bimbo_explore.R

gquai

2019-12-29

```
# DATA SCIENCE ACADEMY
# Big Data Analytics com R e Microsoft Azure Machine Learning
#
# Model to accurately predict inventory demand based on data sales histories
#
# Gabriel Quaiotti
# Nov 2019
#
# In this competition, you will forecast the demand of a product for a given week, at a
# particular store.
# The dataset you are given consists of 9 weeks of sales transactions in Mexico. Every week,
# there are delivery
# trucks that deliver products to the vendors. Each transaction consists of sales and returns.
# Returns are the products that are unsold and expired. The demand for a product in a certain
# week is defined as the sales this week subtracted by the return next week.
#
#
# The train and test dataset are split based on time, as well as the public and private
# leaderboard dataset split.
#
#
# Things to note:
#
# There may be products in the test set that don't exist in the train set. This is the expected
# behavior of inventory data,
# since there are new products being sold all the time. Your model should be able to accommodate
# this.
#
# The adjusted demand (Demanda_uni_equil) is always >= 0 since demand should be either 0 or a
# positive value. The reason that Venta_uni_hoy - Dev_uni_proxima
# sometimes has negative values is that the returns records sometimes carry over a few weeks.

# File descriptions
# train.csv - the training set
# test.csv - the test set
# sample_submission.csv - a sample submission file in the correct format
# cliente_tabla.csv - client names (can be joined with train/test on Cliente_ID)
# producto_tabla.csv - product names (can be joined with train/test on Producto_ID)
# town_state.csv - town and state (can be joined with train/test on Agencia_ID)

# Data fields
# Semana - Week number (From Thursday to Wednesday)
# Agencia_ID - Sales Depot ID
# Canal_ID - Sales Channel ID
# Ruta_SAK - Route ID (Several routes = Sales Depot)
# Cliente_ID - Client ID
# NombreCliente - Client name
```

```
# Producto_ID - Product ID
# NombreProducto - Product Name
# Venta_uni_hoy - Sales unit this week (integer)
# Venta_hoy - Sales this week (unit: pesos)
# Dev_uni_proxima - Returns unit next week (integer)
# Dev_proxima - Returns next week (unit: pesos)
# Demanda_uni_equil - Adjusted Demand (integer) (This is the target you will predict)

# EXPLORE

setwd('D:/Github/DSA_BIMBO_INVENTORY')

library(data.table)
library(ggplot2)
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
library(scales)

v_c_output_warehouse_sum <- "dataset/warehouse_sum.csv"
v_c_output_warehouse_median <- "dataset/warehouse_median.csv"

v_c_output_route_sum <- "dataset/route_sum.csv"
v_c_output_route_median <- "dataset/route_median.csv"

v_c_output_customer_sum <- "dataset/customer_sum.csv"
v_c_output_customer_median <- "dataset/customer_median.csv"

v_c_output_product_sum <- "dataset/product_sum.csv"
v_c_output_product_median <- "dataset/product_median.csv"


v_c_file_train <- "dataset/train.csv"

# v_c_file_item <- "dataset/producto_tabla.csv"
# v_c_file_location <- "dataset/town_state.csv"

# v_c_file_customer <- "dataset/customer.csv"

###################################
# ITEMS
###################################
# item <- data.table::fread(file = v_c_file_item)

# str(item)

# Classes ‘data.table’ and 'data.frame':    2592 obs. of  2 variables:
# $ Producto_ID   : int  0 9 41 53 72 73 98 99 100 106 ...
# $ NombreProducto: chr  "NO IDENTIFICADO 0" "Capuccino Moka 750g NES 9" "Bimbollos Ext sAjonjoli 6p 48

###################################
# LOCATION
```

```
###################################

# location <- data.table::fread(file = v_c_file_location)

# str(location)

# Classes ‘data.table’ and 'data.frame':    790 obs. of  3 variables:
# $ Agencia_ID: int  1110 1111 1112 1113 1114 1116 1117 1118 1119 1120 ...
# $ Town      : chr  "2008 AG. LAGO FILT" "2002 AG. AZCAPOTZALCO" "2004 AG. CUAUTITLAN" "2008 AG. LAGO
# $ State     : chr  "MéXICO, D.F." "MéXICO, D.F." "ESTADO DE MéXICO" "MéXICO, D.F." ...

###################################
# TRAIN DATASET
###################################

train <- data.table::fread(file = v_c_file_train)

str(train)
```

```
## Classes 'data.table' and 'data.frame':   74180464 obs. of  11 variables:
##  $ Semana          : int  3 3 3 3 3 3 3 3 3 3 ...
##  $ Agencia_ID      : int  1110 1110 1110 1110 1110 1110 1110 1110 1110 1110 ...
##  $ Canal_ID        : int  7 7 7 7 7 7 7 7 7 7 ...
##  $ Ruta_SAK        : int  3301 3301 3301 3301 3301 3301 3301 3301 3301 3301 ...
##  $ Cliente_ID      : int  15766 15766 15766 15766 15766 15766 15766 15766 15766 15766 ...
##  $ Producto_ID     : int  1212 1216 1238 1240 1242 1250 1309 3894 4085 5310 ...
##  $ Venta_uni_hoy   : int  3 4 4 4 3 5 3 6 4 6 ...
##  $ Venta_hoy       : num  25.1 33.5 39.3 33.5 22.9 ...
##  $ Dev_uni_proxima : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Dev_proxima     : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Demanda_uni_equil: int  3 4 4 4 3 5 3 6 4 6 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

```
# $ Semana          : int  3 3 3 3 3 3 3 3 3 3 ...
# $ Agencia_ID      : int  1110 1110 1110 1110 1110 1110 1110 1110 1110 1110 ...
# $ Canal_ID        : int  7 7 7 7 7 7 7 7 7 7 ...
# $ Ruta_SAK        : int  3301 3301 3301 3301 3301 3301 3301 3301 3301 3301 ...
# $ Cliente_ID      : int  15766 15766 15766 15766 15766 15766 15766 15766 15766 15766 ...
# $ Producto_ID     : int  1212 1216 1238 1240 1242 1250 1309 3894 4085 5310 ...
# $ Venta_uni_hoy   : int  3 4 4 4 3 5 3 6 4 6 ...
# $ Venta_hoy       : num  25.1 33.5 39.3 33.5 22.9 ...
# $ Dev_uni_proxima : int  0 0 0 0 0 0 0 0 0 0 ...
# $ Dev_proxima     : num  0 0 0 0 0 0 0 0 0 0 ...
# $ Demanda_uni_equil: int  3 4 4 4 3 5 3 6 4 6 ...

# There are no missing values
# any(is.na(train))

# Get random lines to explore
# train_sample <- train[sample(nrow(train), 500000), ]

paste("The training dataset contains", length(unique(train[,Semana])), "weeks")
```

```
## [1] "The training dataset contains 7 weeks"
```

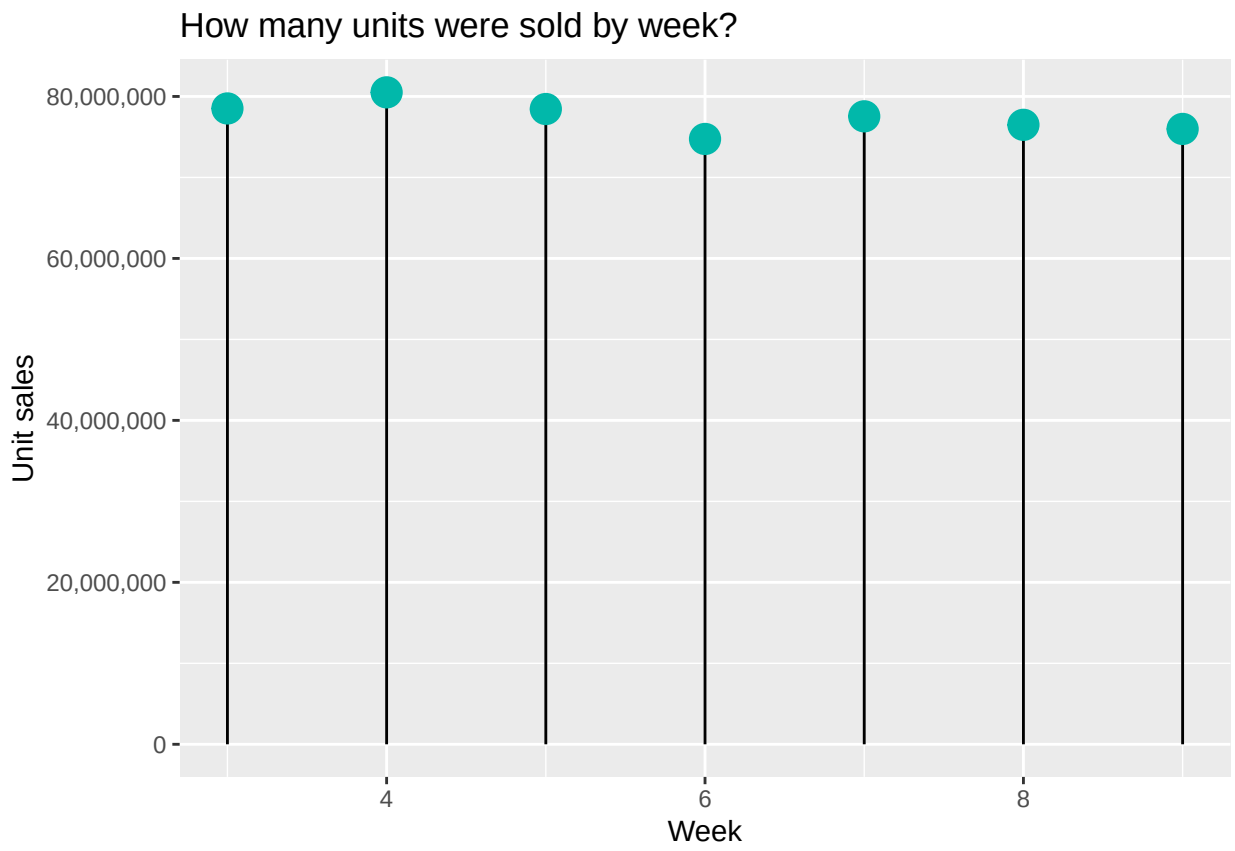
```
unique(train[,Semana])
```

```
## [1] 3 4 5 6 7 8 9
```

```
# Add unit price column
# train[Venta_uni_hoy > 0, unit_price := Venta_hoy / Venta_uni_hoy]

# How many units are sold per week?
df <- train[, sum(Venta_uni_hoy), by=Semana]

summary(df$V1)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 74753832 76244764 77548038 77467331 78485174 80509571
```

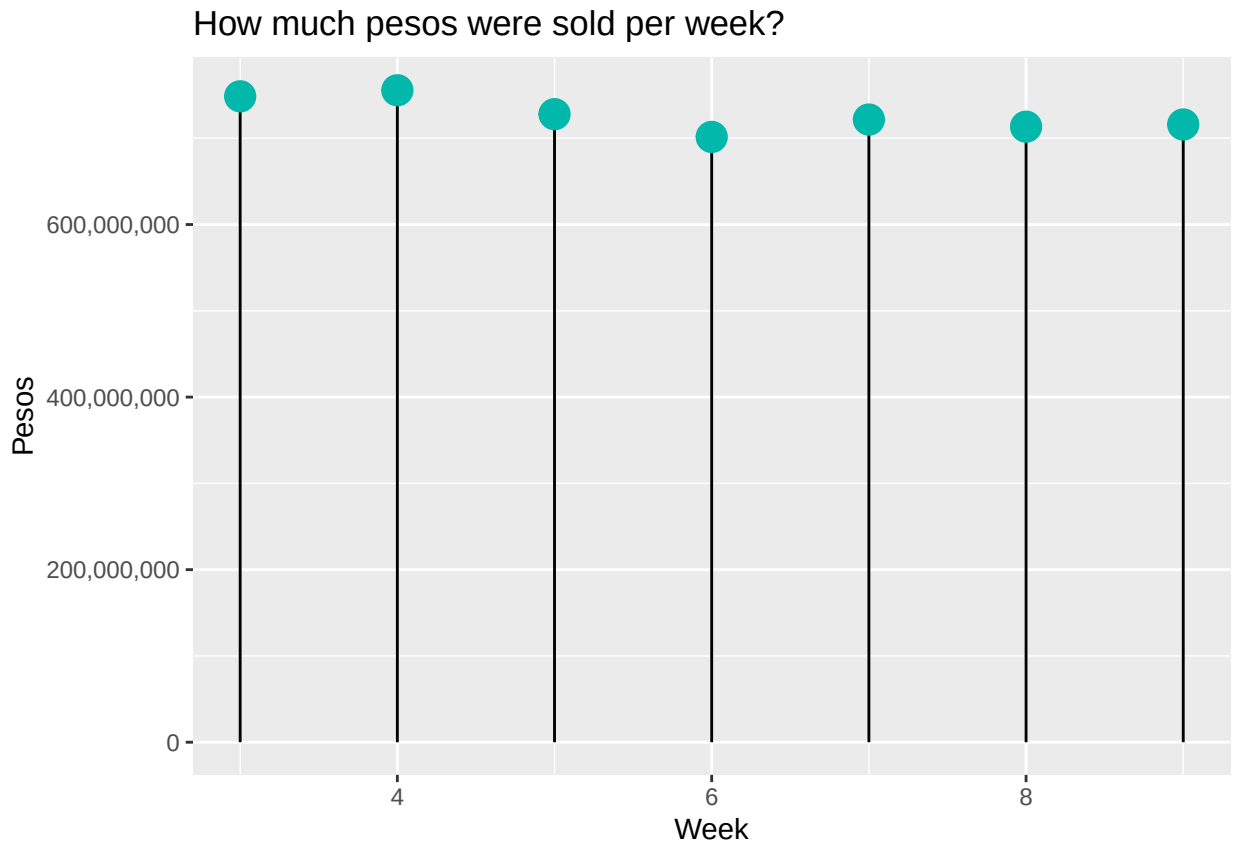
```
ggplot(df, aes(x = Semana, y = V1)) +
  geom_segment(aes(xend = Semana, yend = 0)) +
  geom_point(color = "#01b8aa", size = 5) +
  ggtitle("How many units were sold by week?") +
  ylab("Unit sales") +
  xlab("Week") +
  scale_y_continuous(labels = comma)
```



```
# How much pesos are sold per week?
df <- train[, sum(Venta_hoy), by=Semana]

summary(df$V1)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 701524062 714683710 721662485 726380641 738251590 755607344
```

```
ggplot(df, aes(x = Semana, y = V1)) +
  geom_segment(aes(xend = Semana, yend = 0)) +
  geom_point(color = "#01b8aa", size = 5) +
  ggtitle("How much pesos were sold per week?") +
  ylab("Pesos") +
  xlab("Week") +
  scale_y_continuous(labels = comma)
```




```
remove(df)
gc()
```

```
##             used   (Mb) gc trigger   (Mb)  max used   (Mb)
## Ncells    831001   44.4    1608248   85.9   1221237   65.3
## Vcells 532325111 4061.4  846166684 6455.8 825854908 6300.8
```

```r
# It appears that the week will not be an important feature when training the model
# If there were a complete year, I could look at seasonality

#
# WAREHOUSE ANALISYS
#
# How many sales depot BIMBO has?
paste("The BIMBO group has", length(unique(train[,Agencia_ID])), "sales depots")
```

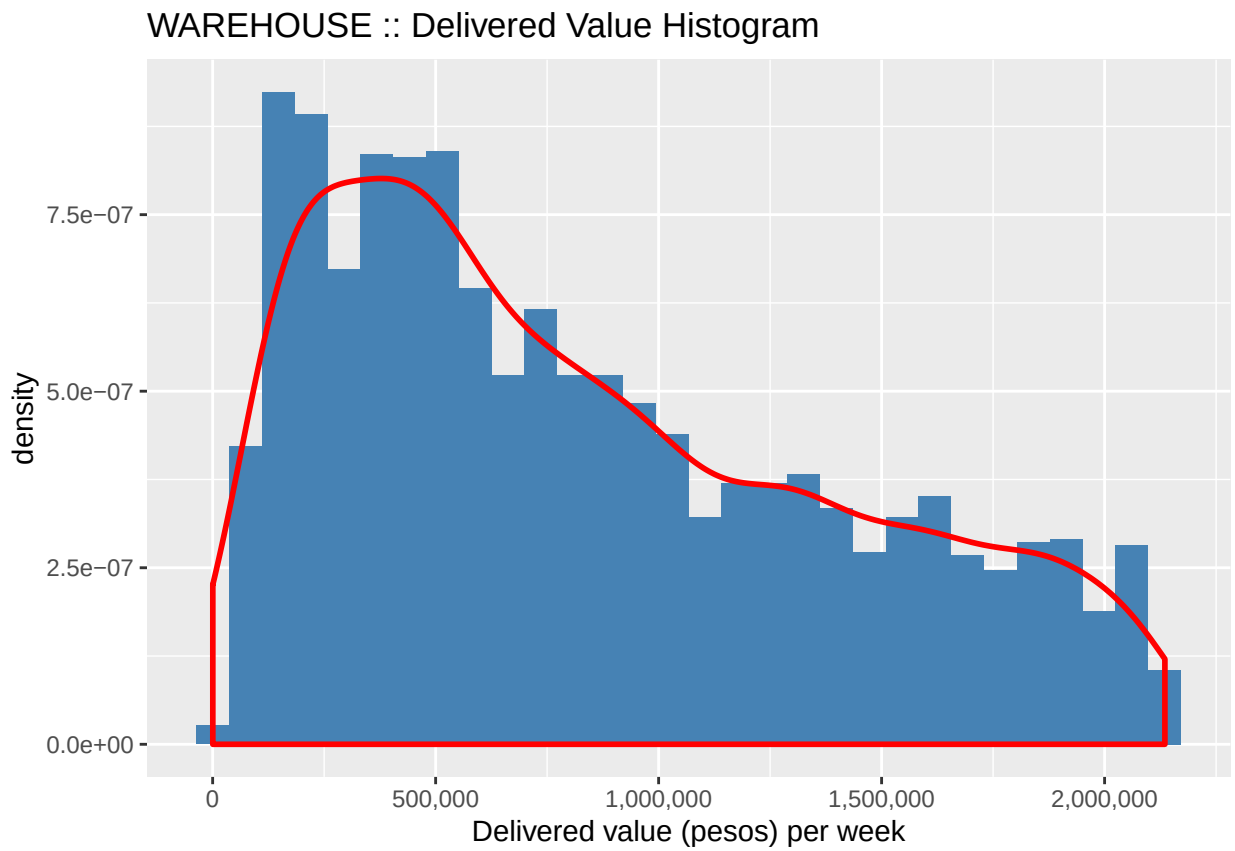
```
## [1] "The BIMBO group has 552 sales depots"
```

```r
# How many items did each sales depot deliver per week?
df <- train[, list(item_sum = sum(Venta_uni_hoy),
                   value_sum = sum(Venta_hoy)),
            by = list(Agencia_ID, Semana)]

# How much PESOS did each warehouse deliver?
summary(df[,value_sum])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     353  442733  954203 1316248 1880732 9766771
```

```r
ggplot(df[value_sum > 0 & value_sum <= quantile(x = df[,value_sum], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) + #, breaks = seq(0, max(df[,value_sum]), 5000000)) +
  geom_histogram(aes(x = value_sum, y = ..density..), fill = "steelblue") +
  geom_density(aes(x = value_sum), color = "red", size = 1) +
  ggtitle("WAREHOUSE :: Delivered Value Histogram") +
  xlab("Delivered value (pesos) per week")
```

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

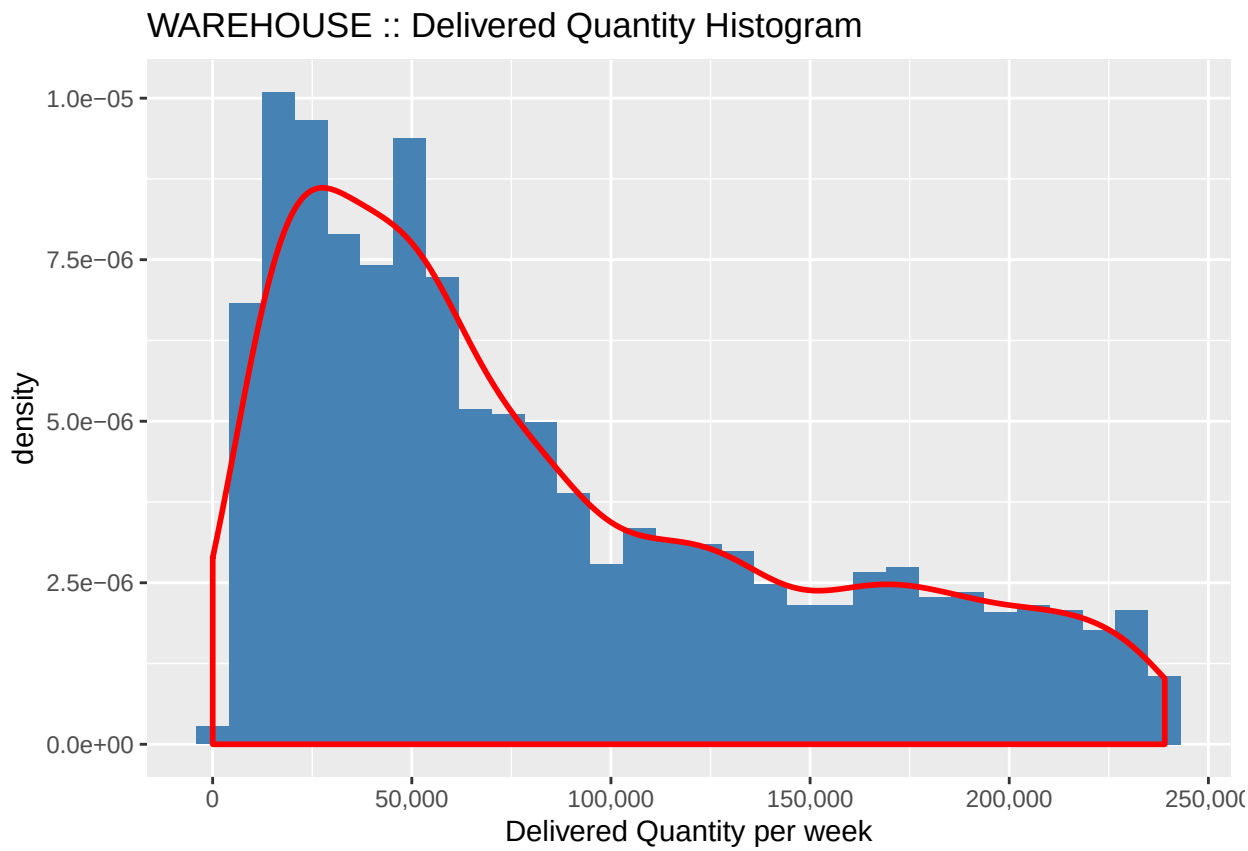



