reg_extreme, "try-error")) {
  qt_llog <- predict(reg_extreme, type = "quantile", p = 1 - surv_levels, newdata = ref)
  df_llog <- data.frame(dist = "extreme", Time = as.numeric(qt_llog), Survival = surv_levels)
  curve_df <- rbind(curve_df, df_llog)}
}
```

Logistic

```
if (!inherits(reg_logistic, "try-error")) {
  qt_llog <- predict(reg_logistic, type = "quantile", p = 1 - surv_levels, newdata = ref)
  df_llog <- data.frame(dist = "logistic", Time = as.numeric(qt_llog), Survival = surv_levels)
  curve_df <- rbind(curve_df, df_llog)}
```

Gaussian

```
if (!inherits(reg_gaussian, "try-error")) {
  qt_llog <- predict(reg_gaussian, type = "quantile", p = 1 - surv_levels, newdata = ref)
  df_llog <- data.frame(dist = "gaussian", Time = as.numeric(qt_llog), Survival = surv_levels)
  curve_df <- rbind(curve_df, df_llog)}
```

Exponential

```
if (!inherits(reg_exponential, "try-error")) {
  qt_llog <- predict(reg_exponential, type = "quantile", p = 1 - surv_levels, newdata = ref)
  df_llog <- data.frame(dist = "exponential", Time = as.numeric(qt_llog), Survival = surv_levels)
  curve_df <- rbind(curve_df, df_llog)}
```

Rayleigh

```
if (!inherits(reg_rayleigh, "try-error")) {
  qt_llog <- predict(reg_rayleigh, type = "quantile", p = 1 - surv_levels, newdata = ref)
  df_llog <- data.frame(dist = "rayleigh", Time = as.numeric(qt_llog), Survival = surv_levels)
  curve_df <- rbind(curve_df, df_llog)}
```

LogGaussian

```
if (!inherits(reg_loggaussian, "try-error")) {
  qt_llog <- predict(reg_loggaussian, type = "quantile", p = 1 - surv_levels, newdata = ref)
  df_llog <- data.frame(dist = "loggaussian", Time = as.numeric(qt_llog), Survival = surv_levels)
  curve_df <- rbind(curve_df, df_llog)}
```

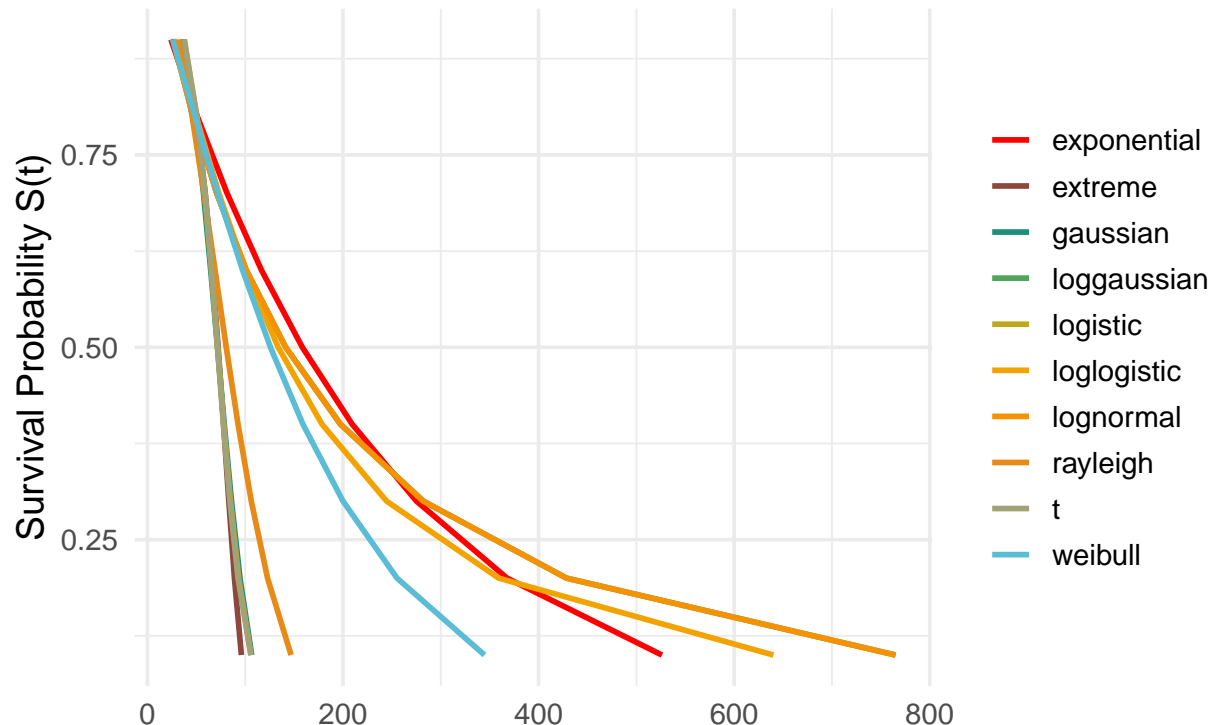
T

```
if (!inherits(reg_t, "try-error")) {
  qt_llog <- predict(reg_t, type = "quantile", p = 1 - surv_levels, newdata = ref)
  df_llog <- data.frame(dist = "t", Time = as.numeric(qt_llog), Survival = surv_levels)
  curve_df <- rbind(curve_df, df_llog)}
```

```
n_models <- length(unique(curve_df$dist))
wes_cols <- wes_palette("Darjeeling1", n_models, type = "continuous")

ggplot(curve_df, aes(x = Time, y = Survival, color = dist)) +
  geom_line(linewidth = 1) +
  scale_color_manual(values = wes_cols) +
  labs(title = "Survival Curves Across Distributions (Wes Anderson palette)",
       x = "", y = "Survival Probability S(t)") +
  theme_minimal(base_size = 14) +
  theme(
    legend.title = element_blank(),
    legend.position = "right")
```

## Survival Curves Across Distributions (Wes Anderson pa



Going back to the comparison table, we may see that the lowest AIC is yielded by the LogNormal and the LogGaussian, however for the CLV analysis, some literature suggested using the LogNormal method is better.

```
summary(reg_lognormal)
```

```
##
## Call:
## survreg(formula = aft_formula, data = telco, dist = "lognormal")
##
```

	Value	Std. Error	z	p
## (Intercept)	2.36227	0.29263	8.07	6.9e-16
## regionZone 2	-0.09704	0.14277	-0.68	0.497
## regionZone 3	0.04822	0.14154	0.34	0.733
## age	0.03267	0.00725	4.50	6.7e-06
## maritalUnmarried	-0.45515	0.11543	-3.94	8.0e-05
## address	0.04254	0.00890	4.78	1.8e-06
## income	0.00140	0.00092	1.52	0.129
## edDid not complete high school	0.37361	0.20159	1.85	0.064
## edHigh school degree	0.31593	0.16318	1.94	0.053
## edPost-undergraduate degree	-0.03436	0.22317	-0.15	0.878
## edSome college	0.27232	0.16535	1.65	0.100
## retireYes	0.02248	0.44407	0.05	0.960
## genderMale	0.05188	0.11429	0.45	0.650
## voiceYes	-0.43379	0.16895	-2.57	0.010
## internetYes	-0.77150	0.14348	-5.38	7.6e-08
## forwardYes	-0.19813	0.18004	-1.10	0.271

```
## custcatE-service          1.06642    0.17053    6.25 4.0e-10
## custcatPlus service       0.92495    0.21575    4.29 1.8e-05
## custcatTotal service      1.19860    0.25045    4.79 1.7e-06
## Log(scale)                0.27577    0.04600    6.00 2.0e-09
##
## Scale= 1.32
##
## Log Normal distribution
## Loglik(model)= -1457    Loglik(intercept only)= -1602.5
## Chisq= 291.01 on 18 degrees of freedom, p= 3.4e-51
## Number of Newton-Raphson Iterations: 5
## n= 1000
```

Here we are selecting the most appropriate features with the lowest yielded p-values.