```
# How many ITEMS did each warehouse deliver?
summary(df[,item_sum])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      18   40531   90021  140376  208410  832945
```

```
ggplot(df[item_sum > 0 & item_sum <= quantile(x = df[,item_sum], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = item_sum, y = ..density..), fill = "steelblue") +
  geom_density(aes(x = item_sum), color = "red", size = 1) +
  ggtitle("WAREHOUSE :: Delivered Quantity Histogram") +
    xlab("Delivered Quantity per week")
```

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

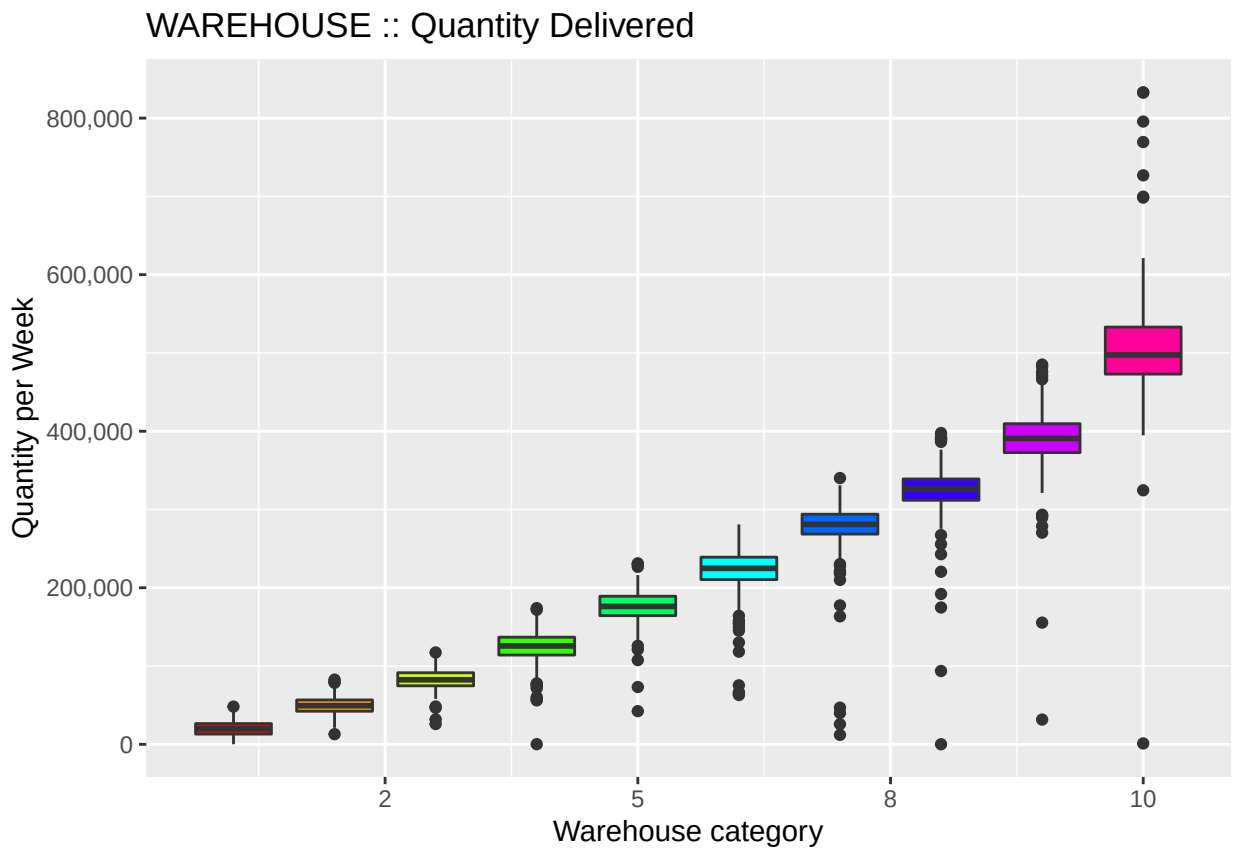



```
ggplot(df) +
  geom_point(aes(x = item_sum, y = value_sum), color = "steelblue") +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("WAREHOUSE :: Deliveries") +
  ylab("Value per Week") +
  xlab("Quantity per Week")
```

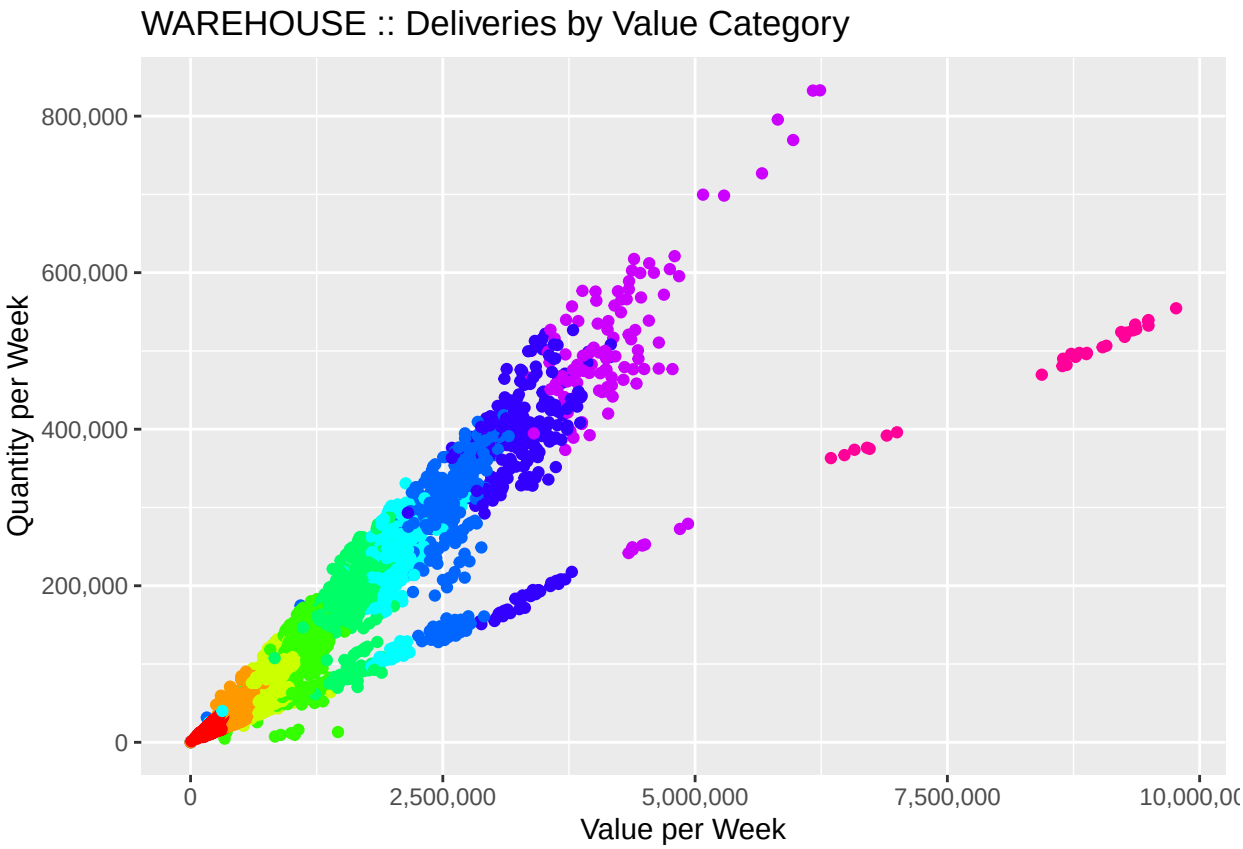

```
# Warehouse category
df1 <- df[, list(item_median = median(item_sum),
                 value_median = median(value_sum)),
          by = Agencia_ID]

cluster <- kmeans(df1[,item_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df1[,item_median], centers)
df1$warehouse_item_category <- cluster$cluster

cluster <- kmeans(df1[,value_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df1[,value_median], centers)
df1$warehouse_value_category <- cluster$cluster

remove(cluster)
remove(centers)

df <- merge(x = df, y = df1[,list(Agencia_ID, warehouse_item_category, warehouse_value_category)])
remove(df1)

ggplot(df) +
  geom_point(aes(x = item_sum, y = value_sum, color = warehouse_item_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("WAREHOUSE :: Deliveries by Quantity Category") +
```

```
  ylab("Value per Week") +
  xlab("Quantity per Week") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```


```
ggplot(df) +
  geom_boxplot(aes(x = warehouse_item_category, y = item_sum, group = warehouse_item_category, fill = wa
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("WAREHOUSE :: Quantity Delivered") +
  ylab("Quantity per Week") +
  xlab("Warehouse category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df) +
  geom_point(aes(x = value_sum, y = item_sum, color = warehouse_value_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("WAREHOUSE :: Deliveries by Value Category") +
  xlab("Value per Week") +
  ylab("Quantity per Week") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df) +
  geom_boxplot(aes(x = warehouse_value_category, y = value_sum, group = warehouse_value_category, fill =
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("WAREHOUSE :: Value Delivered") +
  ylab("Value per Week") +
  xlab("Warehouse category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```


```
# Saving the warehouse categories
fwrite(x = df, file = v_c_output_warehouse_sum)

remove(df)
gc()
```

```
##             used   (Mb) gc trigger   (Mb)  max used   (Mb)
## Ncells   1032735   55.2    1608248   85.9   1608248   85.9
## Vcells 532956448 4066.2  846166684 6455.8 833784052 6361.3
```

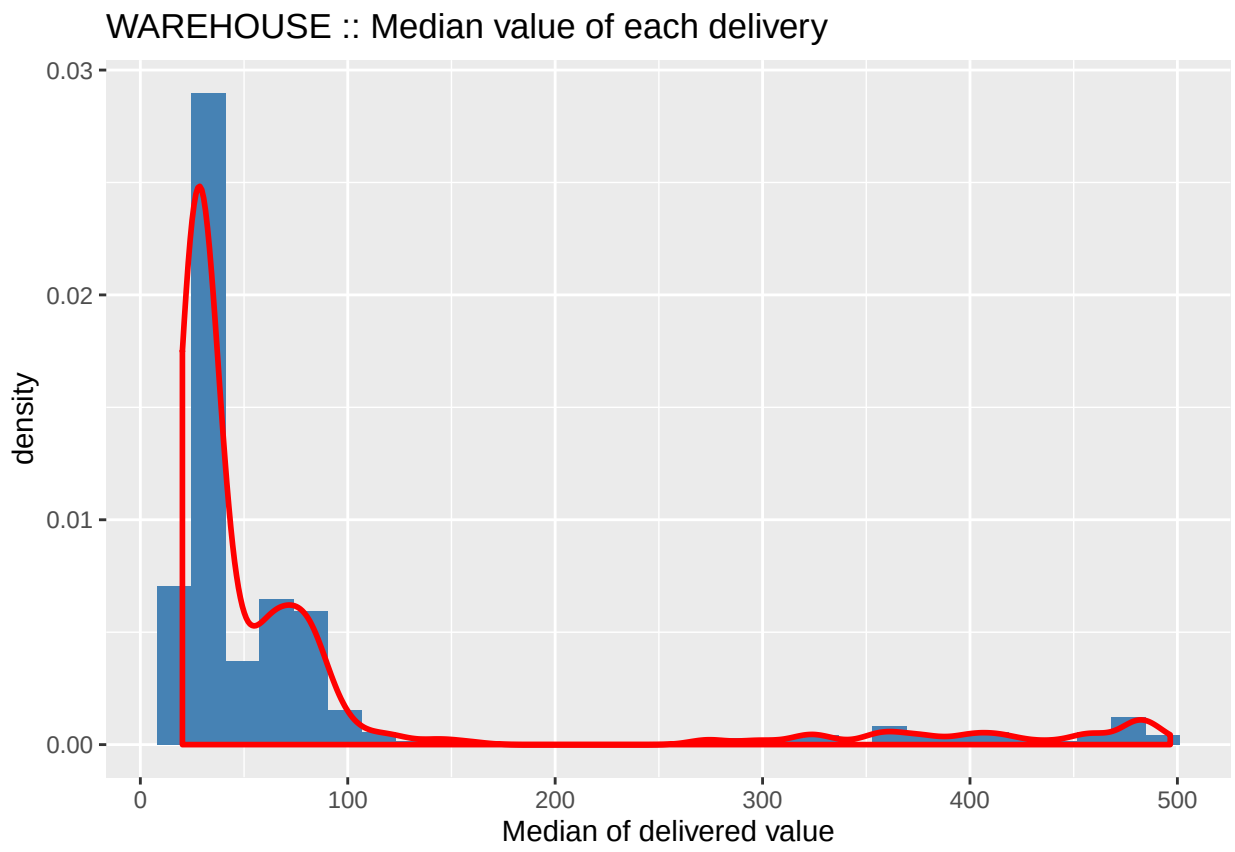
```
# Delivered value and quantity median
df <- train[, list(item_median = median(Venta_uni_hoy),
                   value_median = median(Venta_hoy)),
            by = Agencia_ID]

# What is the PESOS median of each delivery?
summary(df[,value_median])
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##     20.28     28.13     49.38    590.61    378.00 199987.20
```

```
ggplot(df[value_median > 0 & value_median <= quantile(x = df[,value_median], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
```

```
  geom_histogram(aes(x = value_median, y = ..density..), fill = "steelblue") +
  geom_density(aes(x = value_median), color = "red", size = 1) +
  ggtitle("WAREHOUSE :: Median value of each delivery") +
  xlab("Median of delivered value")
```

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

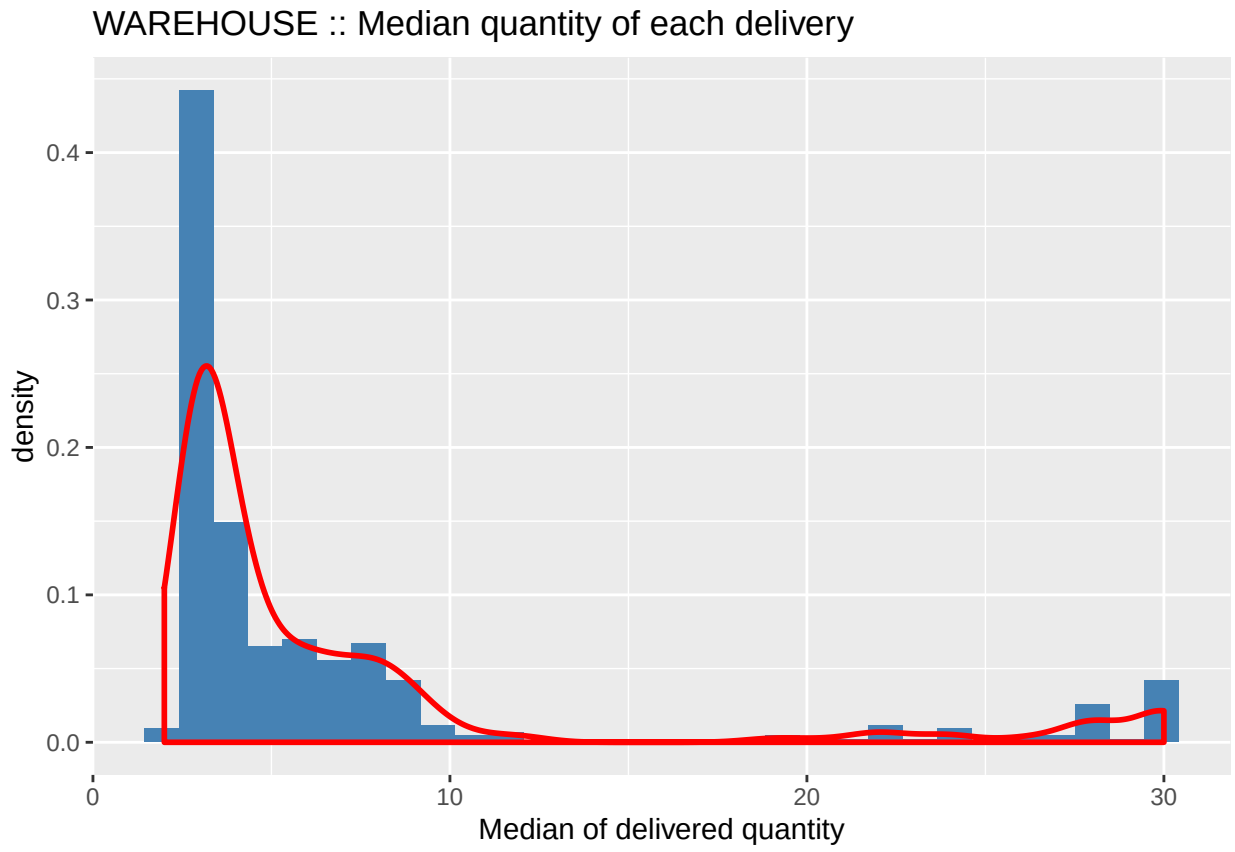



```
# What is the ITEMS median of each delivery?
summary(df[,item_median])
```

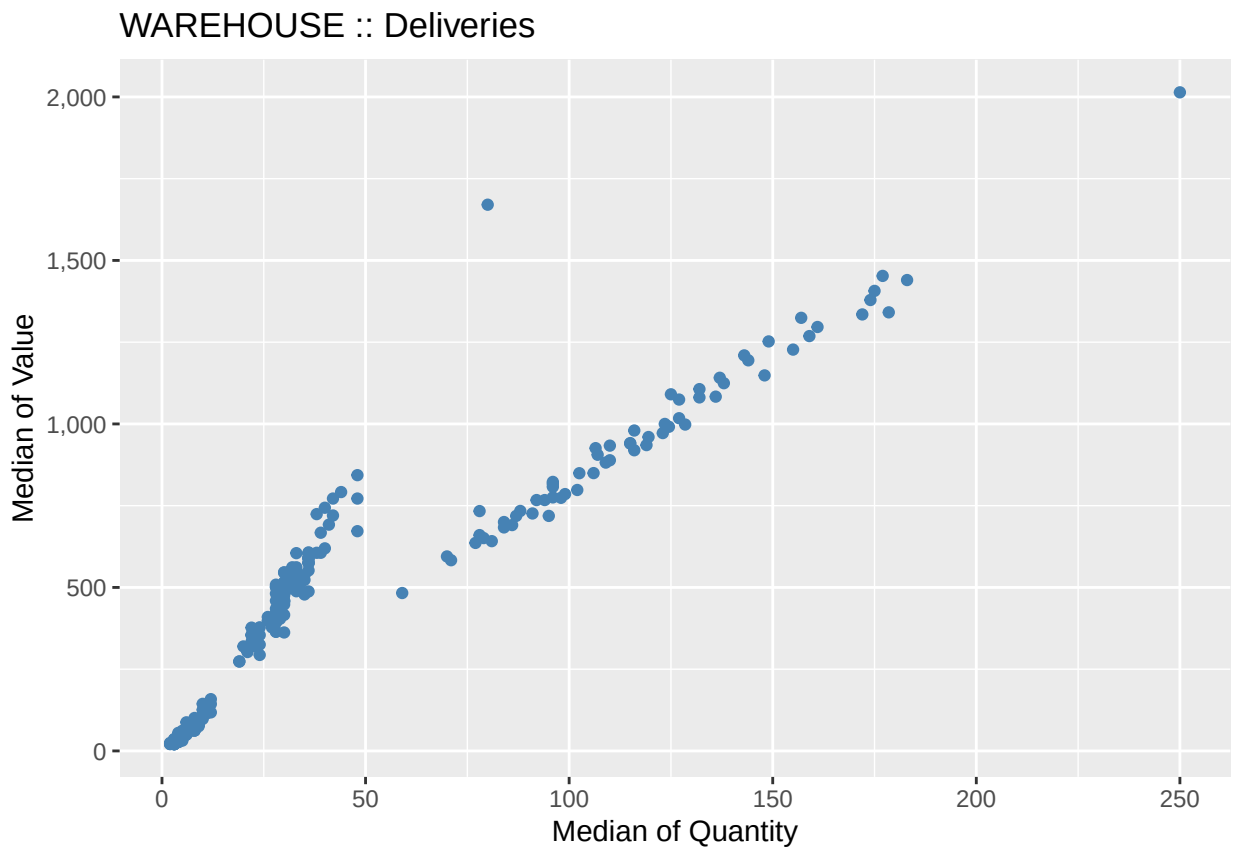
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    2.00    3.00    5.00   26.63   27.00 2304.00
```

```
ggplot(df[item_median > 0 & item_median <= quantile(x = df[,item_median], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = item_median, y = ..density..), fill = "steelblue") +
  geom_density(aes(x = item_median), color = "red", size = 1) +
  ggtitle("WAREHOUSE :: Median quantity of each delivery") +
  xlab("Median of delivered quantity")
```