```
final_formula <- surv_obj ~ age + marital + address + voice + internet + custcat
final_dist    <- "lognormal"
final_model    <- survreg(final_formula, data = telco, dist = final_dist)
summary(final_model)
```

```
##
## Call:
## survreg(formula = final_formula, data = telco, dist = final_dist)
##
##              Value Std. Error      z      p
## (Intercept)    2.53488    0.24261 10.45 < 2e-16
## age            0.03683    0.00640   5.75 8.7e-09
## maritalUnmarried -0.44732    0.11447  -3.91 9.3e-05
## address         0.04282    0.00885   4.84 1.3e-06
## voiceYes        -0.46350    0.16677  -2.78 0.0054
## internetYes     -0.84054    0.13826  -6.08 1.2e-09
## 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
## 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
```

```
fs_fit <- flexsurv::flexsurvreg(final_formula, data = telco, dist = final_dist)
```

```
times <- 1:24
MM <- 1300
r <- 0.10
disc <- 1 / (1 + r/12)^(times - 1)

n <- nrow(telco)
S_mat <- matrix(NA_real_, nrow = n, ncol = length(times))
pb <- txtProgressBar(min = 0, max = n, style = 3)
```

```
## |
```

```
for (i in seq_len(n)) {
  s_obj <- try(
    summary(fs_fit,
            newdata = telco[i, , drop = FALSE],
            type = "survival", t = times),
    silent = TRUE)
  row_vals <- rep(NA_real_, length(times))
  if (!inherits(s_obj, "try-error") && !is.null(s_obj)) {
    if (is.list(s_obj)) {
      row_vals <- vapply(s_obj, function(d) d$est[1], numeric(1))
    } else if (is.data.frame(s_obj) && "est" %in% names(s_obj)) {
      row_vals <- s_obj$est
    }
  }
  if (length(row_vals) < length(times)) {
    row_vals <- c(row_vals, rep(NA_real_, length(times) - length(row_vals)))
  }
  row_vals[is.na(row_vals)] <- 0
  row_vals[row_vals < 0] <- 0
  row_vals[row_vals > 1] <- 1
  S_mat[i, ] <- row_vals
  setTxtProgressBar(pb, i)
}
```

```
## |
```

```
close(pb)
```

```
#  $CLV_i = MM * \sigma_t S_i(t) * discount_t$ 
CLV <- numeric(n)
for (i in seq_len(n)) {
  CLV[i] <- MM * sum(S_mat[i, ] * disc)
}
telco$CLV <- CLV
summary(telco$CLV)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      1145   1297   1300    1296   1300    1300
```

```
# mean CLV by customer category and internet
tab_cust_internet <- aggregate(CLV ~ custcat + internet, data = telco, FUN = mean, na.rm = TRUE)
tab_gender <- aggregate(CLV ~ gender, data = telco, FUN = mean, na.rm = TRUE)

tab_cust_internet
```

```
##      custcat internet      CLV
## 1 Basic service      No 1296.519
## 2   E-service      No 1299.799
## 3 Plus service      No 1299.566
## 4 Total service      No 1299.629
## 5 Basic service     Yes 1273.443
```

```
## 6      E-service      Yes 1297.807
## 7 Plus service      Yes 1297.349
## 8 Total service     Yes 1294.773
```

```
tab_gender
```

```
##  gender      CLV
## 1 Female 1296.548
## 2  Male 1295.950
```

CLV is fairly similar in absolute size across service categories, but there are systematic differences: higher-tier plans and no-internet segments tend to have slightly higher CLV, while Basic + Internet customers have noticeably lower CLV and higher churn.

And as for gender, it does not meaningfully affect customer value either (no discrimination :D)

```
aggregate(CLV ~ gender, data = telco, FUN = mean)
```

```
##  gender      CLV
## 1 Female 1296.548
## 2  Male 1295.950
```

```
ggplot(telco, aes(x = CLV, color = gender, fill = gender)) +
  geom_density(alpha = 0.3, linewidth = 1.2) +
  scale_color_manual(values = wes_palette("FrenchDispatch", 2)) +
  scale_fill_manual(values = wes_palette("FrenchDispatch", 2)) +
  labs(title = "CLV Density by Gender", x = "", y = "") +
  theme_minimal(base_size = 14) +
  theme(legend.title = element_blank())
```

## CLV Density by Gender



Again, this proves that gender has no meaningful impact on the CLV.

```
wes_auto <- function(x, palette = "Darjeeling1") {
  n <- nlevels(x)
  if (n <= length(wesanderson::wes_palettes[[palette]])) {
    wes_palette(palette, n, type = "discrete")
  } else {
    wes_palette(palette, n, type = "continuous")
  }
}
```

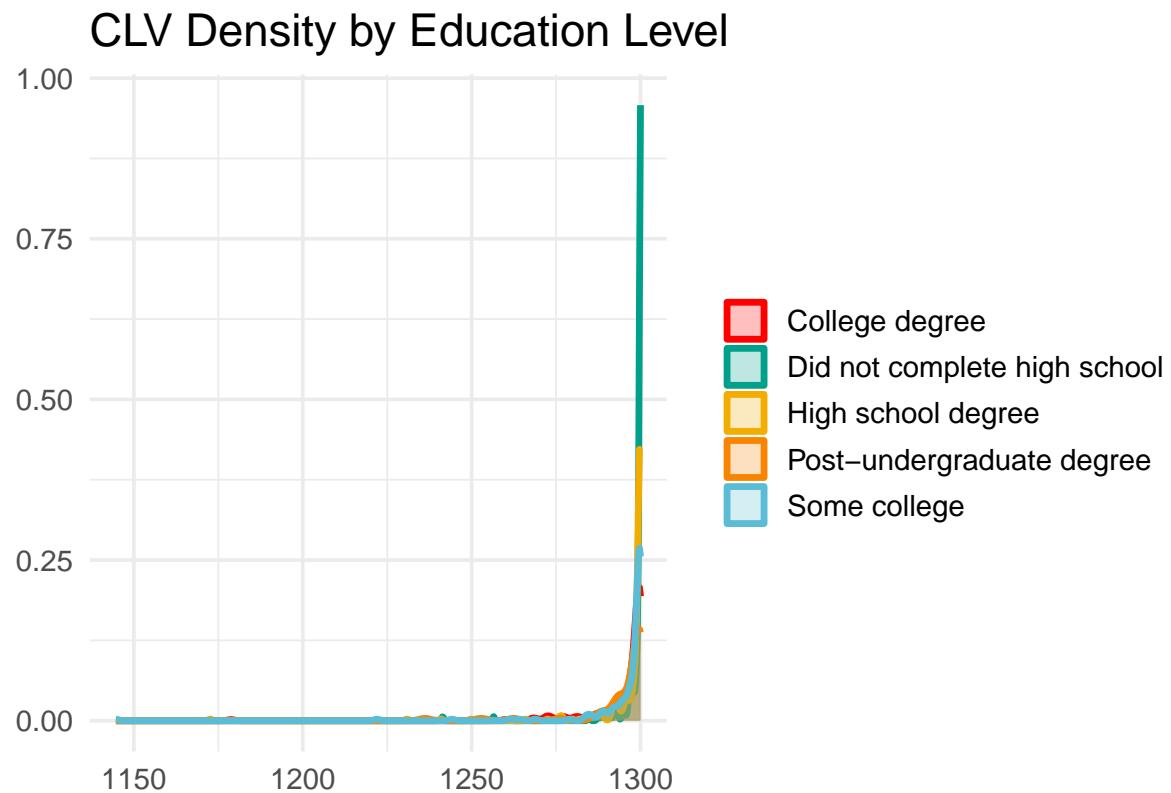
```
aggregate(CLV ~ ed, data = telco, FUN = mean)
```