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

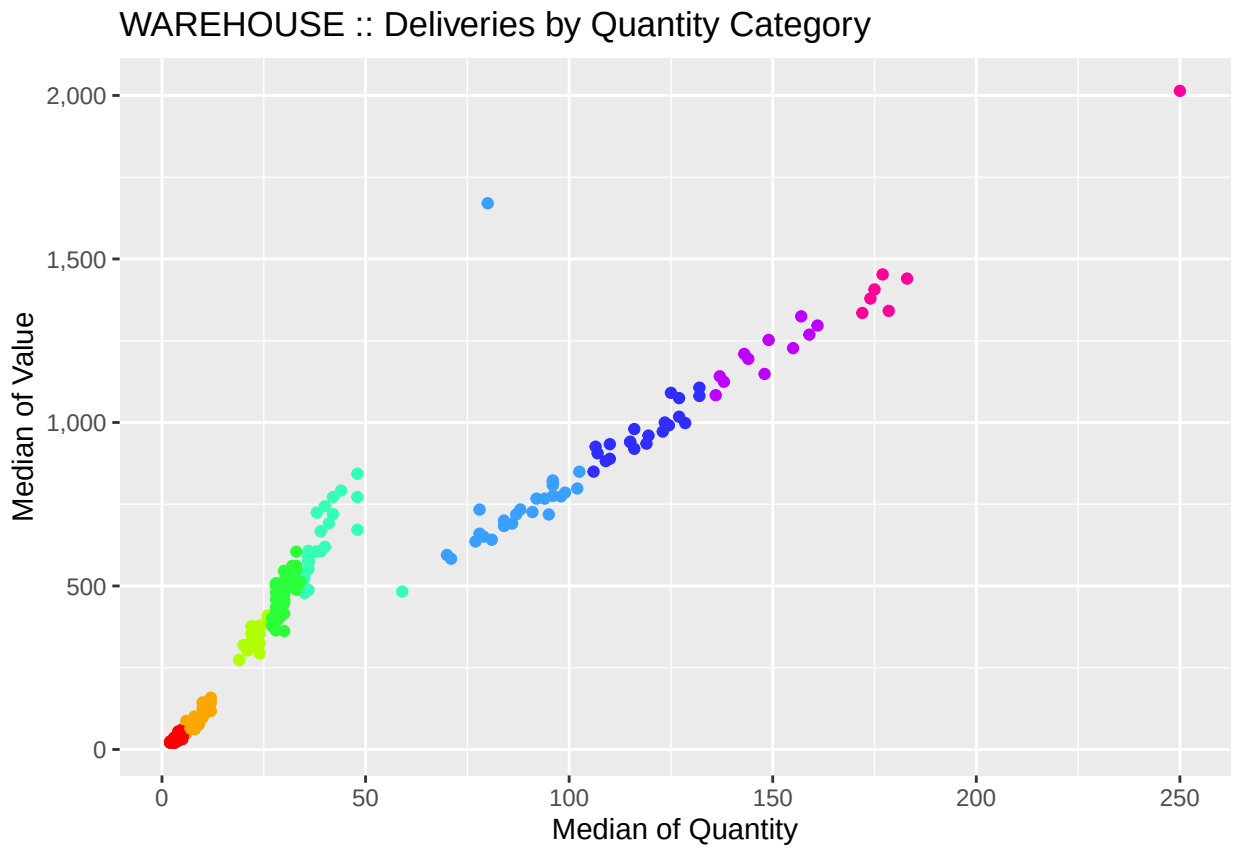



```
m <- mean(df[,item_median])
sd <- sd(df[,item_median])

ggplot(df[item_median >= (m - 3 * sd) & item_median <= (m + 3 * sd)]) +
  geom_point(aes(x = item_median, y = value_median), color = "steelblue") +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("WAREHOUSE :: Deliveries") +
  ylab("Median of Value") +
  xlab("Median of Quantity")
```

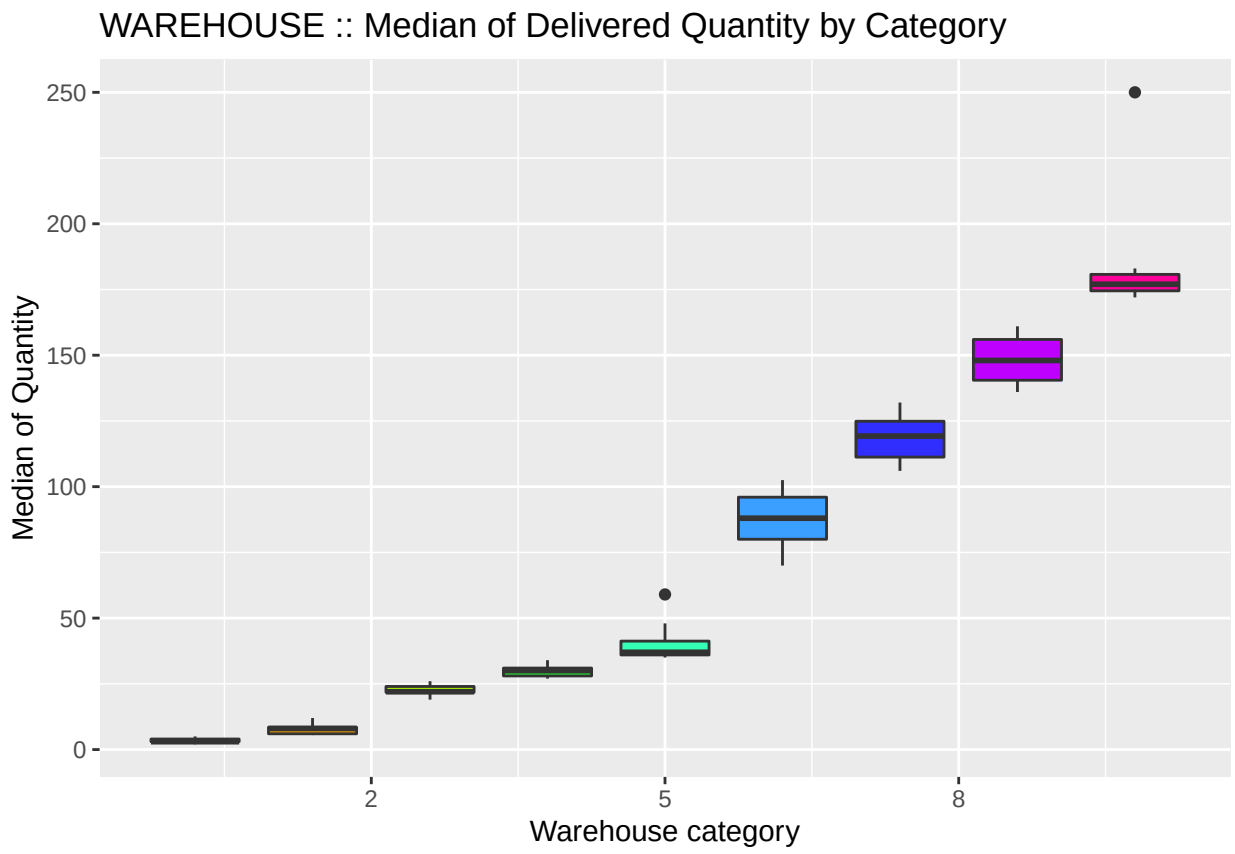



```
remove(m)
remove(sd)

# Warehouse category
cluster <- kmeans(df[,item_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df[,item_median], centers)
df$warehouse_item_median_category <- cluster$cluster

cluster <- kmeans(df[,value_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df[,value_median], centers)
df$warehouse_value_median_category <- cluster$cluster

remove(cluster)
remove(centers)

m <- mean(df[,item_median])
sd <- sd(df[,item_median])

ggplot(df[item_median >= (m - 3 * sd) & item_median <= (m + 3 * sd)]) +
  geom_point(aes(x = item_median, y = value_median, color = warehouse_item_median_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("WAREHOUSE :: Deliveries by Quantity Category") +
  ylab("Median of Value") +
```

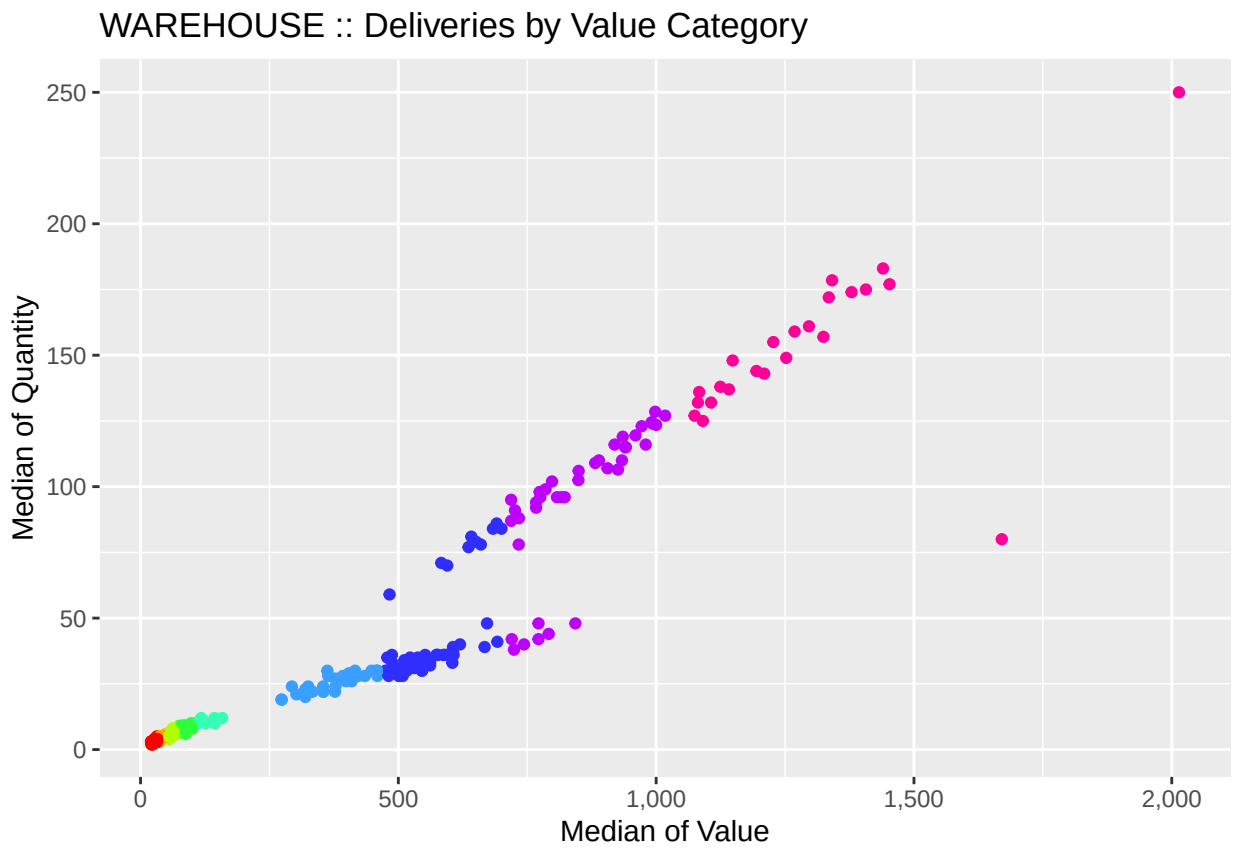
```
  xlab("Median of Quantity") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```



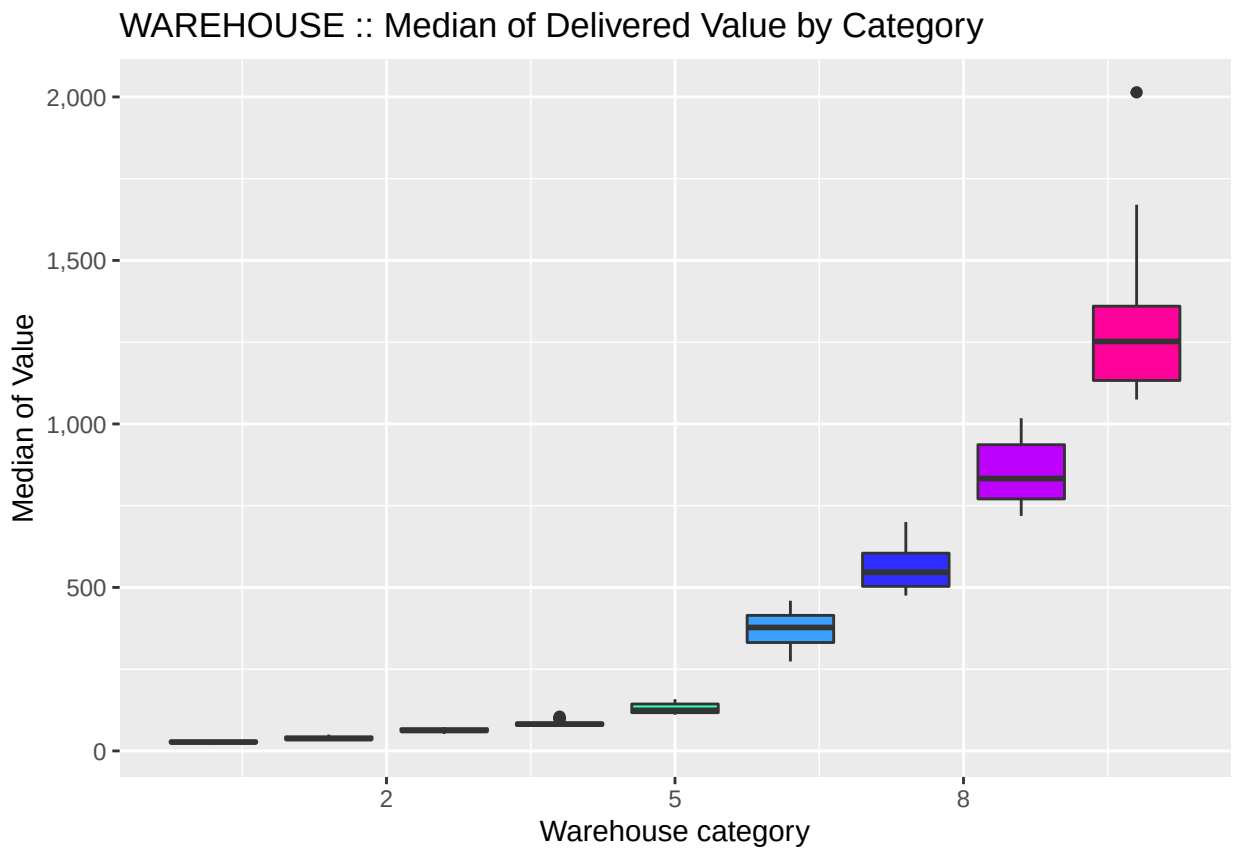

```
ggplot(df[item_median >= (m - 3 * sd) & item_median <= (m + 3 * sd)]) +
  geom_boxplot(aes(x = warehouse_item_median_category, y = item_median, group = warehouse_item_median_ca
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("WAREHOUSE :: Median of Delivered Quantity by Category") +
  ylab("Median of Quantity") +
  xlab("Warehouse category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
m <- mean(df[,value_median])
sd <- sd(df[,value_median])

ggplot(df[value_median >= (m - 3 * sd) & value_median <= (m + 3 * sd)]) +
  geom_point(aes(x = value_median, y = item_median, color = warehouse_value_median_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("WAREHOUSE :: Deliveries by Value Category") +
  ylab("Median of Quantity") +
  xlab("Median of Value") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df[value_median >= (m - 3 * sd) & value_median <= (m + 3 * sd)]) +
  geom_boxplot(aes(x = warehouse_value_median_category, y = value_median, group = warehouse_value_median
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("WAREHOUSE :: Median of Delivered Value by Category") +
  ylab("Median of Value") +
  xlab("Warehouse category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
fwrite(x = df, file = v_c_output_warehouse_median)

remove(df)
gc()
```

```
##             used   (Mb) gc trigger   (Mb)  max used   (Mb)
## Ncells   1171210   62.6    2159170  115.4   1608248   85.9
## Vcells 533255088 4068.5  846166684 6455.8 833784052 6361.3
```

```
#
# ROUTES ANALISYS
#
# How many routes BIMBO uses?
paste("The BIMBO group uses", length(unique(train[,Ruta_SAK])), "routes")
```

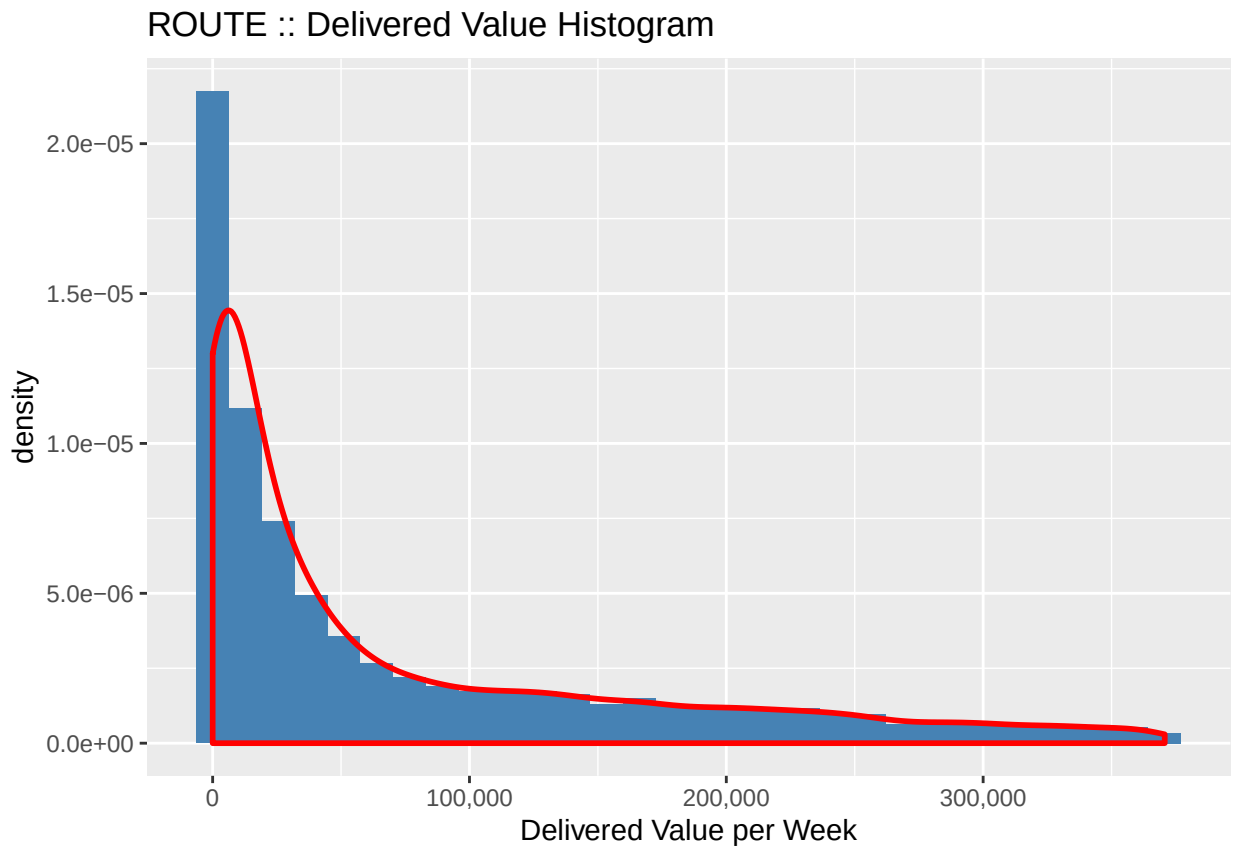
```
## [1] "The BIMBO group uses 3603 routes"
```

```
# How many items did each route deliver per week?
df <- train[, list(item_sum = sum(Venta_uni_hoy),
                   value_sum = sum(Venta_hoy)),
            by = list(Ruta_SAK, Semana)]

# How much PESOS did each route deliver per week?
summary(df[,value_sum])
```

```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##        0    8916   57765  252679  268845 4714049
```

```
ggplot(df[value_sum > 0 & value_sum <= quantile(x = df[,value_sum], probs = 0.8)]) + #[value_sum <= 500
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = value_sum, y = ..density..), fill = "steelblue") +
  geom_density(aes(x = value_sum), color = "red", size = 1) +
  ggtitle("ROUTE :: Delivered Value Histogram") +
  xlab("Delivered Value per Week")
```

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



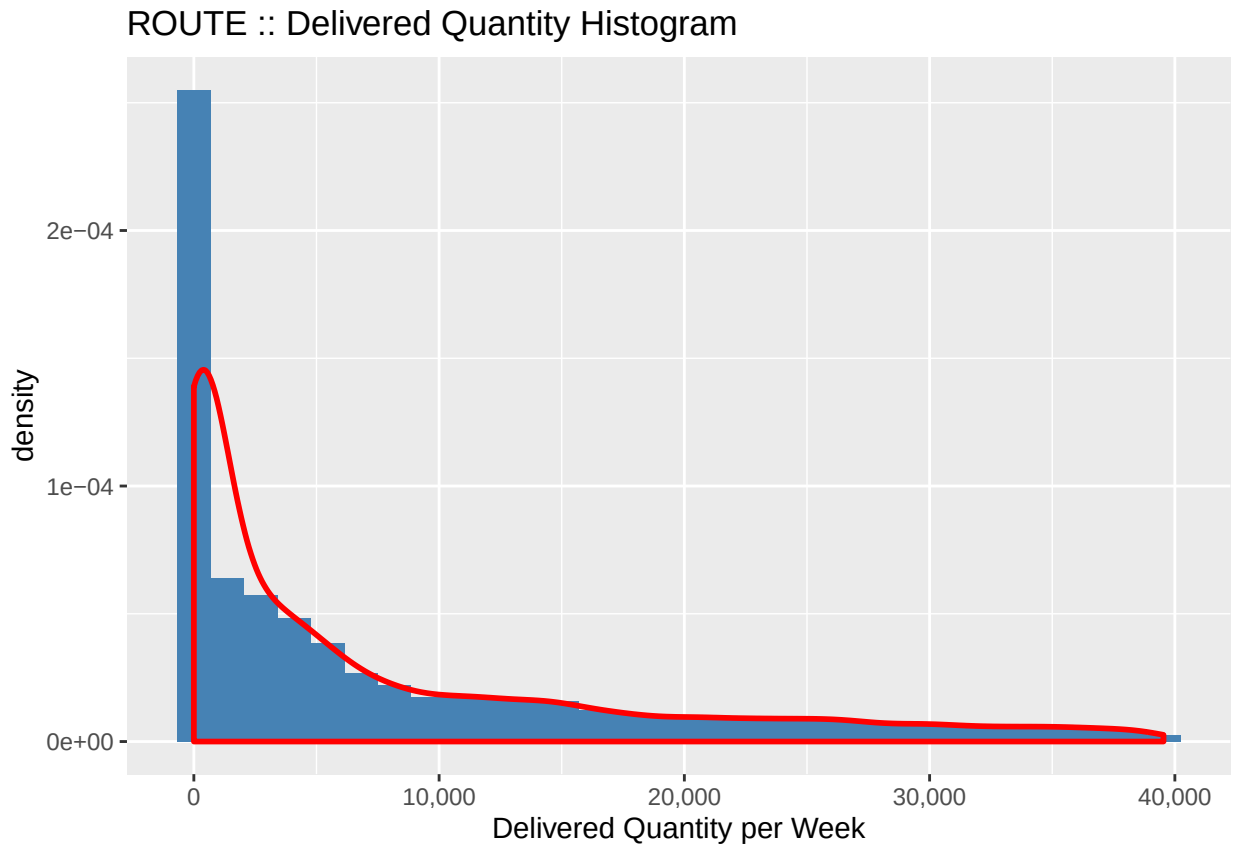