```
##              ed      CLV
## 1      College degree 1295.199
## 2 Did not complete high school 1298.386
## 3      High school degree 1296.397
## 4 Post-undergraduate degree 1294.860
## 5      Some college 1295.622
```

```
pal_ed <- wes_auto(telco$ed)
```

```
ggplot(telco, aes(x = CLV, color = ed, fill = ed)) +
  geom_density(alpha = 0.25, linewidth = 1.2) +
  scale_color_manual(values = pal_ed) +
  scale_fill_manual(values = pal_ed) +
  labs(title = "CLV Density by Education Level", x = "", y = "") +
```

```
theme_minimal(base_size = 14) +
theme(legend.title = element_blank())
```



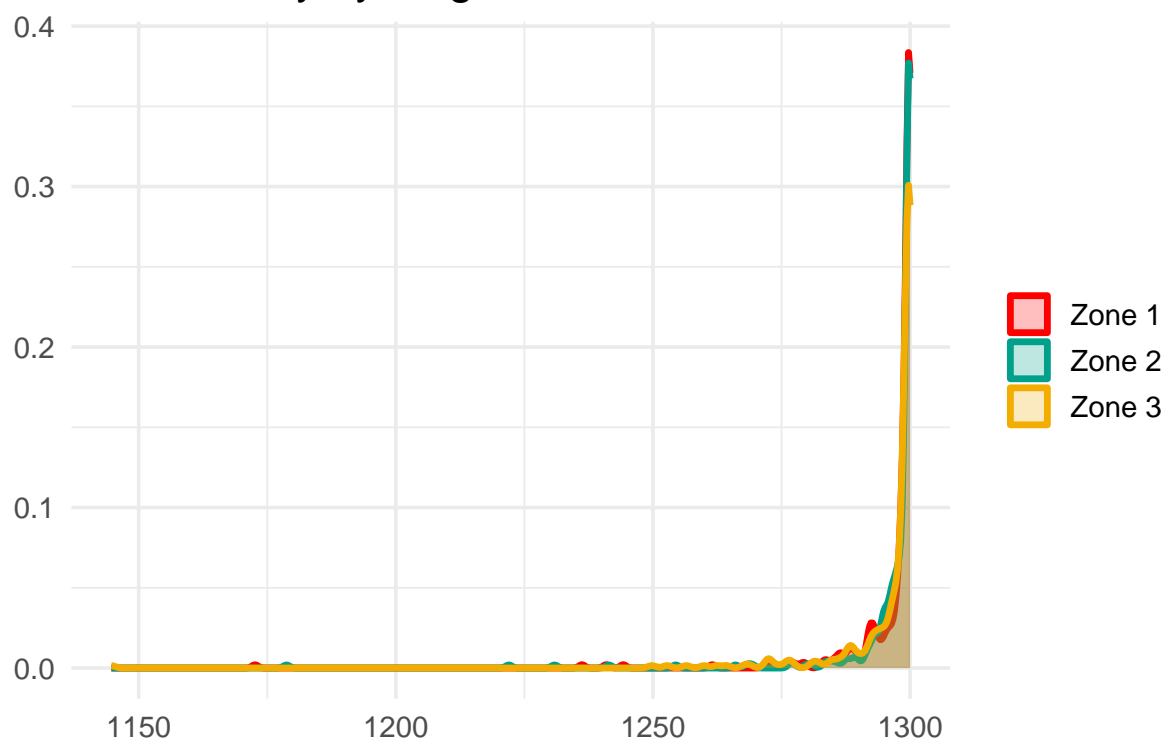
```
aggregate(CLV ~ region, data = telco, FUN = mean)
```

```
##   region      CLV
## 1 Zone 1 1296.381
## 2 Zone 2 1296.609
## 3 Zone 3 1295.805
```

```
pal_region <- wes_auto(telco$region)
```

```
ggplot(telco, aes(x = CLV, color = region, fill = region)) +
  geom_density(alpha = 0.25, linewidth = 1.2) +
  scale_color_manual(values = pal_region) +
  scale_fill_manual(values = pal_region) +
  labs(title = "CLV Density by Region", x = "", y = "") +
  theme_minimal(base_size = 14) +
  theme(legend.title = element_blank())
```

## CLV Density by Region



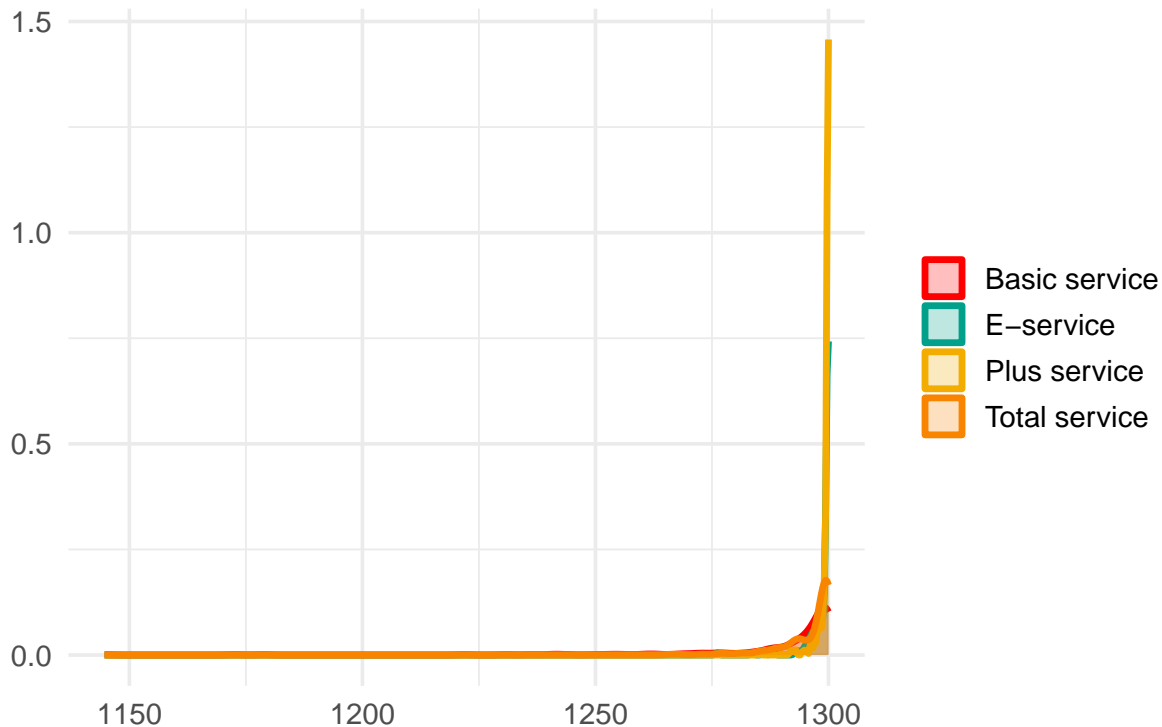
```
aggregate(CLV ~ custcat, data = telco, FUN = mean)
```

```
##      custcat      CLV
## 1 Basic service 1291.054
## 2   E-service 1298.790
## 3 Plus service 1299.392
## 4 Total service 1296.069
```

```
pal_cust <- wes_auto(telco$custcat)
```

```
ggplot(telco, aes(x = CLV, color = custcat, fill = custcat)) +
  geom_density(alpha = 0.25, linewidth = 1.2) +
  scale_color_manual(values = pal_cust) +
  scale_fill_manual(values = pal_cust) +
  labs(title = "CLV Density by Customer Category", x = "", y = "") +
  theme_minimal(base_size = 14) +
  theme(legend.title = element_blank())
```

## CLV Density by Customer Category



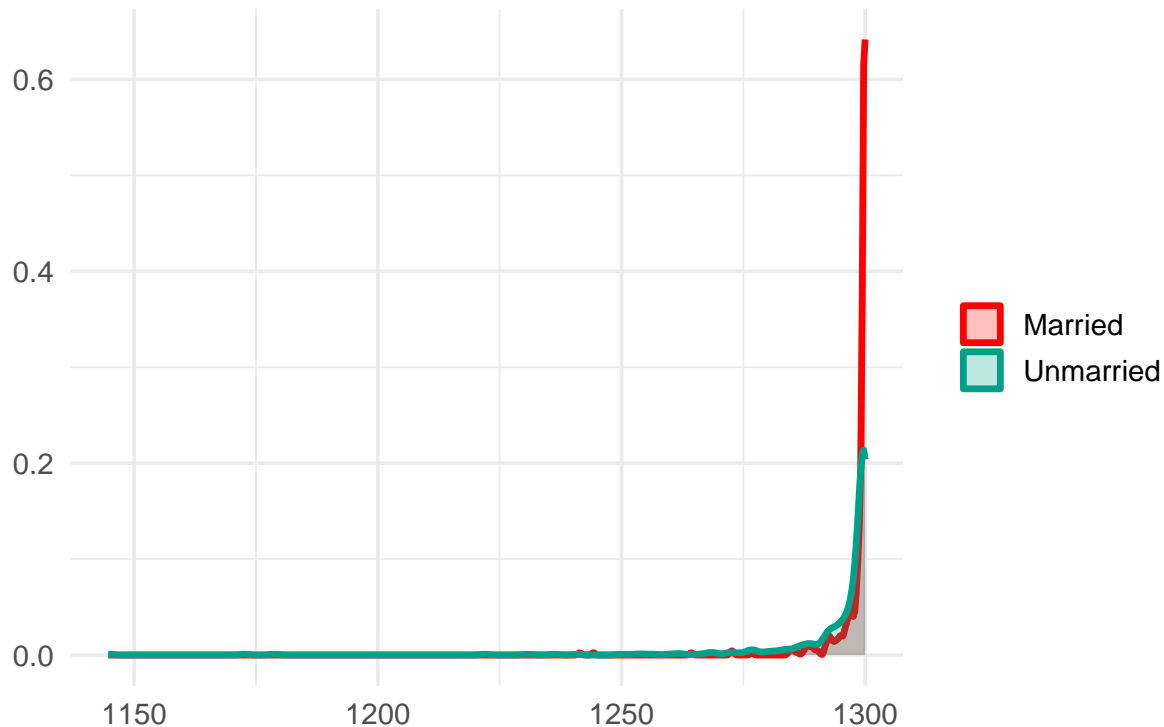
```
aggregate(CLV ~ marital, data = telco, FUN = mean)
```

```
##      marital      CLV
## 1   Married 1298.123
## 2 Unmarried 1294.432
```

```
pal_marital <- wes_auto(telco$marital)
```

```
ggplot(telco, aes(x = CLV, color = marital, fill = marital)) +
  geom_density(alpha = 0.25, linewidth = 1.2) +
  scale_color_manual(values = pal_marital) +
  scale_fill_manual(values = pal_marital) +
  labs(title = "CLV Density by Marital Status", x = "", y = "") +
  theme_minimal(base_size = 14) +
  theme(legend.title = element_blank())
```

## CLV Density by Marital Status



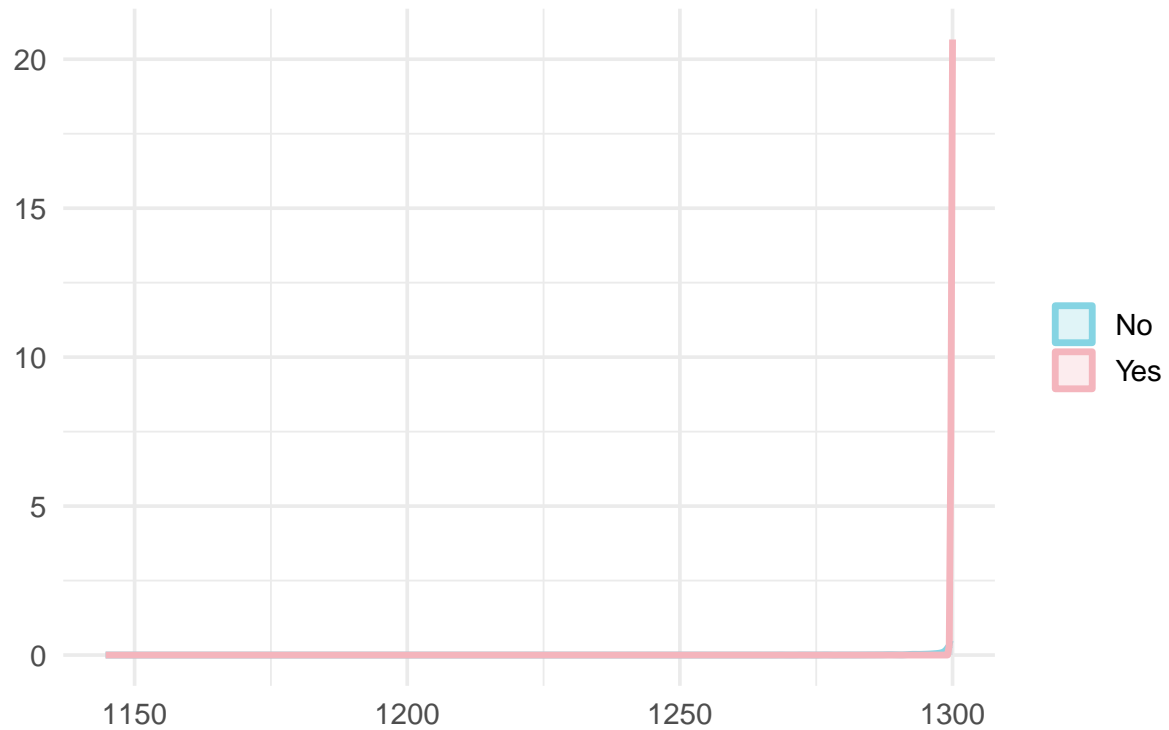
```
aggregate(CLV ~ retire, data = telco, FUN = mean)
```