```
# How many items did each route deliver per week?
summary(df[,item_sum])
```

```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##        0     436    5929   26948   28890  407733
```

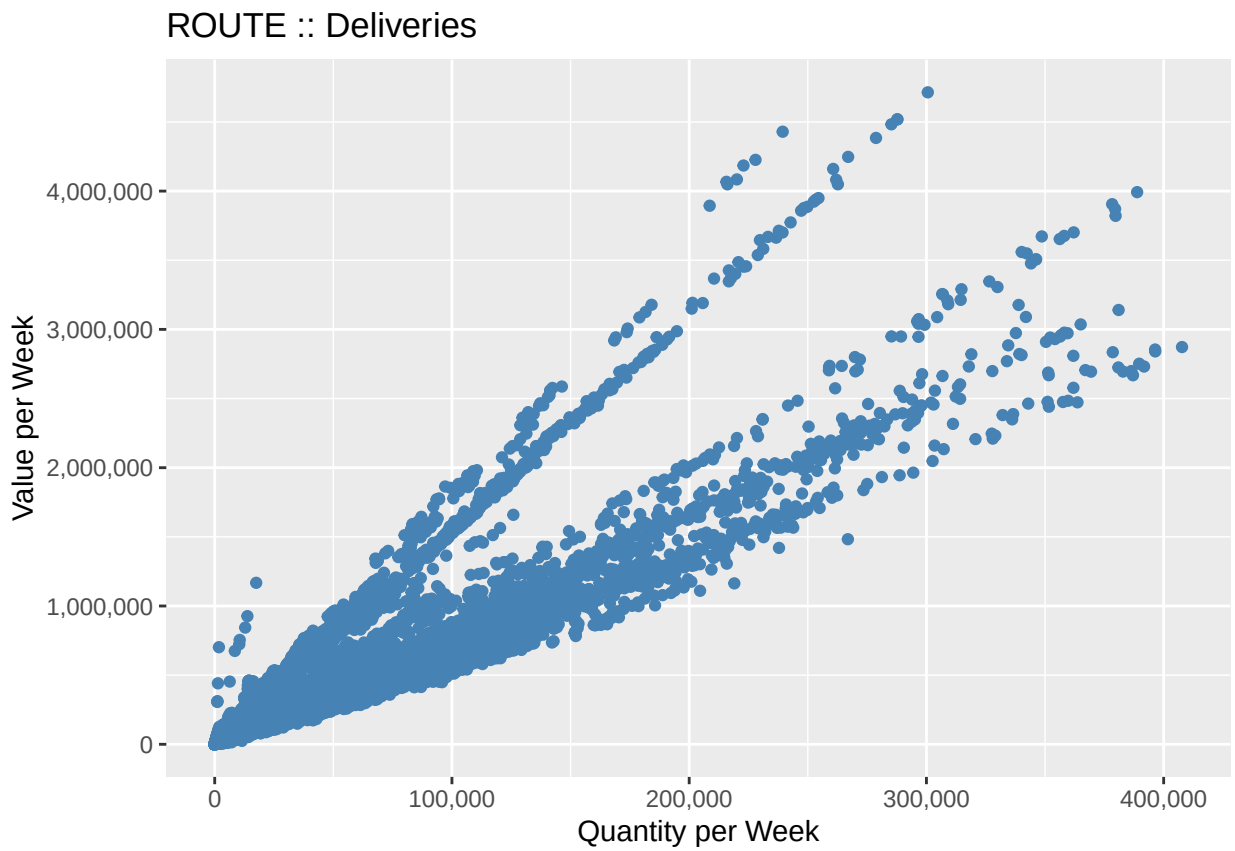
```
ggplot(df[item_sum > 0 & item_sum <= quantile(x = df[,item_sum], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = item_sum, y = ..density..), fill = "steelblue") +
  geom_density(aes(x = item_sum), color = "red", size = 1) +
```

```
  ggtitle("ROUTE :: Delivered Quantity Histogram") +
  xlab("Delivered Quantity per Week")
```

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

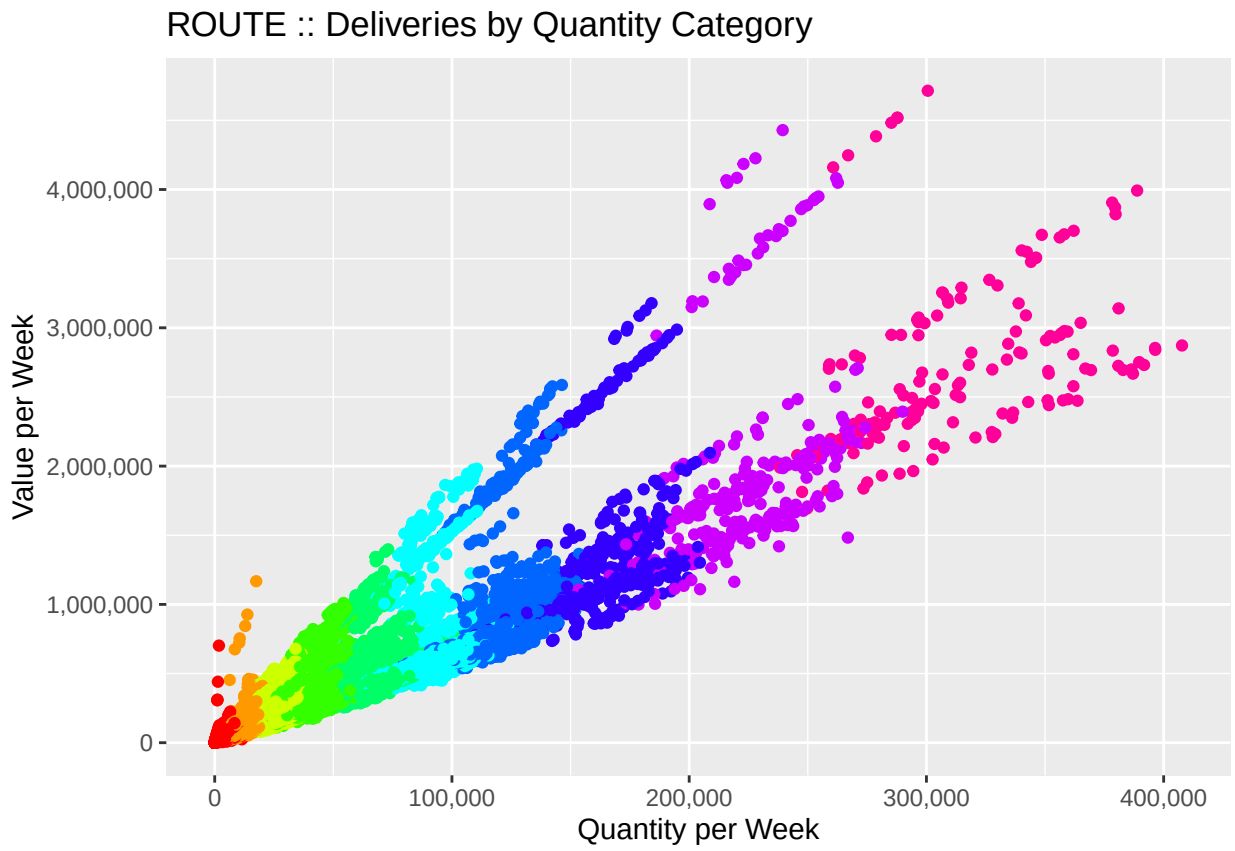



```
ggplot(df) +
  geom_point(aes(x = item_sum, y = value_sum), color = "steelblue") +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("ROUTE :: Deliveries") +
  ylab("Value per Week") +
  xlab("Quantity per Week")
```

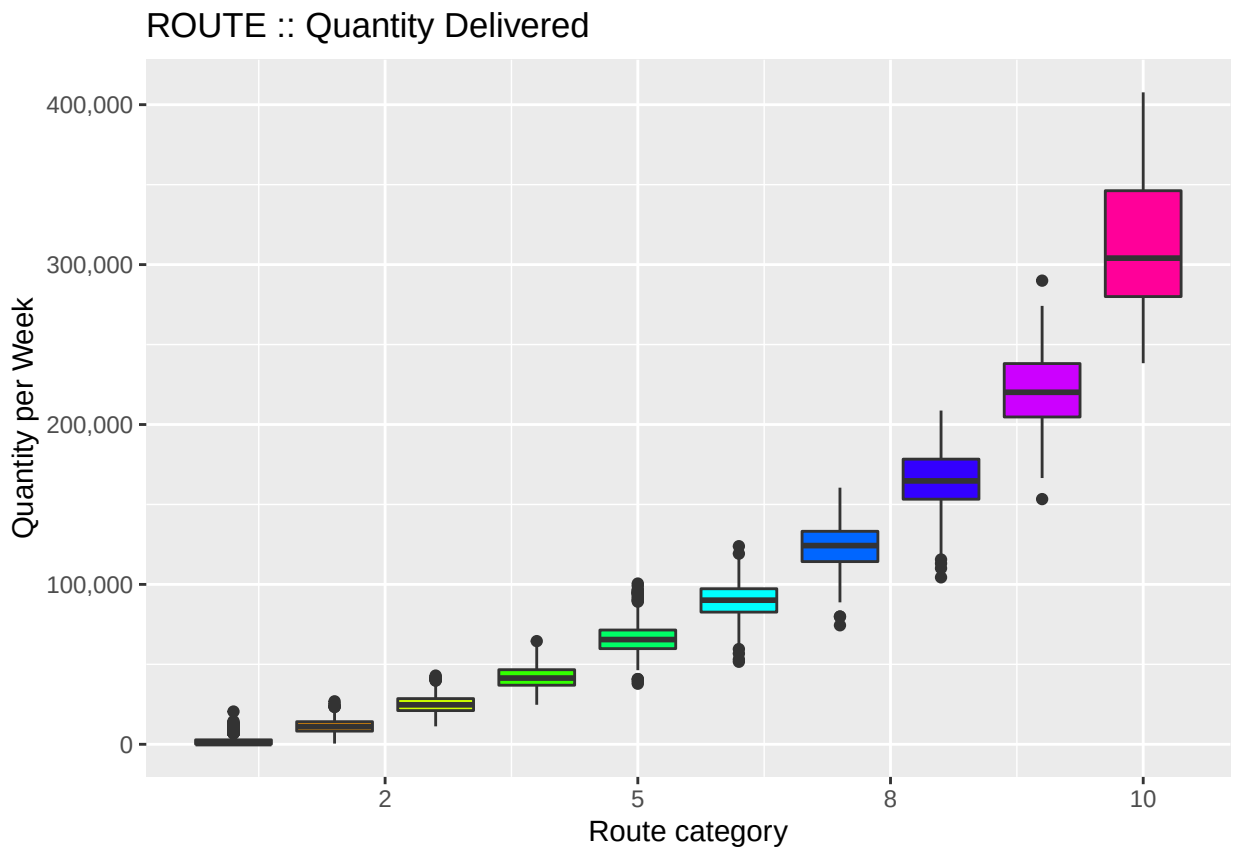



```
# Route category
df1 = df[, list(item_median = median(item_sum),
                value_median = median(value_sum)),
         by = Ruta_SAK]

cluster <- kmeans(df1[,item_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df1[,item_median], centers)
df1$route_item_category <- cluster$cluster

cluster <- kmeans(df1[,value_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df1[,value_median], centers)
df1$route_value_category <- cluster$cluster

remove(cluster)
remove(centers)

df <- merge(x = df, y = df1[,list(Ruta_SAK, route_item_category, route_value_category)])
remove(df1)

ggplot(df) +
  geom_point(aes(x = item_sum, y = value_sum, color = route_item_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("ROUTE :: Deliveries by Quantity Category") +
```

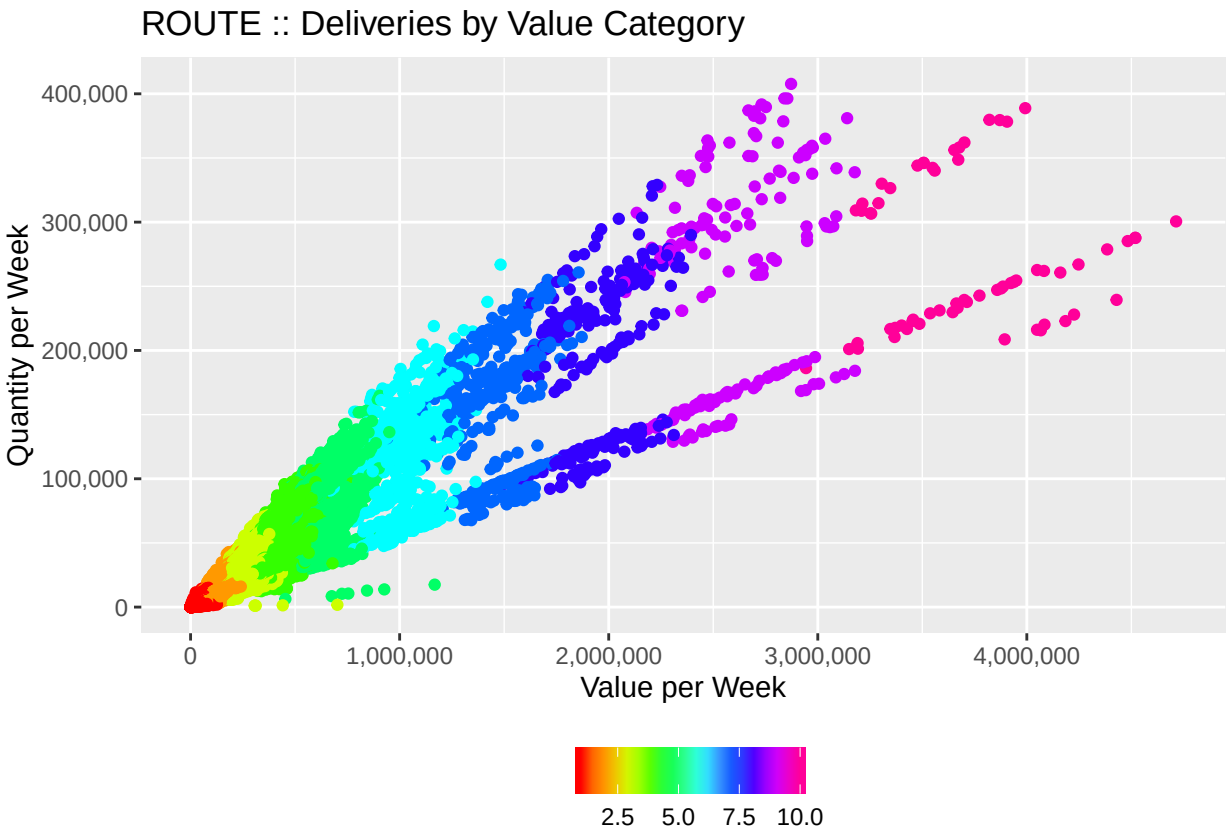
```
  ylab("Value per Week") +
  xlab("Quantity per Week") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```



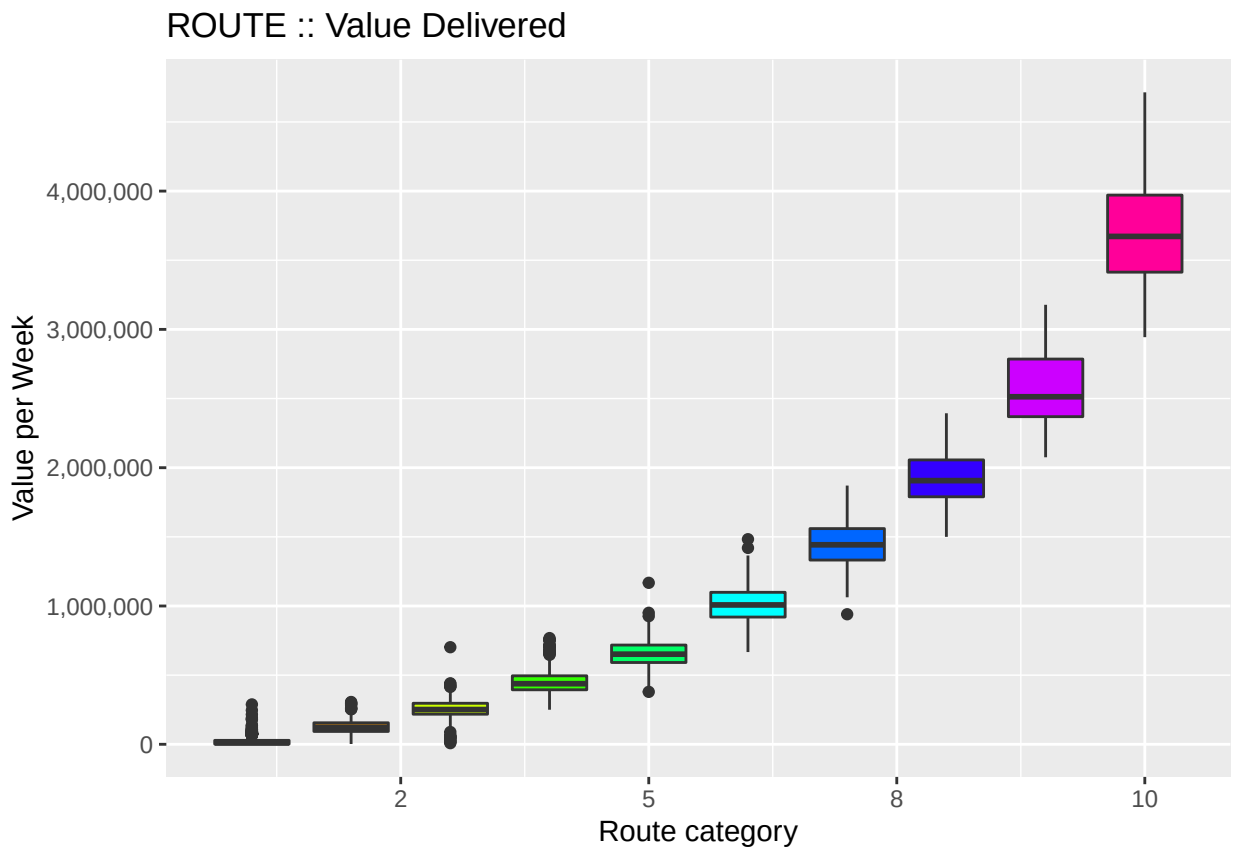

```
ggplot(df) +
  geom_boxplot(aes(x = route_item_category, y = item_sum, group = route_item_category, fill = route_iter
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("ROUTE :: Quantity Delivered") +
  ylab("Quantity per Week") +
  xlab("Route category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df) +
  geom_point(aes(x = value_sum, y = item_sum, color = route_value_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("ROUTE :: Deliveries by Value Category") +
  ylab("Quantity per Week") +
  xlab("Value per Week") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position = "bottom", legend.title = element_blank())
```




```
ggplot(df) +
  geom_boxplot(aes(x = route_value_category, y = value_sum, group = route_value_category, fill = route_
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("ROUTE :: Value Delivered") +
  ylab("Value per Week") +
  xlab("Route category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
# Saving the route categories into train dataset
# train <- merge(x = train, y = df[,c("Ruta_SAK","route_value_category", "route_item_category")], by.x
fwrite(x = df, file = v_c_output_route_sum)

remove(df)
gc()
```

```
##            used   (Mb) gc trigger   (Mb)  max used   (Mb)
## Ncells   1325551   70.8    2159170  115.4   2159170  115.4
## Vcells 534586996 4078.6  846166684 6455.8 834671739 6368.1
```

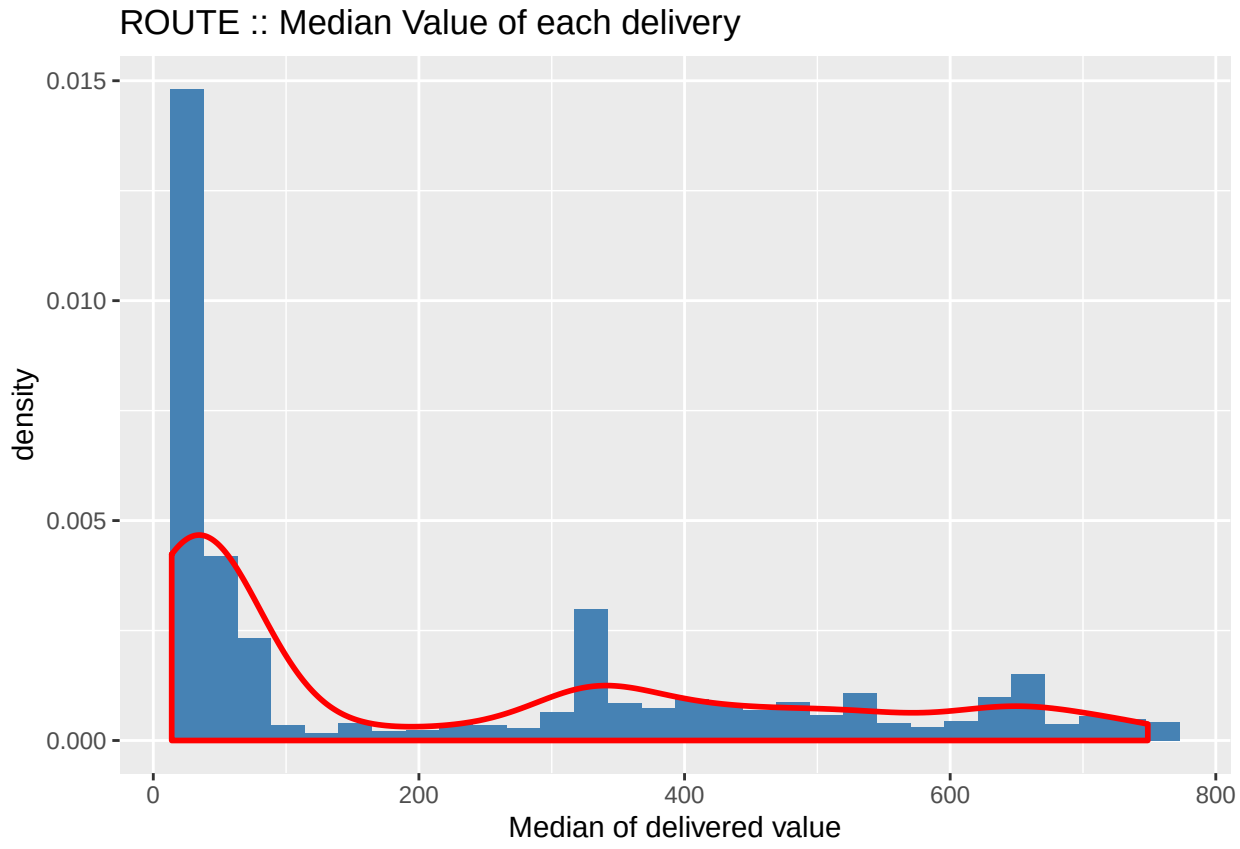
```
# Unit sales median
df <- train[, list(item_median = median(Venta_uni_hoy),
                   value_median = median(Venta_hoy)),
            by = Ruta_SAK]

# What is the PESOS median of each delivery?
summary(df[,value_median])
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##     0.00    31.43   326.10   901.95   658.80 68347.50
```

```
ggplot(df[value_median > 0 & value_median <= quantile(x = df[,value_median], probs = 0.8)]) +
  scale_y_continuous() +
```

```
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = value_median, y = ..density..), fill = "steelblue") +
  geom_density(aes(x = value_median), color = "red", size = 1) +
  ggtitle("ROUTE :: Median Value of each delivery") +
  xlab("Median of delivered value")
```

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

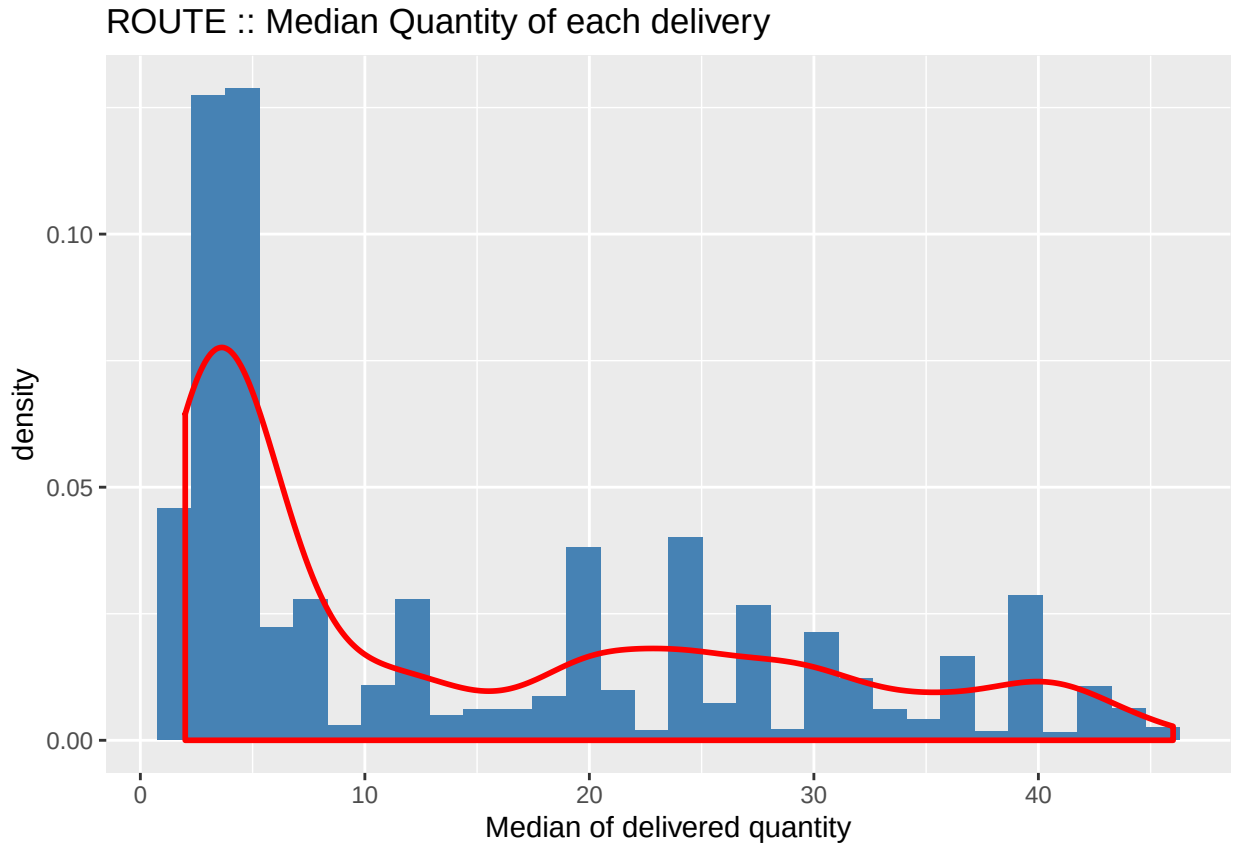



```
# What is the ITEMS median of each delivery?
summary(df[,item_median])
```

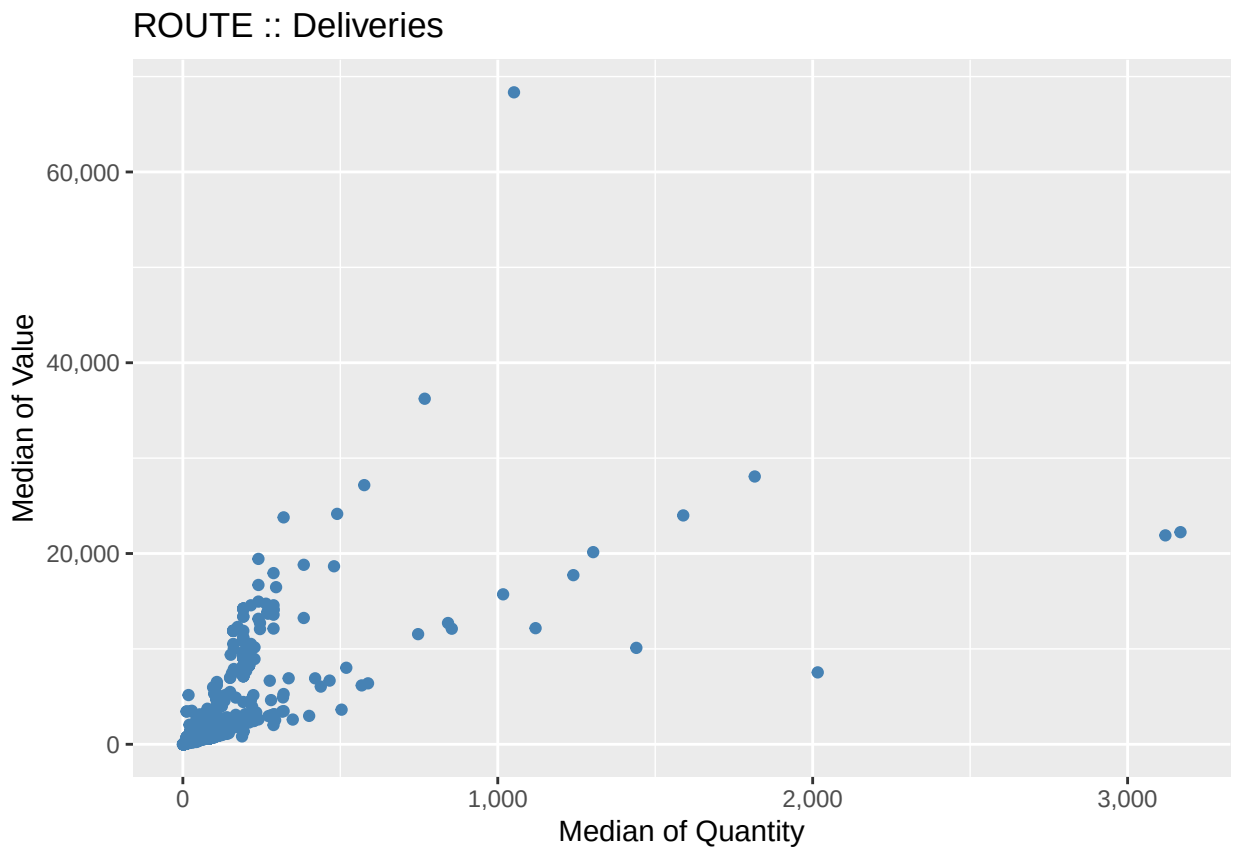
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    4.00   18.00   39.87   40.00 3168.00
```

```
ggplot(df[item_median > 0 & item_median <= quantile(x = df[,item_median], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = item_median, y = ..density..), fill = "steelblue") +
  geom_density(aes(x = item_median), color = "red", size = 1) +
  ggtitle("ROUTE :: Median Quantity of each delivery") +
  xlab("Median of delivered quantity")
```

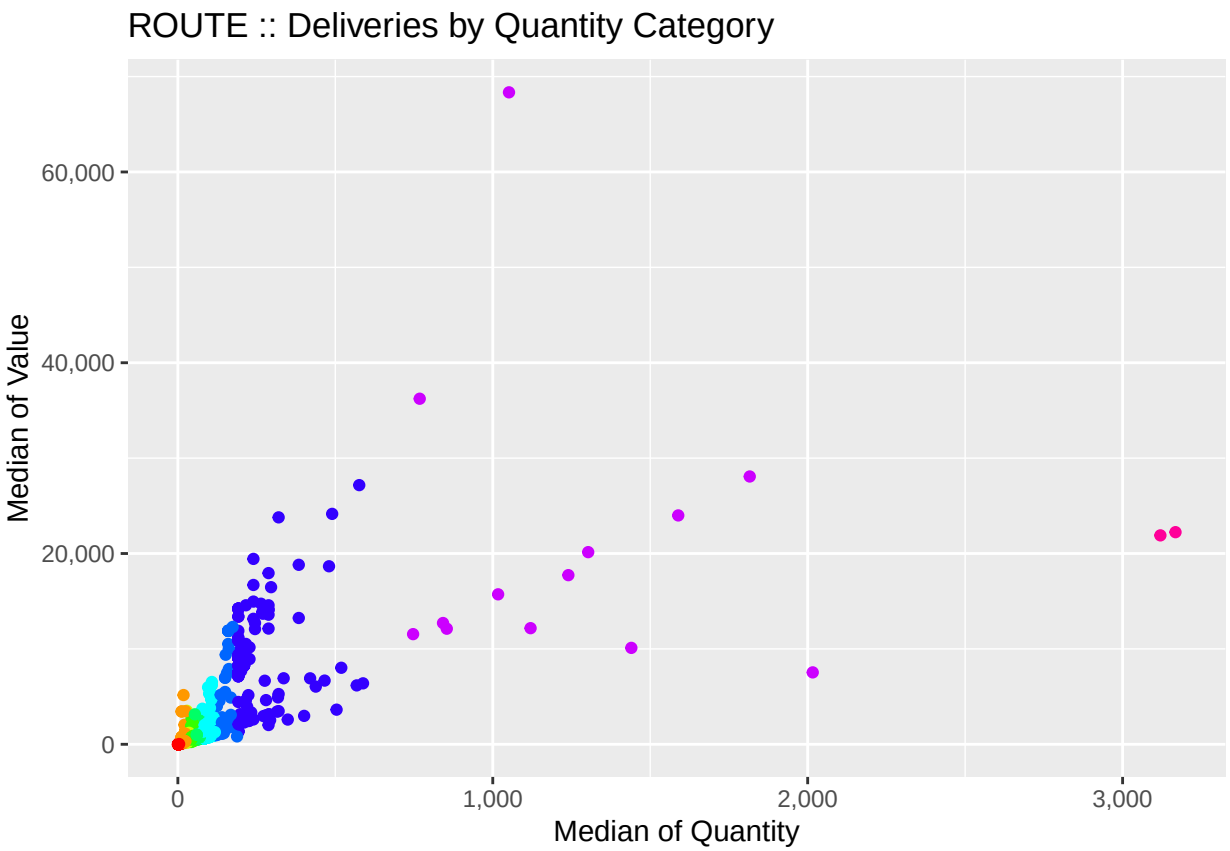
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



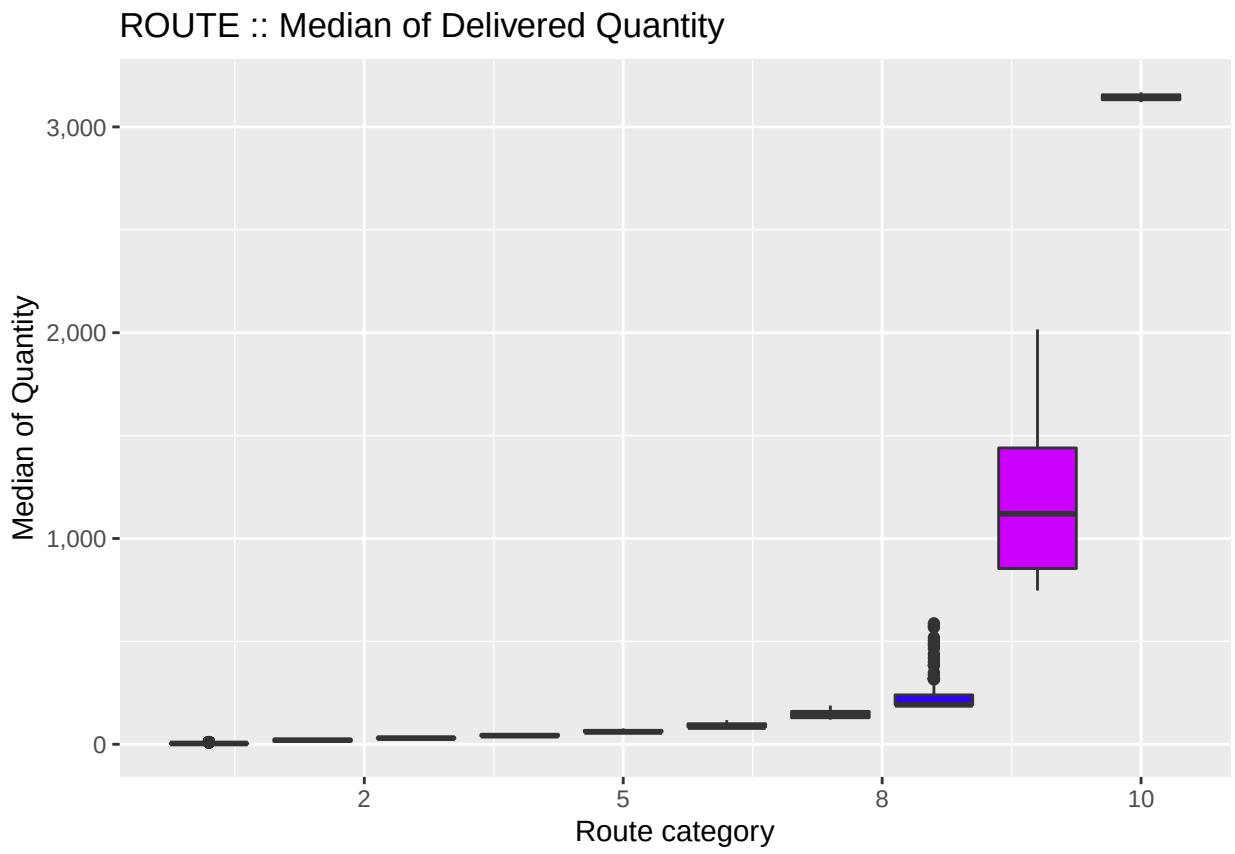

```
ggplot(df) +
  geom_point(aes(x = item_median, y = value_median), color = "steelblue") +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("ROUTE :: Deliveries") +
  ylab("Median of Value") +
  xlab("Median of Quantity")
```



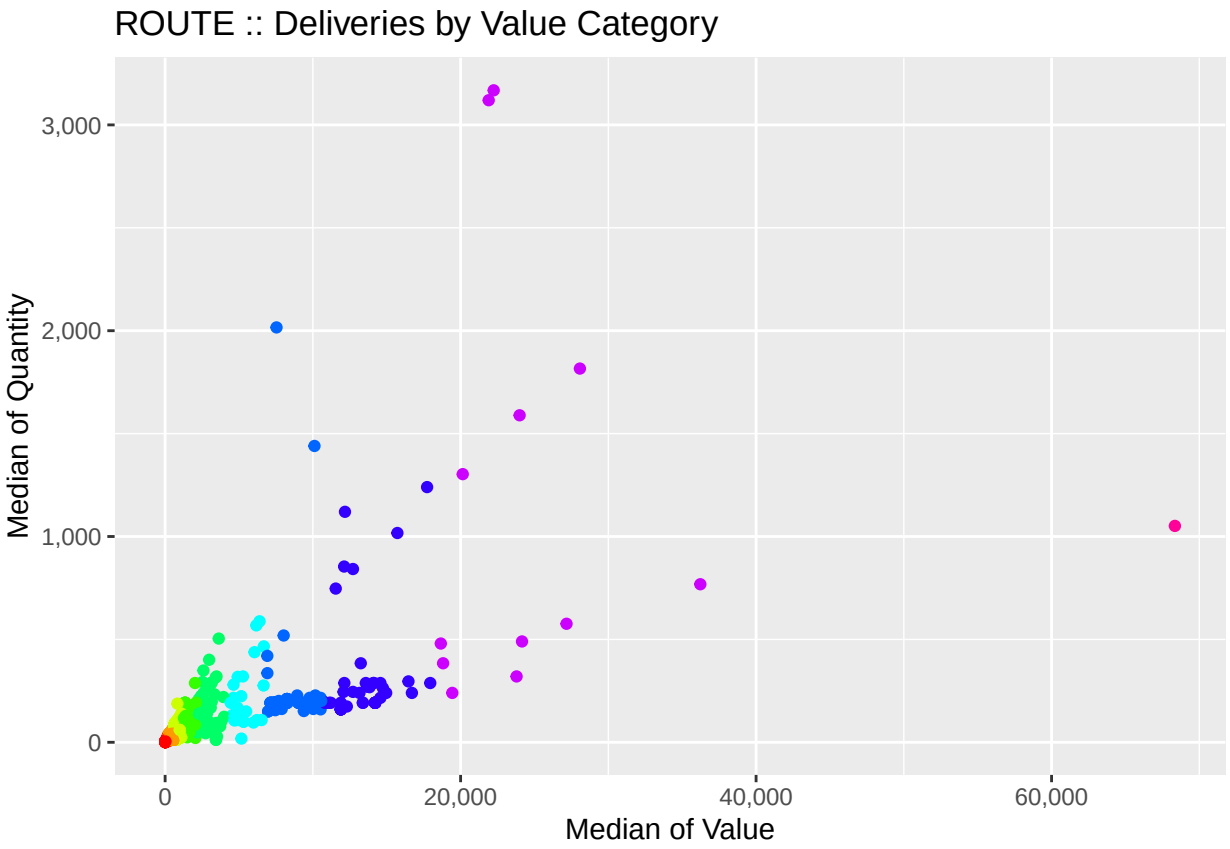

```
# Route category
cluster <- kmeans(df[,item_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df[,item_median], centers)
df$route_item_median_category <- cluster$cluster

cluster <- kmeans(df[,value_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df[,value_median], centers)
df$route_value_median_category <- cluster$cluster

remove(cluster)
remove(centers)

ggplot(df) +
  geom_point(aes(x = item_median, y = value_median, color = route_item_median_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("ROUTE :: Deliveries by Quantity Category") +
  ylab("Median of Value") +
  xlab("Median of Quantity") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```



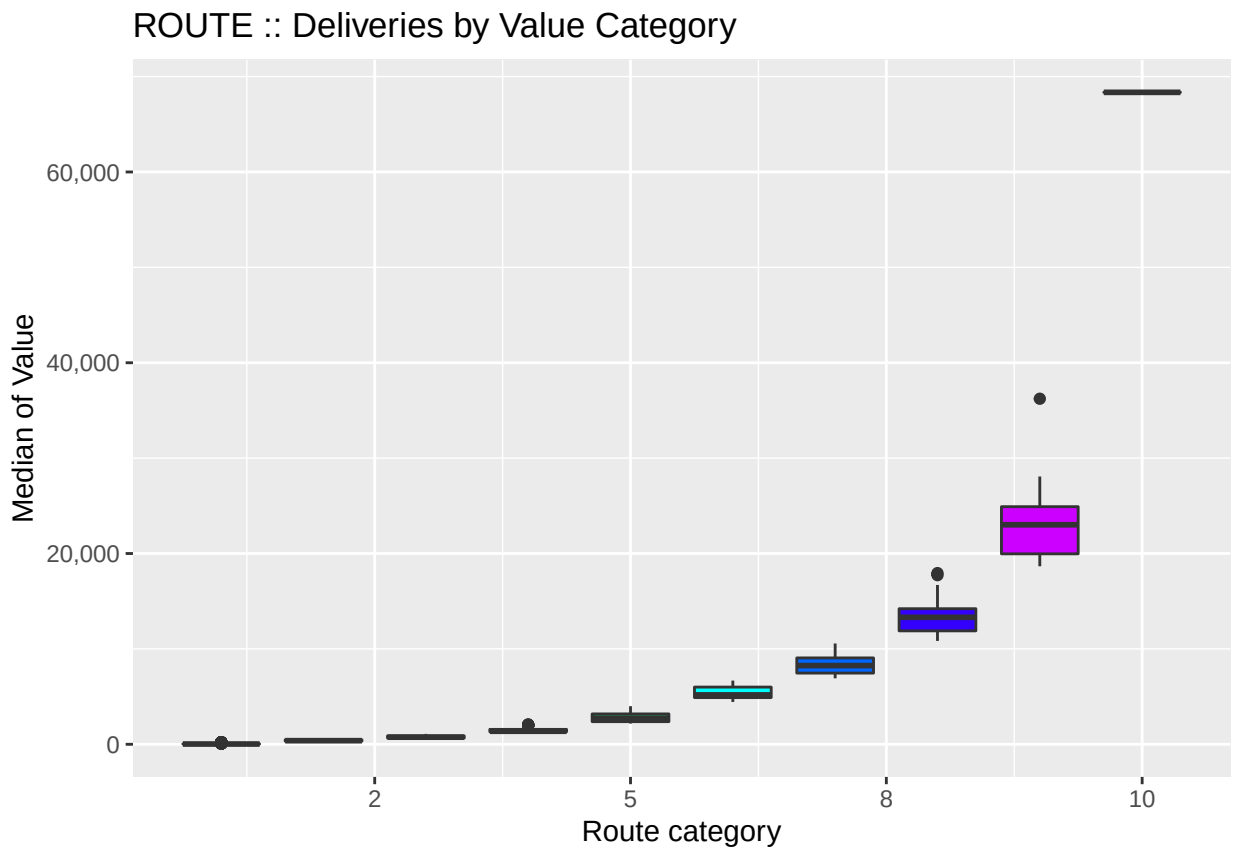

```
ggplot(df) +
  geom_boxplot(aes(x = route_item_median_category, y = item_median, group = route_item_median_category,
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("ROUTE :: Median of Delivered Quantity") +
  ylab("Median of Quantity") +
  xlab("Route category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df) +
  geom_point(aes(x = value_median, y = item_median, color = route_value_median_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("ROUTE :: Deliveries by Value Category") +
  ylab("Median of Quantity") +
  xlab("Median of Value") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df) +
  geom_boxplot(aes(x = route_value_median_category, y = value_median, group = route_value_median_categor
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("ROUTE :: Deliveries by Value Category") +
  ylab("Median of Value") +
  xlab("Route category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
# train <- merge(x = train, y = df[,c("Ruta_SAK","route_value_median_category", "route_item_median_cate
fwrite(x = df, file = v_c_output_route_median)

remove(df)
gc()
```

```
##             used   (Mb) gc trigger    (Mb)  max used    (Mb)
## Ncells   1463089   78.2    2631004  140.6   2159170  115.4
## Vcells 534859238 4080.7  846166684 6455.8 834671739 6368.1
```

```
#
# CUSTOMER ANALISYS
#
# How many customers BIMBO has?
paste("The BIMBO group has", length(unique(train[,Cliente_ID])), "customers")
```