```
##   retire    CLV
## 1     No 1296.077
## 2     Yes 1299.960
```

```
pal_retire <- wes_auto(telco$retire, palette = "Moonrise3")

ggplot(telco, aes(x = CLV, color = retire, fill = retire)) +
  geom_density(alpha = 0.25, linewidth = 1.2) +
  scale_color_manual(values = pal_retire) +
  scale_fill_manual(values = pal_retire) +
  labs(title = "CLV Density by Retirement Status", x = "", y = "") +
  theme_minimal(base_size = 14) +
  theme(legend.title = element_blank())
```

## CLV Density by Retirement Status

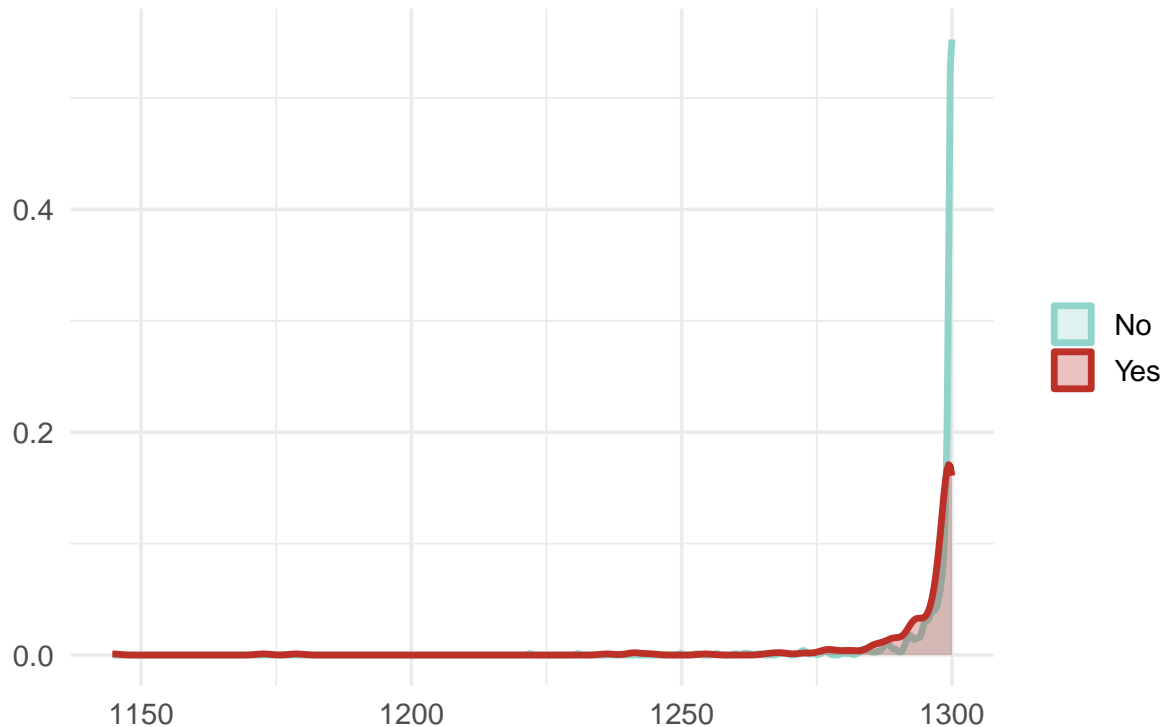


```
aggregate(CLV ~ voice, data = telco, FUN = mean)
```

```
##   voice      CLV
## 1    No 1297.271
## 2    Yes 1293.944
```

```
ggplot(telco, aes(x = CLV, color = voice, fill = voice)) +
  geom_density(alpha = 0.3, linewidth = 1.2) +
  scale_color_manual(values = wes_palette("FrenchDispatch", 2)) +
  scale_fill_manual(values = wes_palette("FrenchDispatch", 2)) +
  labs(title = "CLV Density by Voice Subscription", x = "", y = "") +
  theme_minimal(base_size = 14) +
  theme(legend.title = element_blank())
```

## CLV Density by Voice Subscription

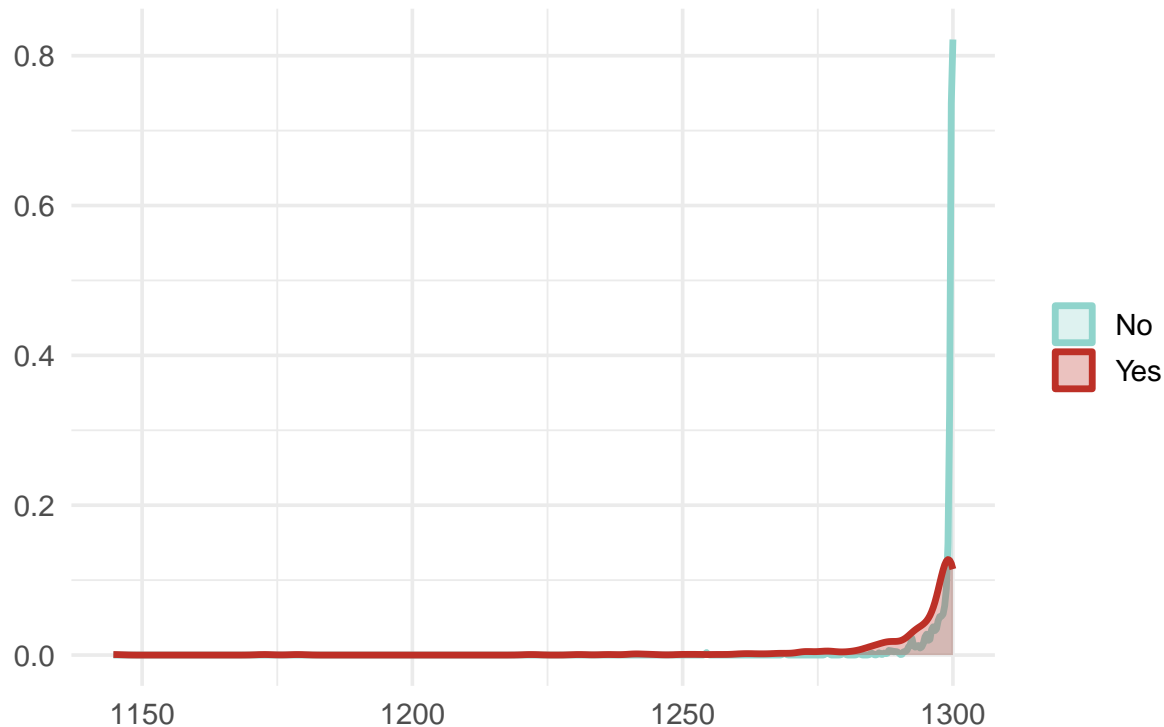