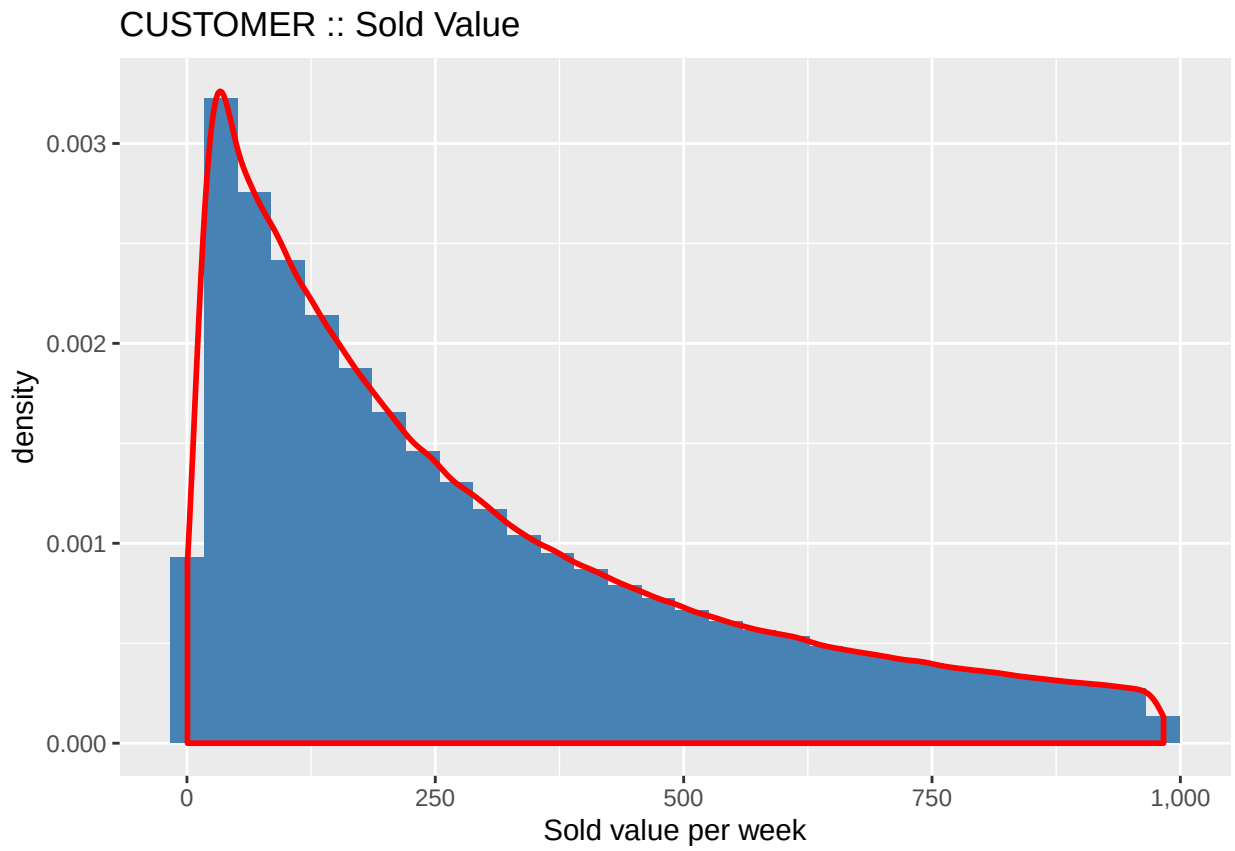
```
## [1] "The BIMBO group has 880604 customers"
```

```
# How many items each customer sell?
df <- train[, list(item_sum = sum(Venta_uni_hoy),
                   value_sum = sum(Venta_hoy)),
            by = list(Cliente_ID, Semana)]

# How much PESOS did each customer sell?
summary(df[,value_sum])
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##         0       117       307       962       783  23184774
```

```
ggplot(df[value_sum > 0 & value_sum <= quantile(x = df[,value_sum], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = value_sum, y = ..density..), fill = "steelblue") +
  geom_density(aes(x = value_sum), color = "red", size = 1) +
  ggtitle("CUSTOMER :: Sold Value") +
  xlab("Sold value per week")
```

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



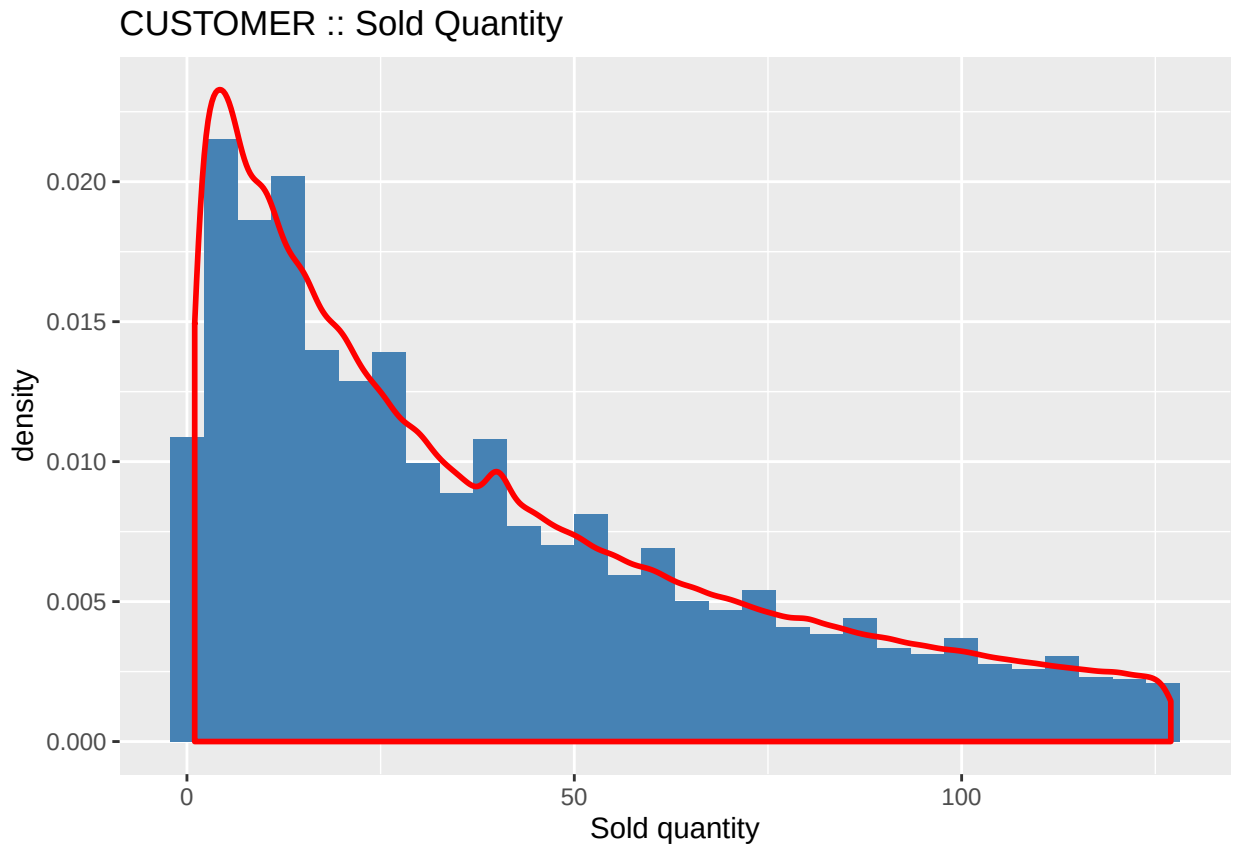

```
# How many ITEMS did each customer deliver?
summary(df[,item_sum])
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##       0.0      16.0      42.0     102.5     103.0 2806918.0
```

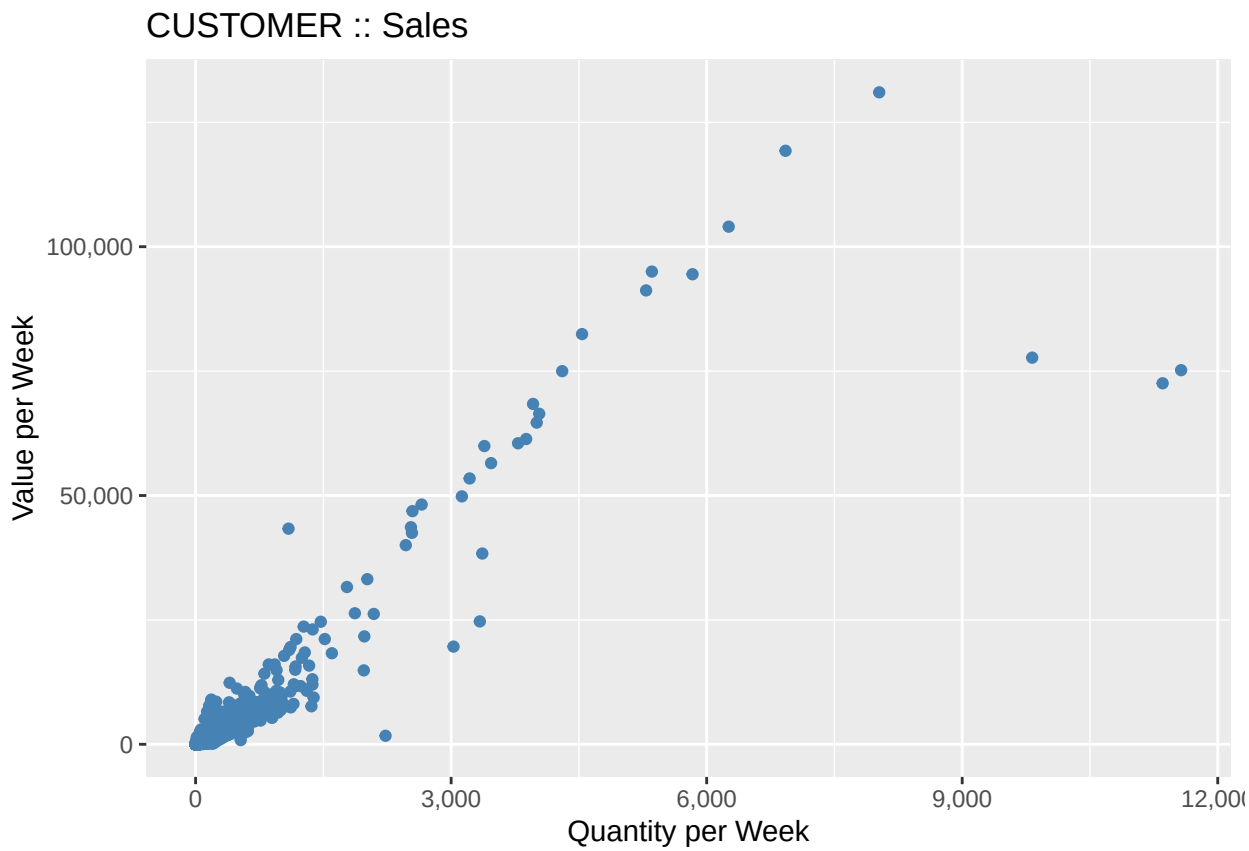
```
ggplot(df[item_sum > 0 & item_sum <= quantile(x = df[,item_sum], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = item_sum, y = ..density..), fill = "steelblue") +
  geom_density(aes(x = item_sum), color = "red", size = 1) +
```

```
  ggtitle("CUSTOMER :: Sold Quantity") +
  xlab("Sold quantity")
```

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




```
# To much points .. will need a sample

df_sample <- df[sample(nrow(df), 10000)]

ggplot(df_sample) +
  geom_point(aes(x = item_sum, y = value_sum), color = "steelblue") +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("CUSTOMER :: Sales") +
  ylab("Value per Week") +
  xlab("Quantity per Week")
```

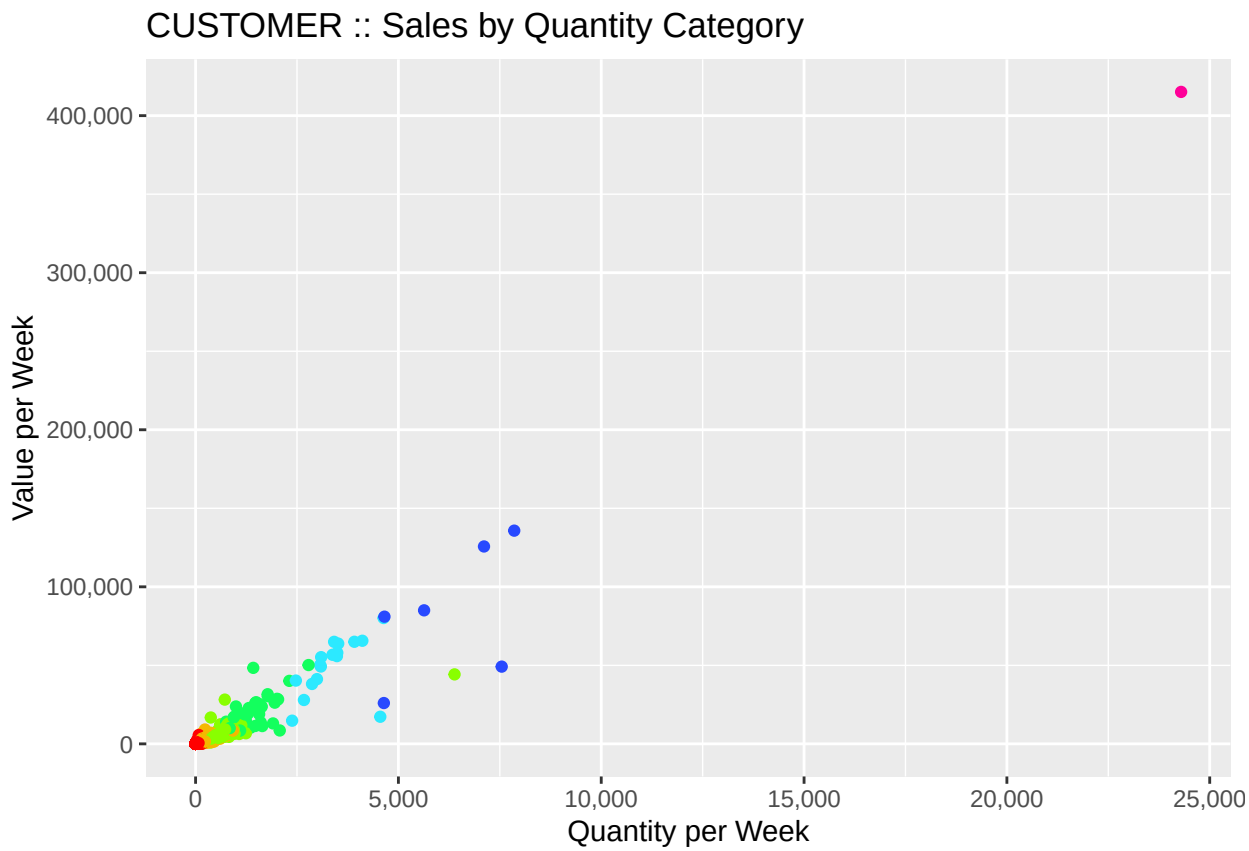



```
# customer category
df1 <- df[, list(item_median = median(item_sum),
                 value_median = median(value_sum)),
          by = Cliente_ID]

cluster <- kmeans(df1[,item_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df1[,item_median], centers)
df1$customer_item_category <- cluster$cluster

cluster <- kmeans(df1[,value_median], 10)
```

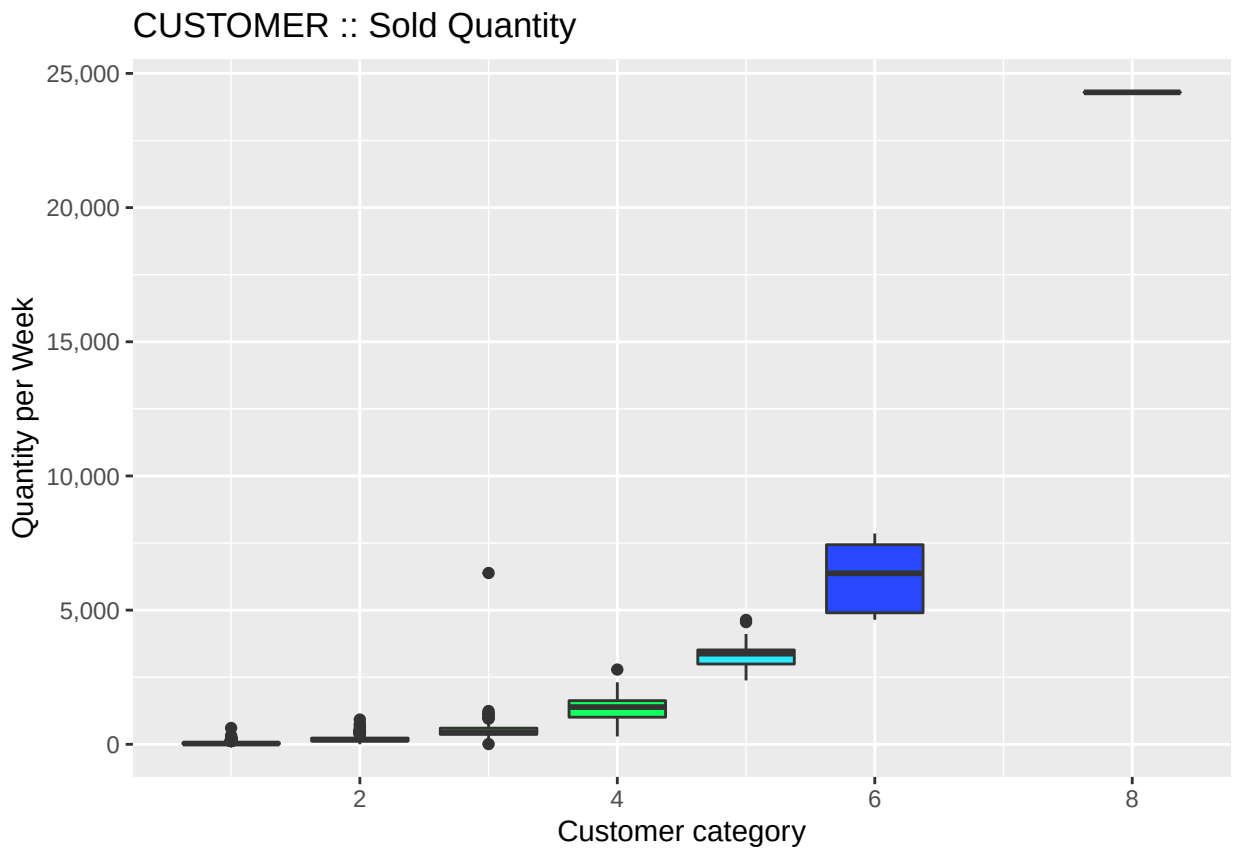
```
## Warning: Quick-TRANSfer stage steps exceeded maximum (= 44030200)
```

```
centers <- sort(cluster$centers)
cluster <- kmeans(df1[,value_median], centers)
df1$customer_value_category <- cluster$cluster

remove(cluster)
remove(centers)

df <- merge(x = df, y = df1[,list(Cliente_ID, customer_item_category, customer_value_category)])

remove(df1)
```

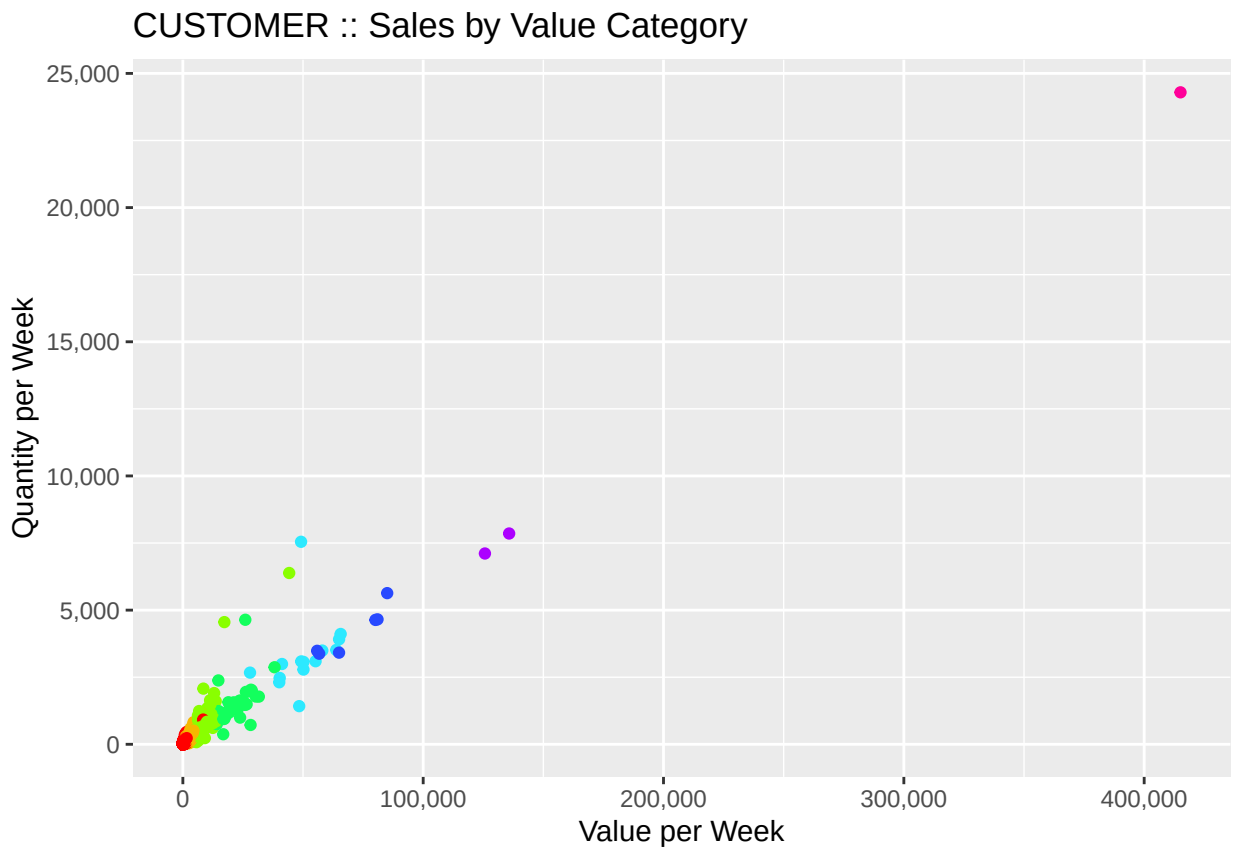
```
df_sample <- df[sample(nrow(df), 10000)]

ggplot(df_sample) +
  geom_point(aes(x = item_sum, y = value_sum, color = customer_item_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("CUSTOMER :: Sales by Quantity Category") +
  ylab("Value per Week") +
  xlab("Quantity per Week") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```



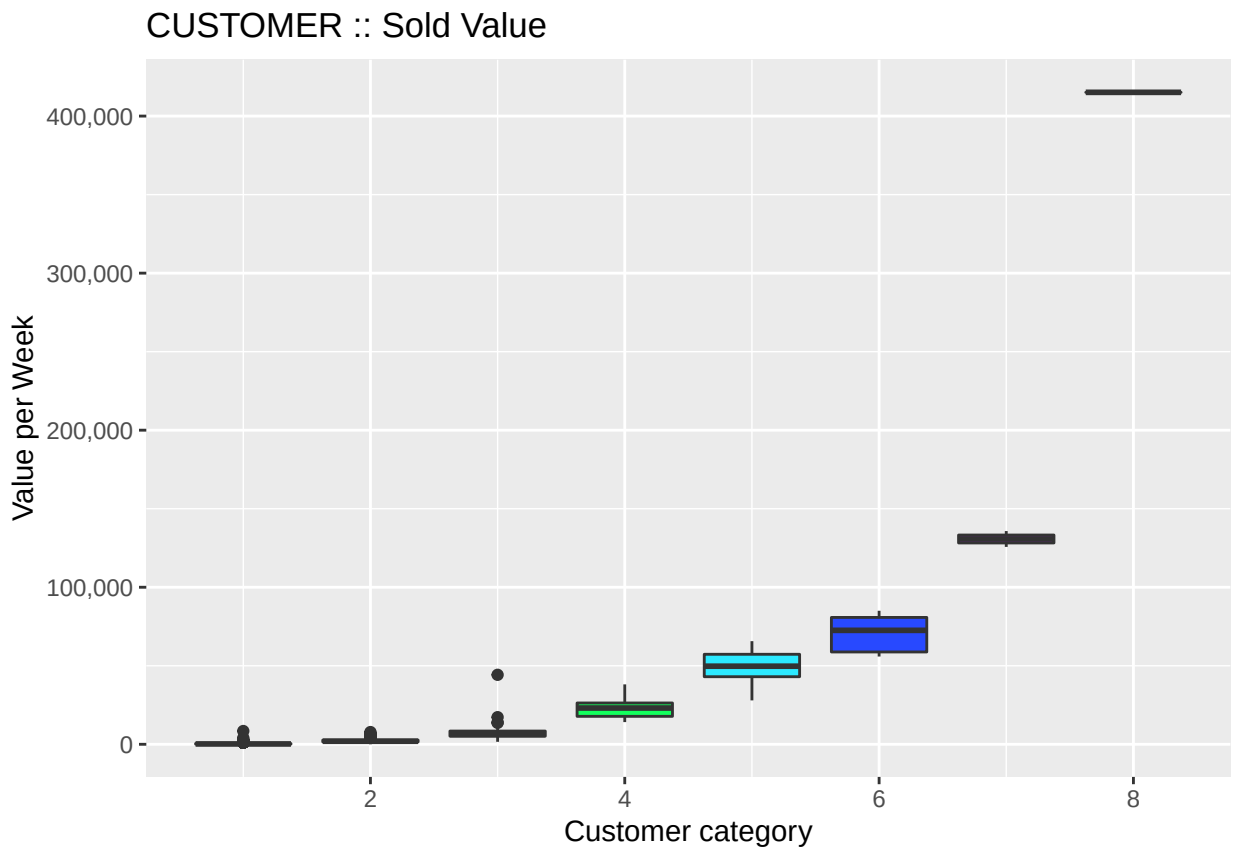

```
ggplot(df_sample) +
  geom_boxplot(aes(x = customer_item_category, y = item_sum, group = customer_item_category, fill = cus
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("CUSTOMER :: Sold Quantity") +
  ylab("Quantity per Week") +
  xlab("Customer category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df_sample) +
  geom_point(aes(x = value_sum, y = item_sum, color = customer_value_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("CUSTOMER :: Sales by Value Category") +
  ylab("Quantity per Week") +
  xlab("Value per Week") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df_sample) +
  geom_boxplot(aes(x = customer_value_category, y = value_sum, group = customer_value_category, fill = c
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("CUSTOMER :: Sold Value") +
  ylab("Value per Week") +
  xlab("Customer category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
remove(df_sample)

# Saving the customer categories into train dataset
# train <- merge(x = train, y = df[,c("Cliente_ID","customer_value_category", "customer_item_category")],
fwrite(x = df, file = v_c_output_customer_sum)
remove(df)
gc()
```

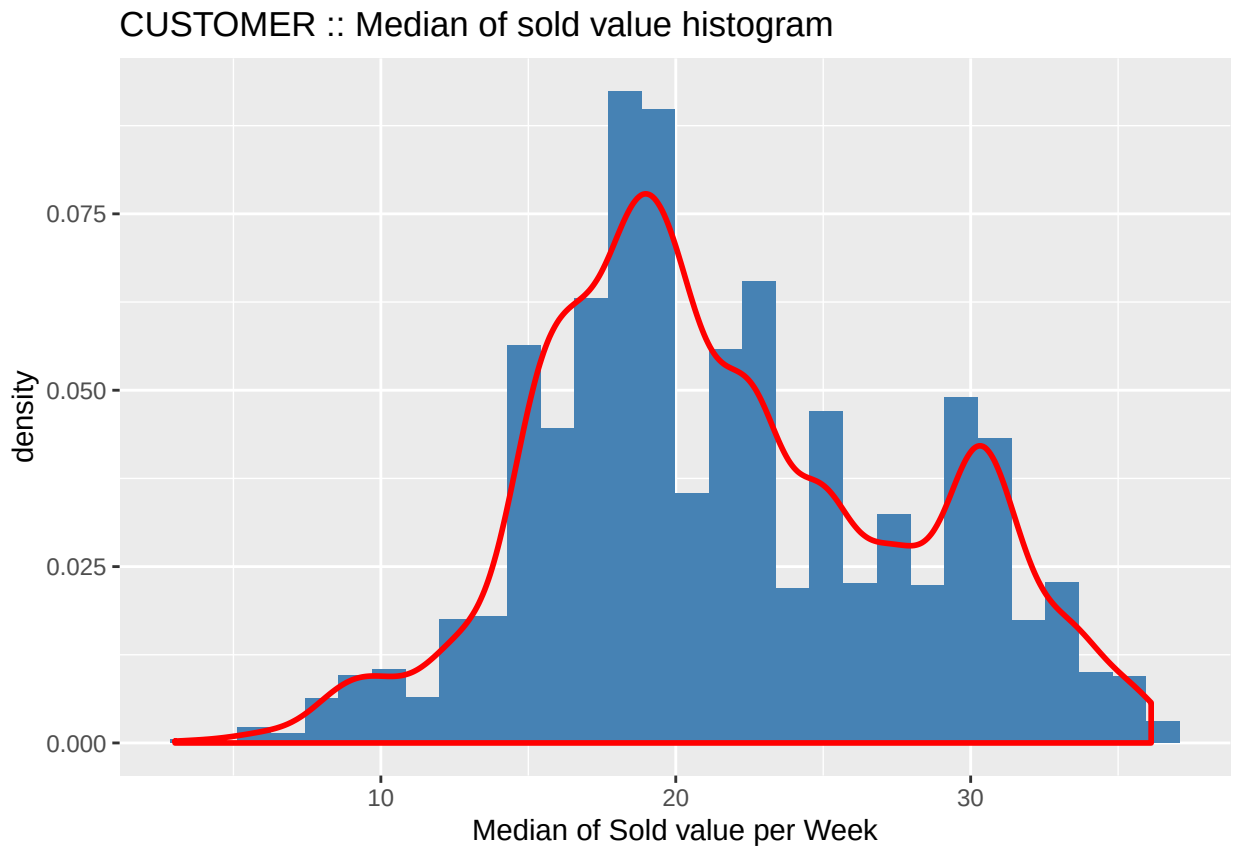
```
##             used   (Mb) gc trigger   (Mb)  max used   (Mb)
## Ncells   1601204   85.6    2631004  140.6   2631004  140.6
## Vcells 535638538 4086.6  847872332 6468.8 847872332 6468.8
```

```
# Unit sales median
df <- train[, list(item_median = median(Venta_uni_hoy),
                   value_median = median(Venta_hoy)),
            by = Cliente_ID]

df_sample <- df[sample(nrow(df), 10000)]

# What is the PESOS median of each sale?
summary(df[,value_median])
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##      0.00     18.24     22.91     45.09     31.68 258944.00
```

```
ggplot(df_sample[value_median > 0 & value_median <= quantile(x = df[,value_median], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = value_median, y = ..density..), fill = "steelblue") +
  geom_density(aes(x = value_median), color = "red", size = 1) +
  ggtitle("CUSTOMER :: Median of sold value histogram") +
  xlab("Median of Sold value per Week")
```

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

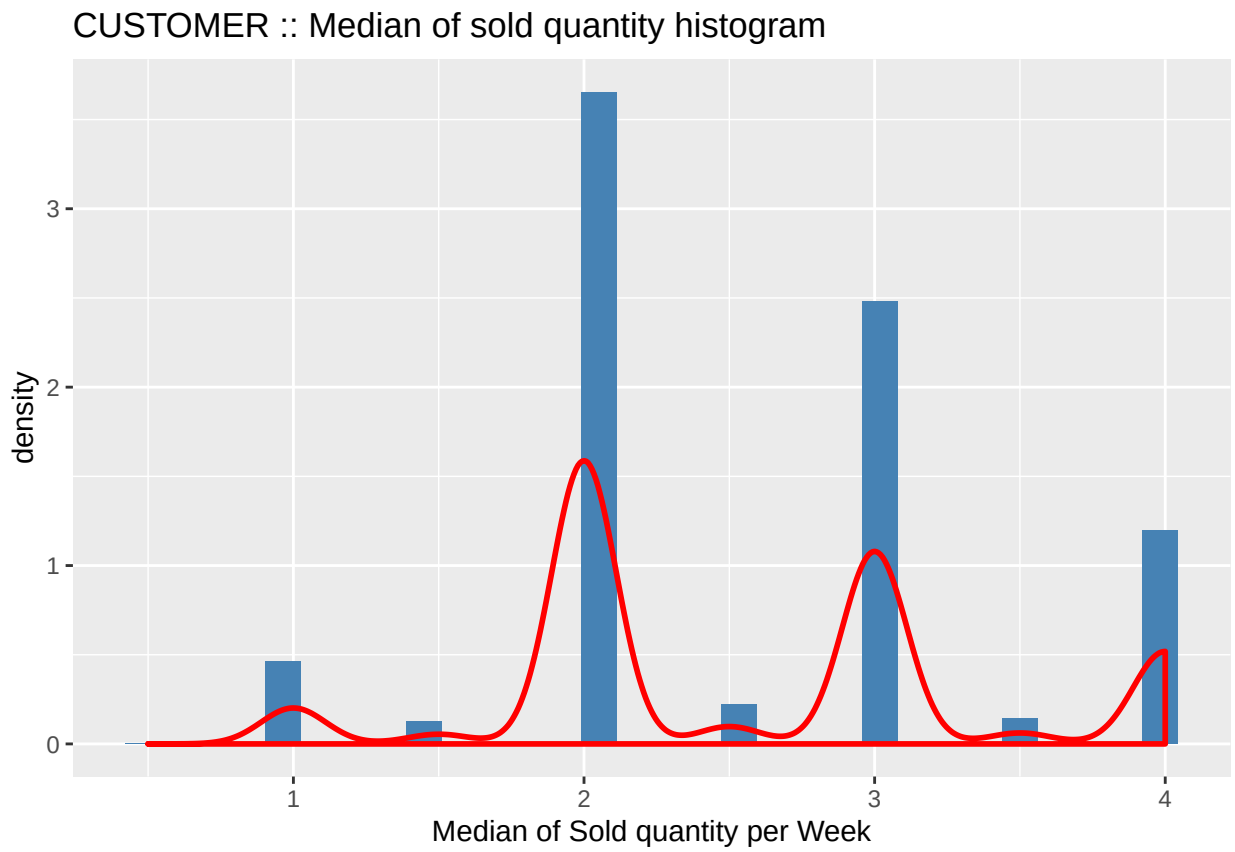



```
# What is the ITEMS median of each sale?
summary(df[,item_median])
```

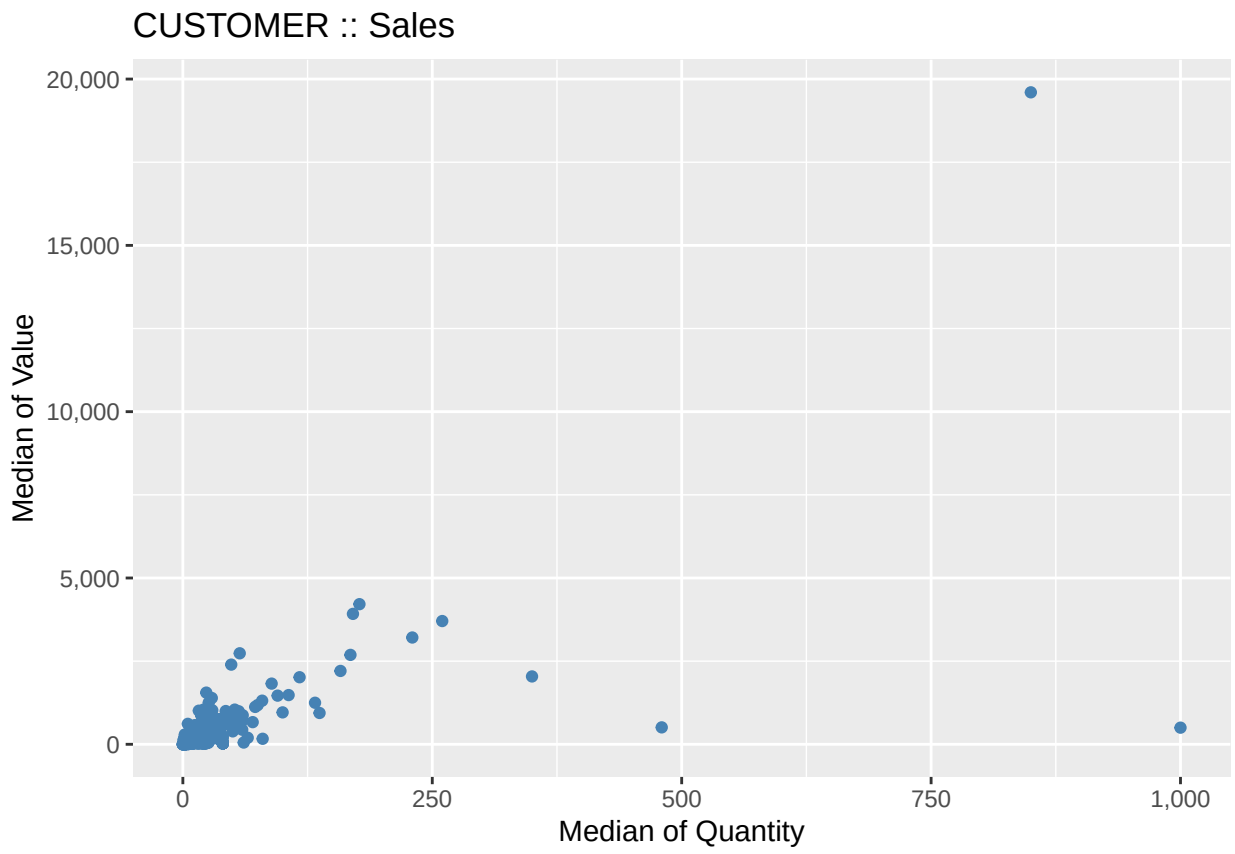
```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##    0.000    2.000    3.000    4.234    4.000 4983.000
```

```
ggplot(df_sample[item_median > 0 & item_median <= quantile(x = df[,item_median], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = item_median, y = ..density..), fill = "steelblue") +
  geom_density(aes(x = item_median), color = "red", size = 1) +
  ggtitle("CUSTOMER :: Median of sold quantity histogram") +
  xlab("Median of Sold quantity per Week")
```

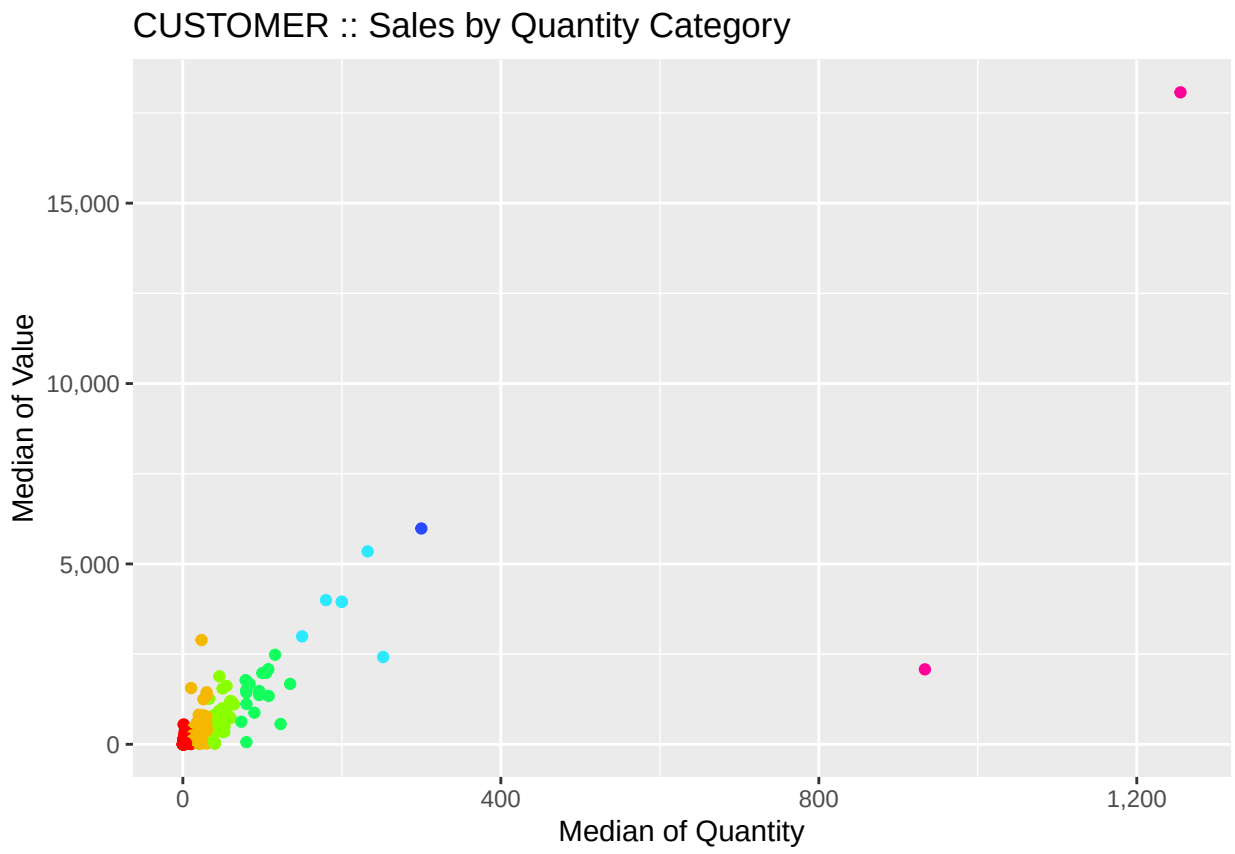
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



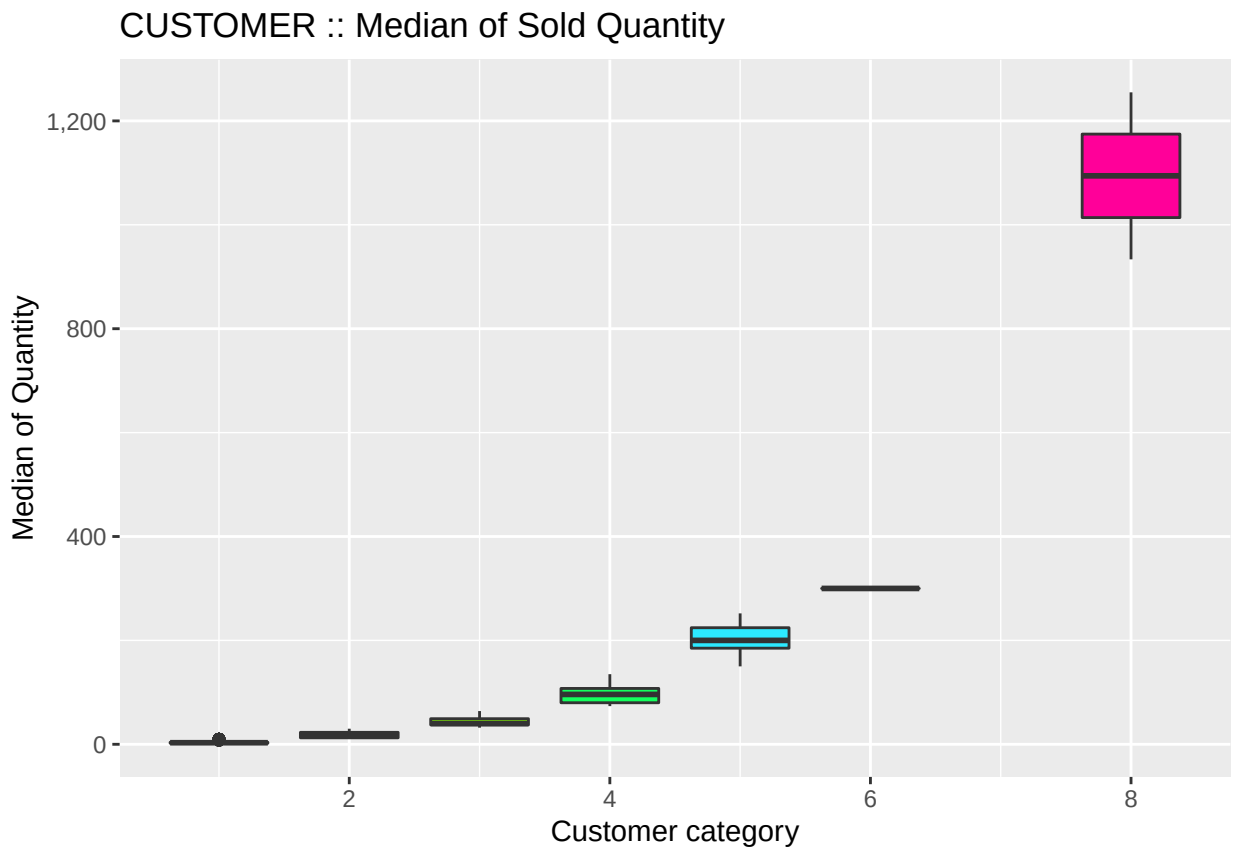