```
aggregate(CLV ~ internet, data = telco, FUN = mean)
```

```
##  internet      CLV
## 1       No 1298.633
## 2       Yes 1292.182
```

```
ggplot(telco, aes(x = CLV, color = internet, fill = internet)) +
  geom_density(alpha = 0.3, linewidth = 1.2) +
  scale_color_manual(values = wes_palette("FrenchDispatch", 2)) +
  scale_fill_manual(values = wes_palette("FrenchDispatch", 2)) +
  labs(title = "CLV Density by Internet Subscription", x = "", y = "") +
  theme_minimal(base_size = 14) +
  theme(legend.title = element_blank())
```

## CLV Density by Internet Subscription

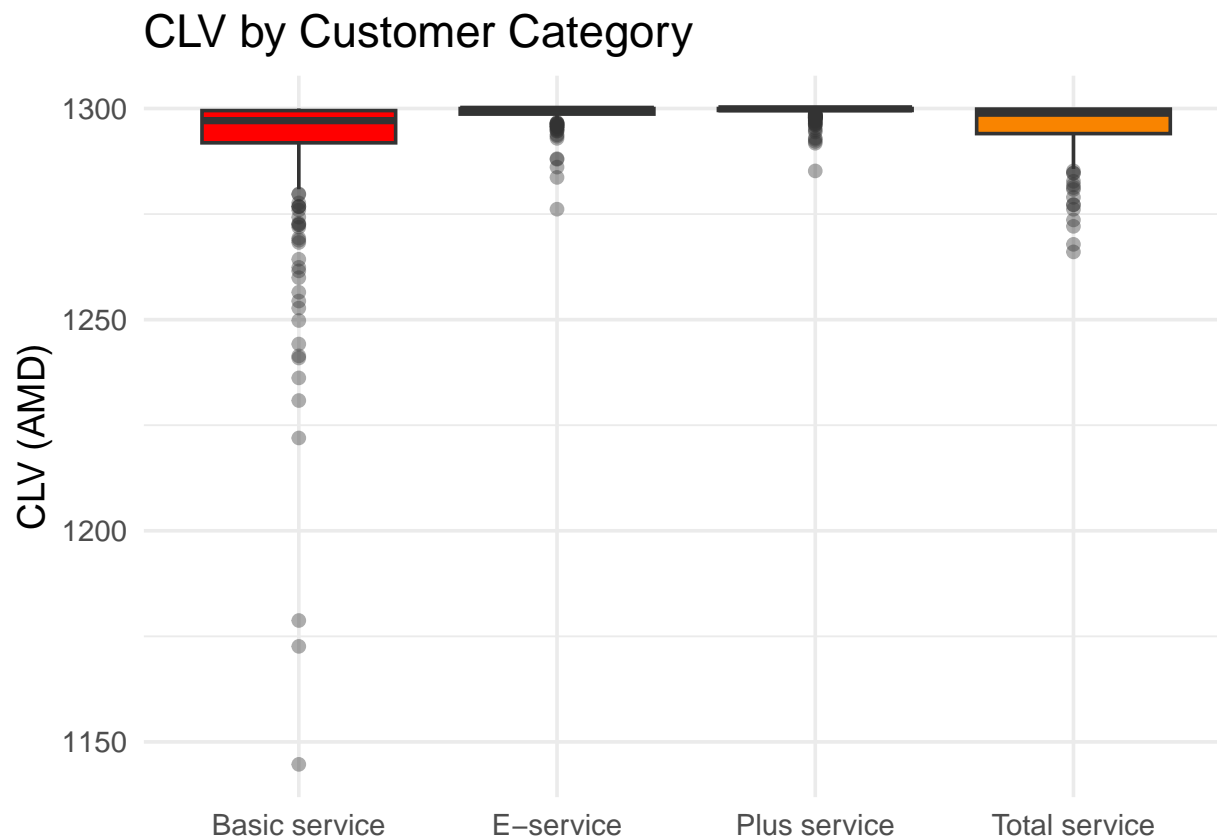


```
telco$custcat <- factor(telco$custcat)
telco$custcat <- droplevels(telco$custcat)

n_levels <- nlevels(telco$custcat)
stopifnot(n_levels > 0)
pal_name <- "Darjeeling1"

if (n_levels <= length(wesanderson::wes_palettes[[pal_name]])) {
  pal_cat <- wes_palette(pal_name, n_levels, type = "discrete")
} else {
  pal_cat <- wes_palette(pal_name, n_levels, type = "continuous")
}

ggplot(telco, aes(x = custcat, y = CLV, fill = custcat)) +
  geom_boxplot(outlier.alpha = 0.4) +
  scale_fill_manual(values = pal_cat) +
  labs(title = "CLV by Customer Category", x = NULL, y = "CLV (AMD)") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "none")
```



Higher service plans like the E-service, Plus, Total service have higher and more stable CLV, suggesting stronger loyalty and lower churn risk. In contrast, Basic service customers show noticeably lower and more variable CLV, indicating higher churn likelihood and inconsistent engagement.

```
n <- nrow(telco)

get_est <- function(out) {
  if (is.list(out) && !is.data.frame(out)) {
    vapply(out, function(x) x$est[1], numeric(1))
  } else if (is.data.frame(out) && "est" %in% names(out)) {
    out$est
  } else numeric(0)
}

times12 <- 1:12
disc12 <- (1 + r/12)^(-(times12 - 1))

S12 <- numeric(n)
CLV12 <- numeric(n)

for (i in seq_len(n)) {
  s12 <- summary(fs_fit, newdata = telco[i, , drop = FALSE],
                type = "survival", t = 12)
  e12 <- get_est(s12)
  S12[i] <- if (length(e12)) pmin(pmax(e12[1], 0), 1) else 0
  s_list <- summary(fs_fit, newdata = telco[i, , drop = FALSE],
                    type = "survival", t = times12)
```

```

Si <- get_est(s_list)

if (length(Si) < length(times12)) {
  pad <- if (length(Si)) rep(tail(Si, 1), length(times12) - length(Si)) else rep(0, length(times12))
  Si <- c(Si, pad)
} else if (length(Si) > length(times12)) {
  Si <- Si[seq_along(times12)]
}

Si <- pmin(pmax(Si, 0), 1)
CLV12[i] <- MM * sum(Si * disc12)
}

telco$S12 <- S12
telco$PrChurn_12m <- 1 - S12
telco$at_risk_12m <- telco$S12 <= 0.5
telco$CLV12 <- CLV12

expected_loss <- sum(telco$PrChurn_12m * telco$CLV12, na.rm = TRUE)
budget_15pct <- 0.15 * expected_loss
budget_20pct <- 0.20 * expected_loss
format(round(budget_15pct, 0), big.mark = ",")

## [1] "244,781"

format(round(budget_20pct, 0), big.mark = ",")

## [1] "326,374"

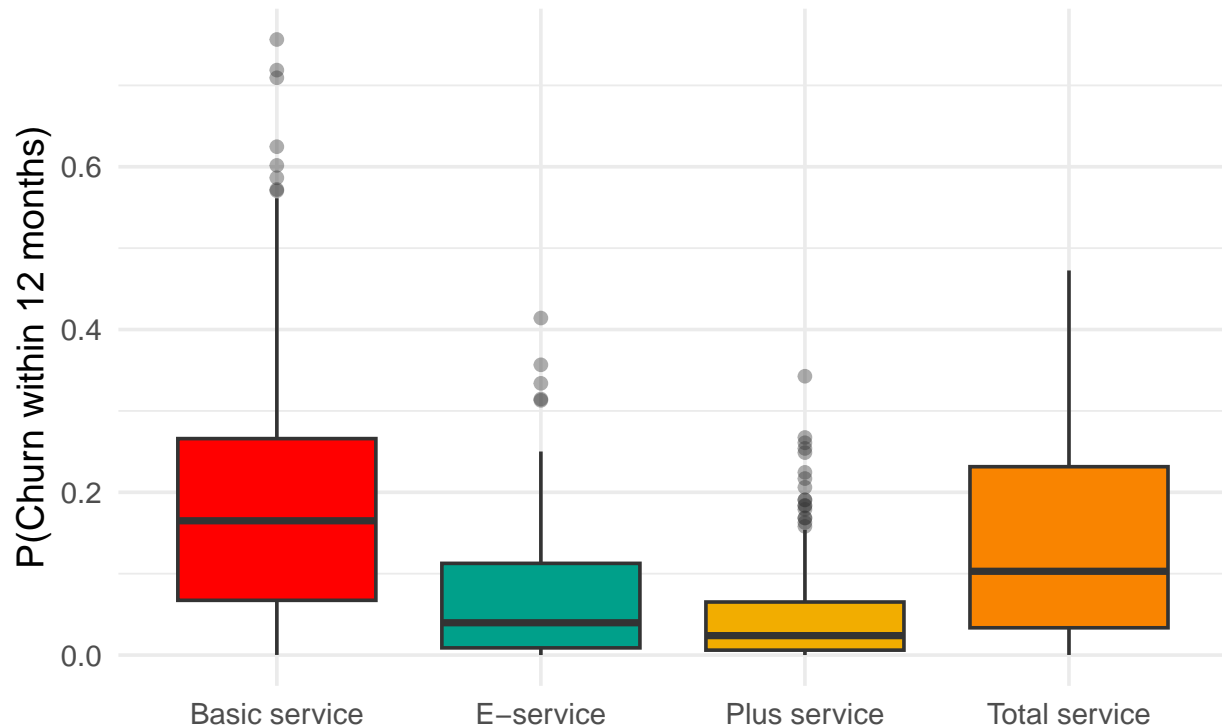
at_risk <- telco$PrChurn_12m >= 0.5
expected_loss_risk <- sum(telco$CLV12[at_risk], na.rm = TRUE)
budget_risk_15pct <- 0.15 * expected_loss_risk
format(round(budget_risk_15pct, 0), big.mark = ",")

## [1] "29,565"

ggplot(telco, aes(x = custcat, y = PrChurn_12m, fill = custcat)) +
  geom_boxplot(outlier.alpha = 0.4) +
  scale_fill_manual(values = pal_cat) +
  labs(title = "12-Month Churn Probability by Service Tier",
       x = "", y = "P(Churn within 12 months)") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "none")

```

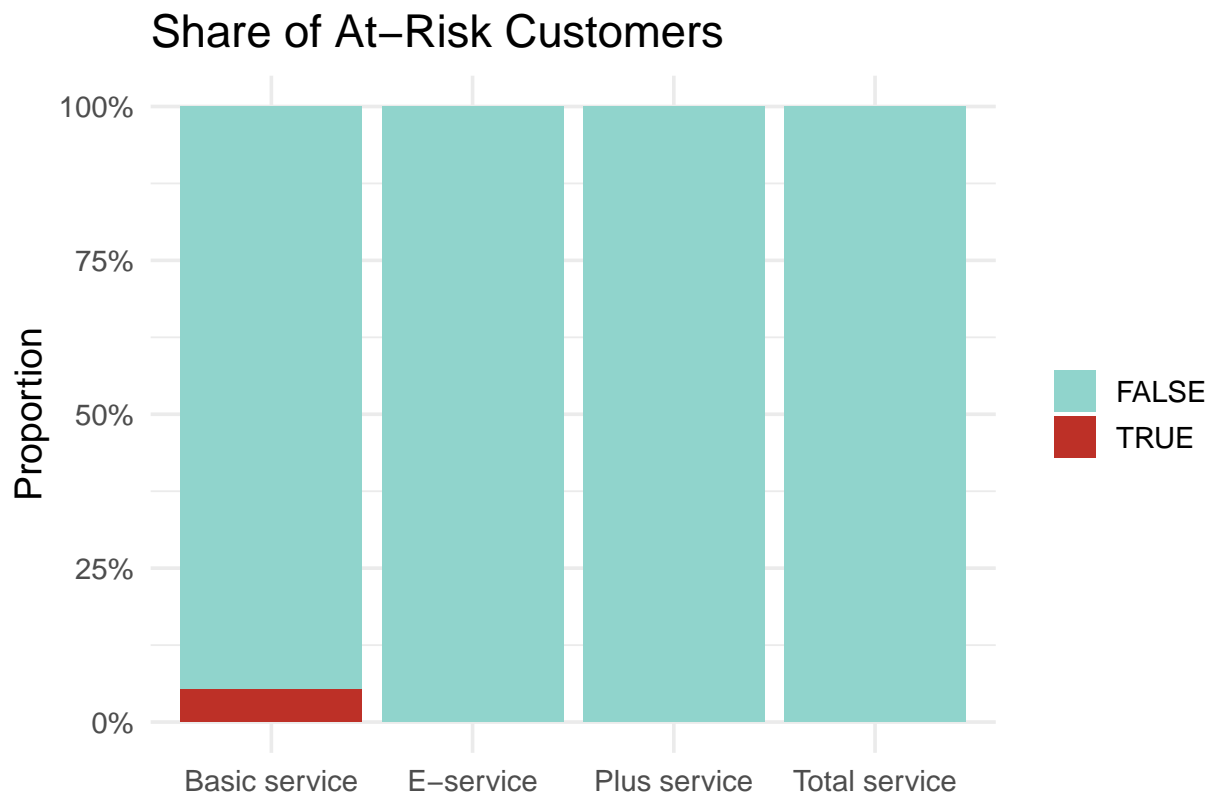
## 12-Month Churn Probability by Service Tier



The risk of churn varies substantially by service tier. Basic service customers have the highest predicted churn probabilities (as expected based on the past analysis), with a median risk around 15–25% and some individuals exceeding 60% within a year. This indicates that customers with minimal service bundles are more price-sensitive and more likely to switch. In contrast, Plus and E-service customers show much lower churn risk, typically below 10%, reflecting stronger loyalty and perceived value in enhanced service features. Total service customers show a wider distribution, suggesting that while many of them are loyal, a proper count of them demonstrates churn risk—possibly due to higher expectations or pricing sensitivity.

```
risk_summary <- telco %>%
  group_by(custcat, at_risk_12m) %>%
  summarise(n = n(), .groups = "drop") %>%
  group_by(custcat) %>%
  mutate(prop = n / sum(n))

ggplot(risk_summary, aes(x = custcat, y = prop, fill = at_risk_12m)) +
  geom_col(position = "fill") +
  scale_fill_manual(values = wes_palette("FrenchDispatch", 2)) +
  labs(title = "Share of At-Risk Customers",
       x = "", y = "Proportion") +
  theme_minimal(base_size = 14) +
  scale_y_continuous(labels = scales::percent) +
  theme(legend.title = element_blank())
```



The customers most likely to churn within the next 12 months are the Basic service.

```
seg <- telco |>
  dplyr::group_by(custcat, internet) |>
  dplyr::summarise(
    mean_CLV = mean(CLV, na.rm = TRUE),
    mean_CLV12= mean(CLV12, na.rm = TRUE),
    med_pchurn= median(PrChurn_12m, na.rm = TRUE),
    n = dplyr::n(),
    .groups = "drop"
  ) |>
  dplyr::arrange(dplyr::desc(mean_CLV))
seg
```

```
## # A tibble: 8 x 6
##   custcat      internet mean_CLV mean_CLV12 med_pchurn      n
##   <fct>        <fct>      <dbl>      <dbl>    <dbl> <int>
## 1 E-service    No          1300.      14908.    0.0131   107
## 2 Total service No          1300.      14906.    0.0225    63
## 3 Plus service No          1300.      14905.    0.0204   259
## 4 E-service    Yes          1298.      14885.    0.0932   110
## 5 Plus service Yes          1297.      14880.    0.0870    22
## 6 Basic service No          1297.      14870.    0.121    203
## 7 Total service Yes          1295.      14850.    0.143   173
## 8 Basic service Yes          1273.      14605.    0.350    63
```

Up to this point, we studied the model - LogNormal as it had the lowest AIC, now let's consider all the

other models available.

```
base_formula <- Surv(tenure, churn == 1) ~ age + marital + address + voice + internet + custcat
dists <- c("exp", "weibull", "weibullph", "gompertz", "lognormal", "llogis", "gengamma", "genf")

fits <- setNames(vector("list", length(dists)), dists)
for (i in seq_along(dists)) {
  fits[[i]] <- try(flexsurv::flexsurvreg(base_formula, data = telco, dist = dists[i]), silent = TRUE)}
fits <- fits[!vapply(fits, function(x) inherits(x, "try-error"), logical(1))]

cmp <- dplyr::bind_rows(lapply(seq_along(fits), function(i) {
  f <- fits[[i]]
  k <- nrow(f$res)                                # number of parameters
  tibble::tibble(
    dist = dists[i],
    logLik = as.numeric(stats::logLik(f)),
    k = k,
    AIC = f$AIC,
    BIC = f$AIC + log(nrow(telco)) * k
  )
})) %>% dplyr::arrange(AIC)
cmp
```

```
## # A tibble: 8 x 5
##   dist      logLik      k   AIC   BIC
##   <chr>      <dbl> <int> <dbl> <dbl>
## 1 lognormal -1462.    10 2944. 3013.
## 2 gengamma  -1462.    11 2945. 3021.
## 3 genf      -1462.    12 2947. 3030.
## 4 llogis    -1464.    10 2947. 3016.
## 5 gompertz  -1467.    10 2954. 3023.
## 6 weibull   -1468.    10 2956. 3025.
## 7 weibullph -1468.    10 2956. 3025.
## 8 exp       -1473.     9 2963. 3026.
```

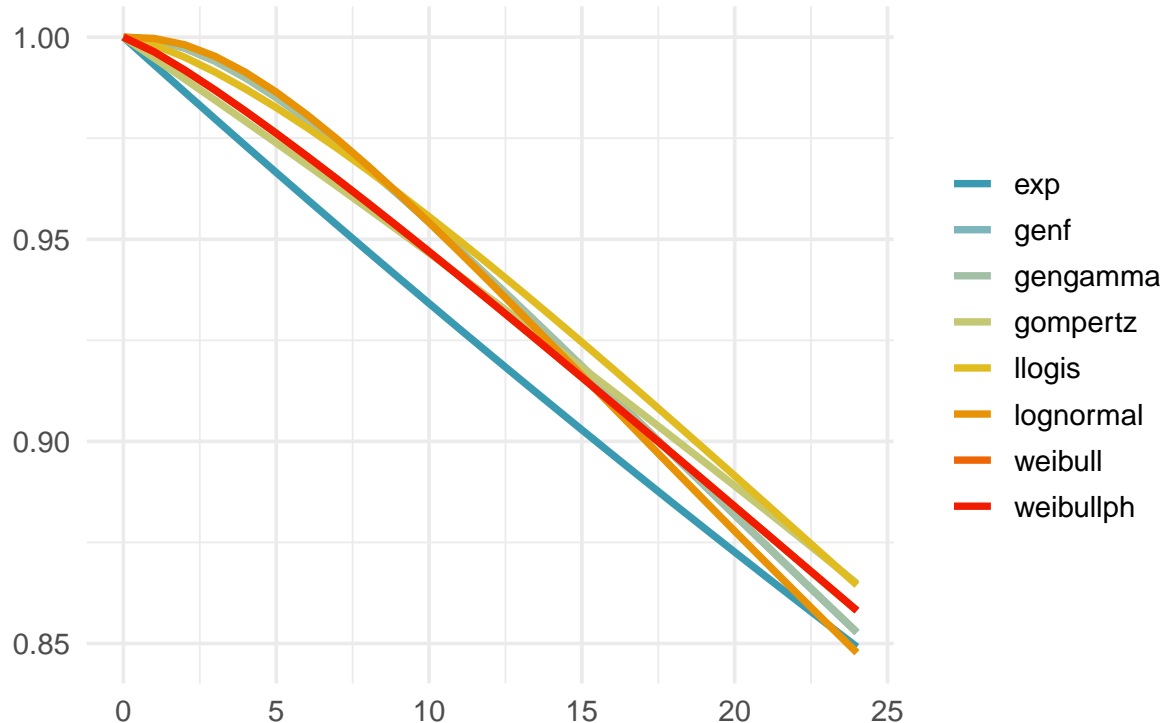
```
times <- 0:24
newref <- telco[1, , drop = FALSE]

curves <- dplyr::bind_rows(lapply(dists, function(d) {
  s <- summary(fits[[d]], newdata = newref, t = times, type = "survival")
  data.frame(dist = d, t = times, S = s[[1]]$est)
}))

ggplot(curves, aes(t, S, color = dist)) +
  geom_line(linewidth = 1.3) +
  scale_color_manual(values = wes_palette("Zissou1", n_distinct(curves$dist),
                                          type = "continuous")) +
  labs(title = "Parametric Survival Model Comparison",
       x = "", y = "",
       color = "Distribution") +
```

```
theme_minimal(base_size = 14) +
theme(legend.position = "right", legend.title = element_blank())
```

## Parametric Survival Model Comparison



```
coef_tab <- broom::tidy(final_model) |>
  dplyr::mutate(time_ratio = exp(estimate)) |>
  dplyr::select(term, estimate, std.error, statistic, p.value, time_ratio)
coef_tab
```

```
## # A tibble: 10 x 6
##   term                estimate std.error statistic  p.value time_ratio
##   <chr>                <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)          2.53      0.243     10.4 1.49e-25    12.6
## 2 age                  0.0368    0.00640     5.75 8.69e- 9     1.04
## 3 maritalUnmarried    -0.447    0.114     -3.91 9.32e- 5     0.639
## 4 address              0.0428    0.00885     4.84 1.30e- 6     1.04
## 5 voiceYes            -0.463    0.167     -2.78 5.45e- 3     0.629
## 6 internetYes         -0.841    0.138     -6.08 1.21e- 9     0.431
## 7 custcatE-service     1.03     0.169     6.07 1.29e- 9     2.79
## 8 custcatPlus service  0.823    0.169     4.85 1.21e- 6     2.28
## 9 custcatTotal service 1.01     0.210     4.83 1.33e- 6     2.75
## 10 Log(scale)          0.283    0.0460     6.15 7.74e-10     1.33
```

Insights. To model churn behaviour, a series of Accelerated Failure Time (AFT) models were fitted using both survreg and flexsurvreg. The comparison across distributions showed that the LogNormal model achieved

the lowest AIC among the parametric forms tested that yielded the values of AIC being approximately 2944, and BIC – 3013. The LogNormal specification was therefore selected as the final model. The final model identified age, marital status, years at current address, voice subscription, internet subscription, and customer category as statistically significant predictors (p-value being less than 0.05) of time-to-churn. Interpreting coefficients via time ratios ( $\exp(\beta)$ ), subscriptions to voice and internet services are associated with shorter expected lifetimes (time ratios  $< 1$ ), indicating faster churn hazard and lower retention stability than for otherwise similar customers without these services. Basic service customers exhibit lower time ratios, implying faster churn, while higher-end users like Plus, E-service, and Total customers have longer survival times. The positive association between address duration and retention suggests that settled customers are more loyal, while certain marital categories reflect more stable usage patterns consistent with likely shared household plans. Customer Lifetime Value (CLV) was estimated from predicted survival curves using monthly discounting. Mean 12-month CLV ranged from roughly 14,605 AMD to 14,908 AMD across service tiers. While CLV was nearly identical across gender and education groups, it varied meaningfully across customer categories and internet segments. For example, E-service (No internet) customers exhibited the highest mean CLV12 of about 14,908 AMD and very low median churn probability about 1.3%, while Basic + Internet customers had the lowest CLV12 about 14,605 AMD and the highest median churn probability around 35%. This demonstrates that service tier and internet subscription structure, rather than demographic traits, are the primary drivers of a customer’s value. Using the model-derived churn probabilities, customers with  $S(12) \leq 0.5$  were labeled as being at-risk. These were concentrated overwhelmingly in the Basic category which was expected. The expected monetary loss from churn across the full base was estimated at around 244,781–326,374 AMD, depending on assumed intervention intensity, while targeting only at-risk subscribers reduces the required annual retention budget to approximately 29,565 AMD at a 15% incentive level.

To conclude, the most valuable customers are the high-tier subscribers, especially E-service and Plus customers without internet, who exhibit the highest CLV and the lowest churn risk. The most effective retention focus is on Basic service customers with high churn risk, particularly those with internet (and possibly voice) services, who combine relatively low CLV with very high 12-month churn probabilities.

P.S. I apologise for the long report.