```
ggplot(df_sample) +
  geom_point(aes(x = item_median, y = value_median), color = "steelblue") +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("CUSTOMER :: Sales") +
  ylab("Median of Value") +
  xlab("Median of Quantity")
```



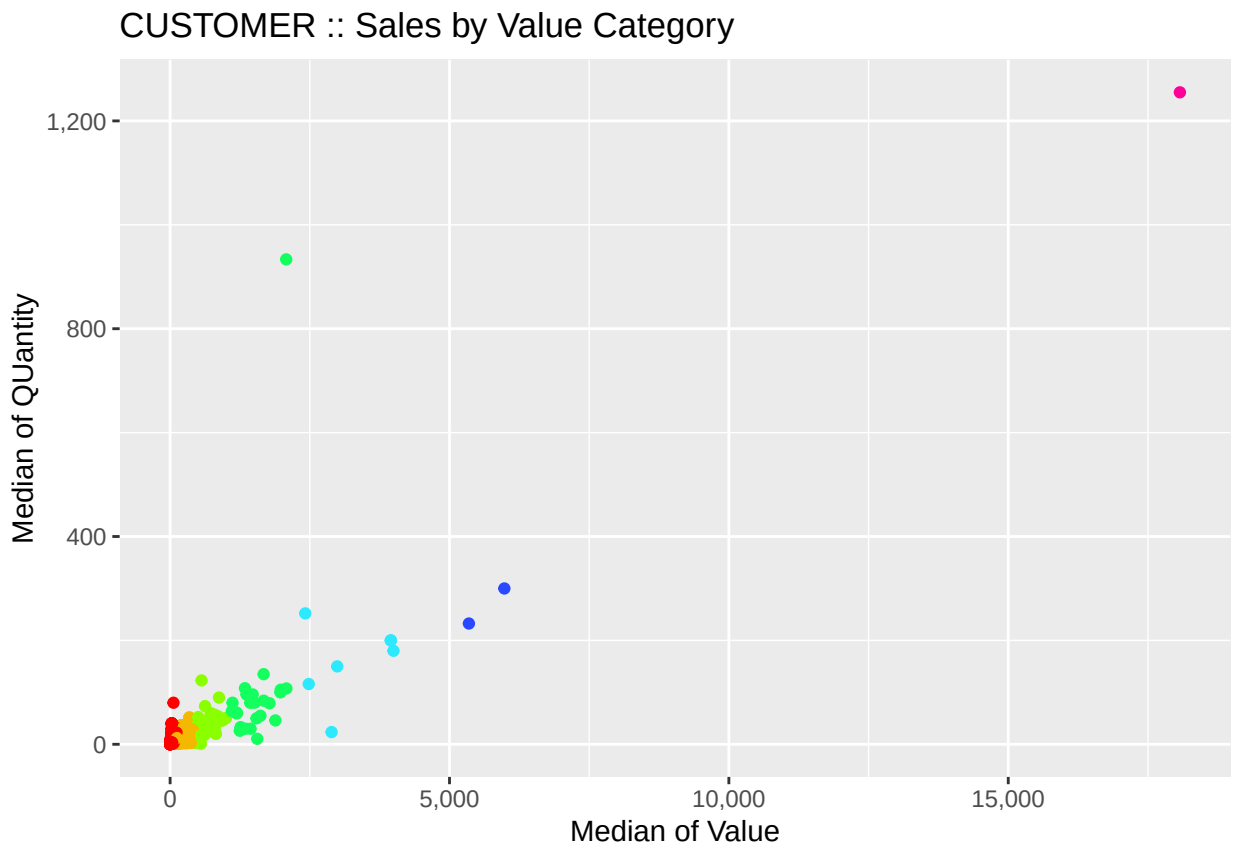

```
# Customer category
cluster <- kmeans(df[,item_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df[,item_median], centers)
df$customer_item_median_category <- cluster$cluster

# Customer category
cluster <- kmeans(df[,value_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df[,value_median], centers)
df$customer_value_median_category <- cluster$cluster

remove(cluster)
remove(centers)

df_sample <- df[sample(nrow(df), 10000)]

ggplot(df_sample) + #[item_median >= (m - 3 * sd) & item_median <= (m + 3 * sd)]) +
  geom_point(aes(x = item_median, y = value_median, color = customer_item_median_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("CUSTOMER :: Sales by Quantity Category") +
  ylab("Median of Value") +
  xlab("Median of Quantity") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```



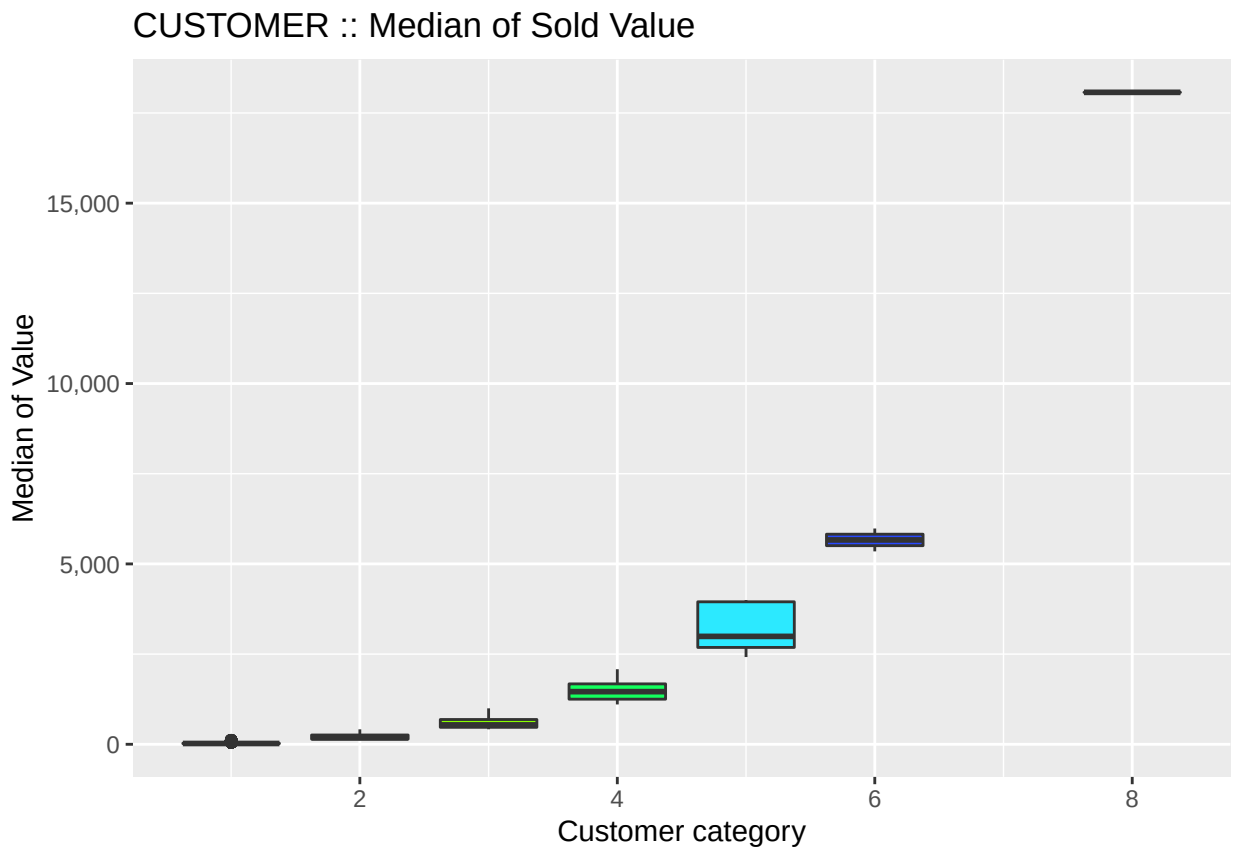

```
ggplot(df_sample) +
  geom_boxplot(aes(x = customer_item_median_category, y = item_median, group = customer_item_median_cat
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("CUSTOMER :: Median of Sold Quantity") +
  ylab("Median of Quantity") +
  xlab("Customer category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df_sample) +
  geom_point(aes(x = value_median, y = item_median, color = customer_value_median_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("CUSTOMER :: Sales by Value Category") +
  ylab("Median of QUantity") +
  xlab("Median of Value") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df_sample) +
  geom_boxplot(aes(x = customer_value_median_category, y = value_median, group = customer_value_median_
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("CUSTOMER :: Median of Sold Value") +
  ylab("Median of Value") +
  xlab("Customer category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
remove(df_sample)

fwrite(x = df, file = v_c_output_customer_median)

remove(df)
gc()
```

```
##             used   (Mb) gc trigger   (Mb)  max used   (Mb)
## Ncells   1737797   92.9    2631004  140.6   2631004  140.6
## Vcells 536346378 4092.0  847872332 6468.8 847872332 6468.8
```

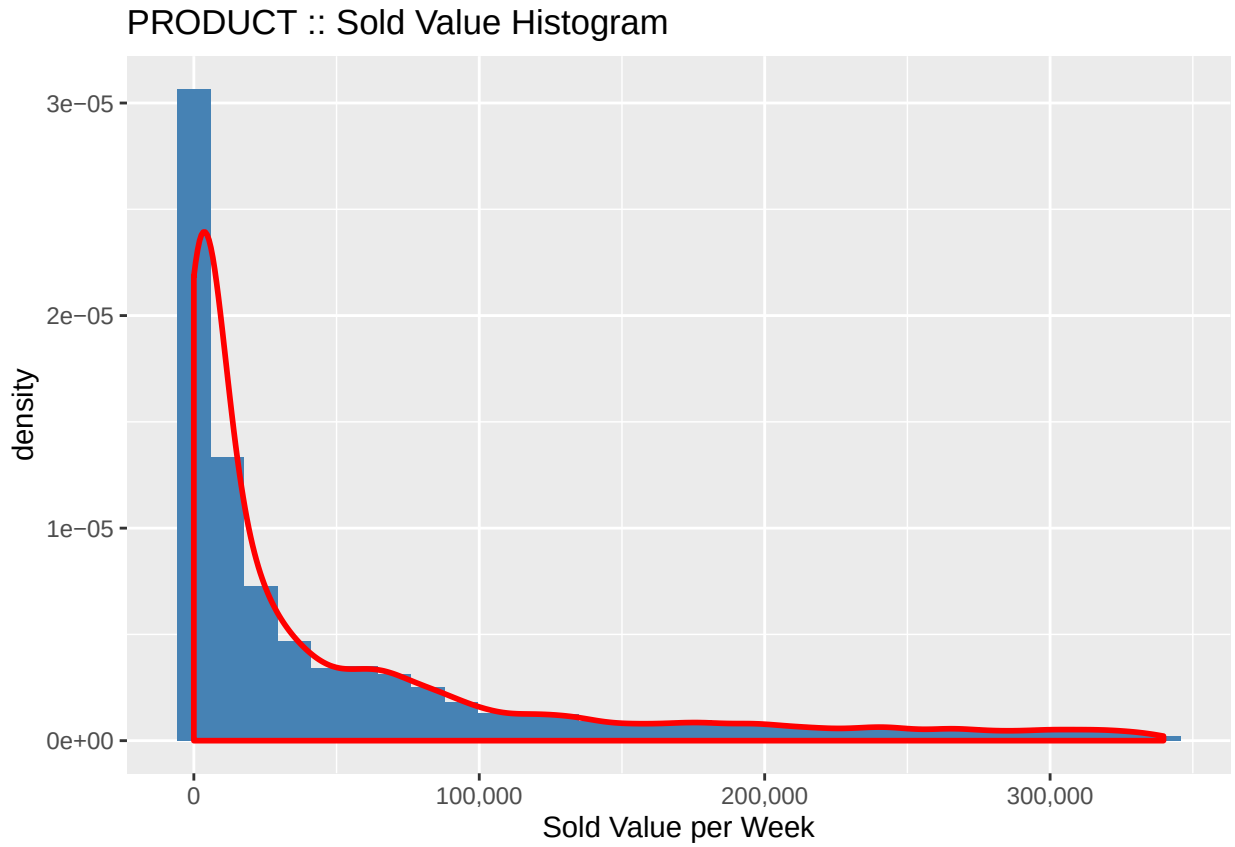
```
#
# PRODUCT ANALISYS
#
# How many products BIMBO has?
paste("The BIMBO group has", length(unique(train[,Producto_ID])), "products")
```

```
## [1] "The BIMBO group has 1799 products"
```

```
df <- train[, list(item_sum = sum(Venta_uni_hoy),
                   value_sum = sum(Venta_hoy)),
            by = list(Producto_ID, Semana)]

# How much was sold from each product?
summary(df[,value_sum])
```

```
##      Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##         0     3167    30757   466141   215897 34290415
```

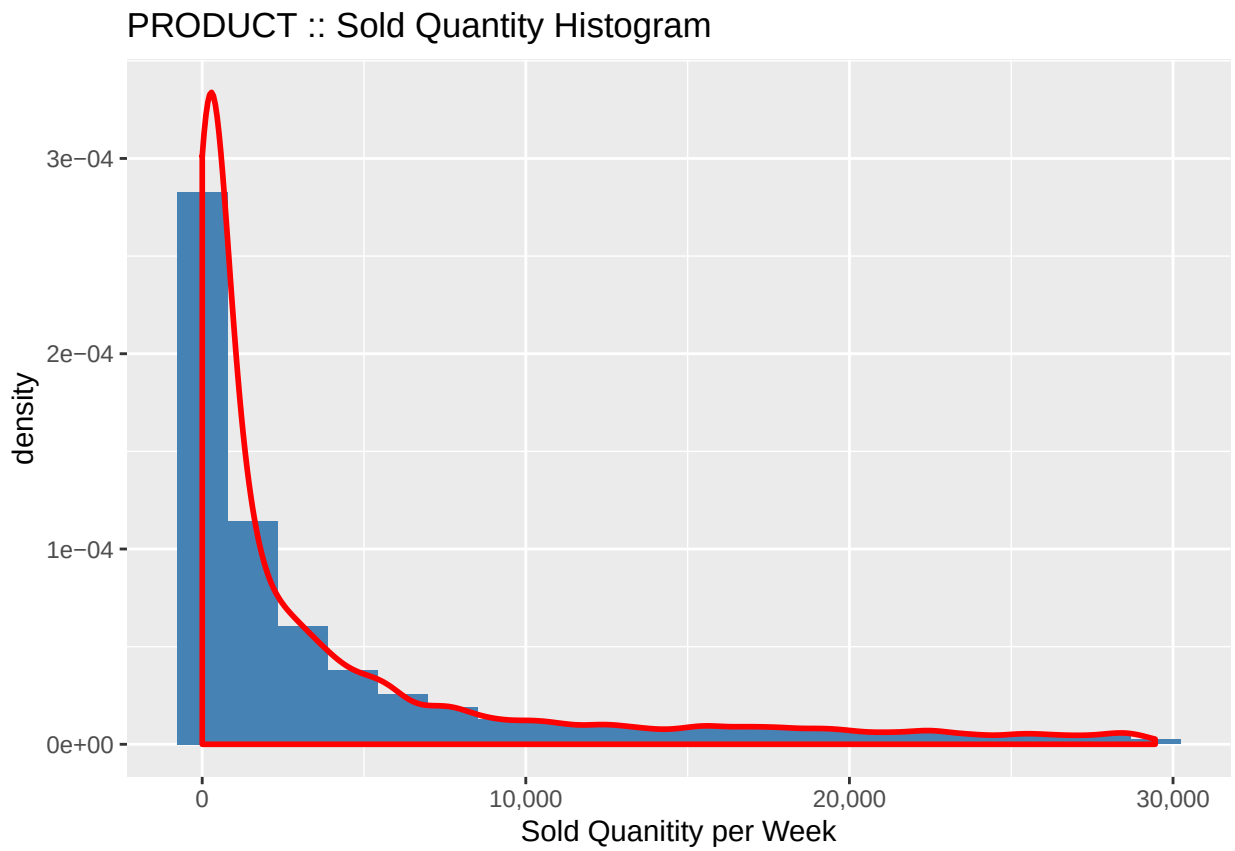
```
ggplot(df[value_sum > 0 & value_sum <= quantile(x = df[,value_sum], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = value_sum, y = ..density..), bins = 30, fill = "steelblue") +
  geom_density(aes(x = value_sum), color = "red", size = 1) +
  ggtitle("PRODUCT :: Sold Value Histogram") +
  xlab("Sold Value per Week")
```



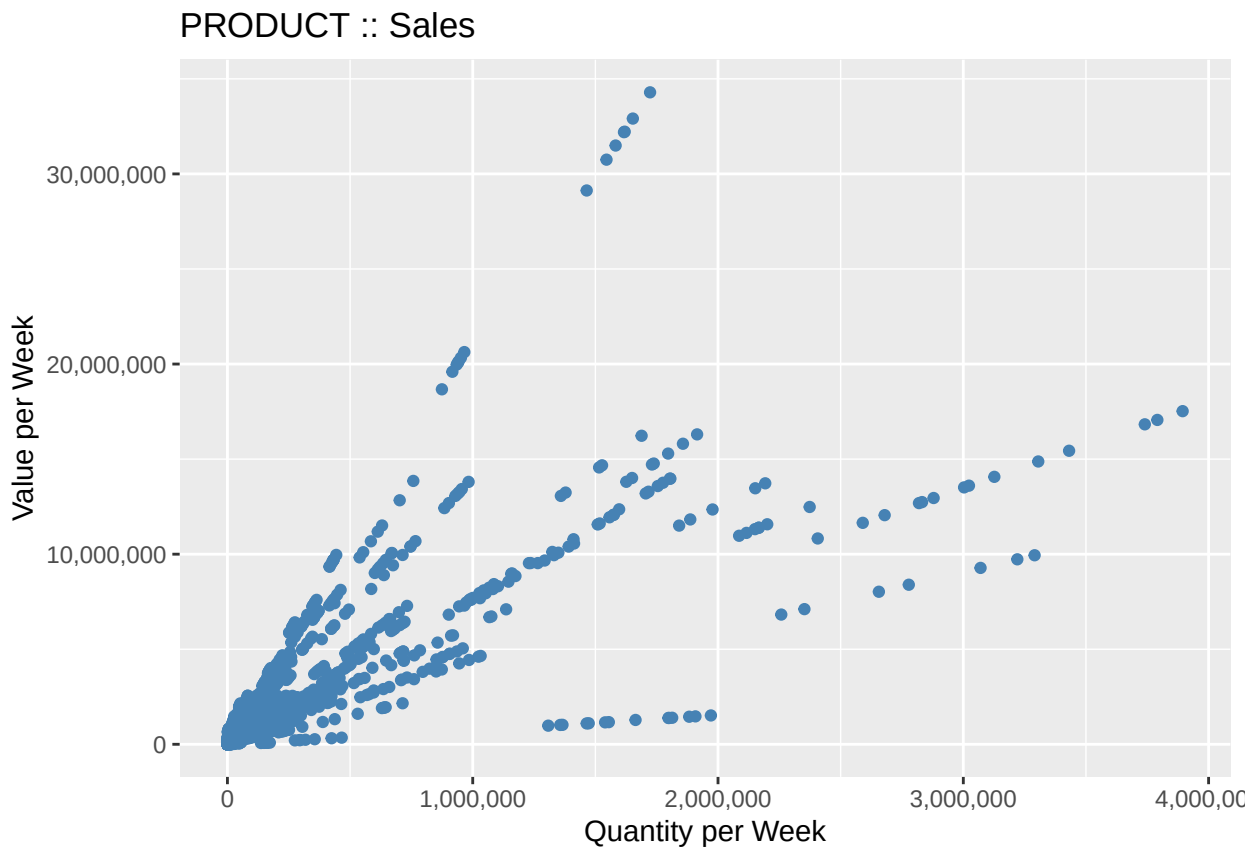

```
# How many was sold for each product?
summary(df[,item_sum])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0     255    2241   49713   18448 3894123
```

```
ggplot(df[item_sum > 0 & item_sum <= quantile(x = df[,item_sum], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = item_sum, y = ..density..), bins = 20, fill = "steelblue") +
  geom_density(aes(x = item_sum), color = "red", size = 1) +
  ggtitle("PRODUCT :: Sold Quantity Histogram") +
  xlab("Sold Quanitity per Week")
```

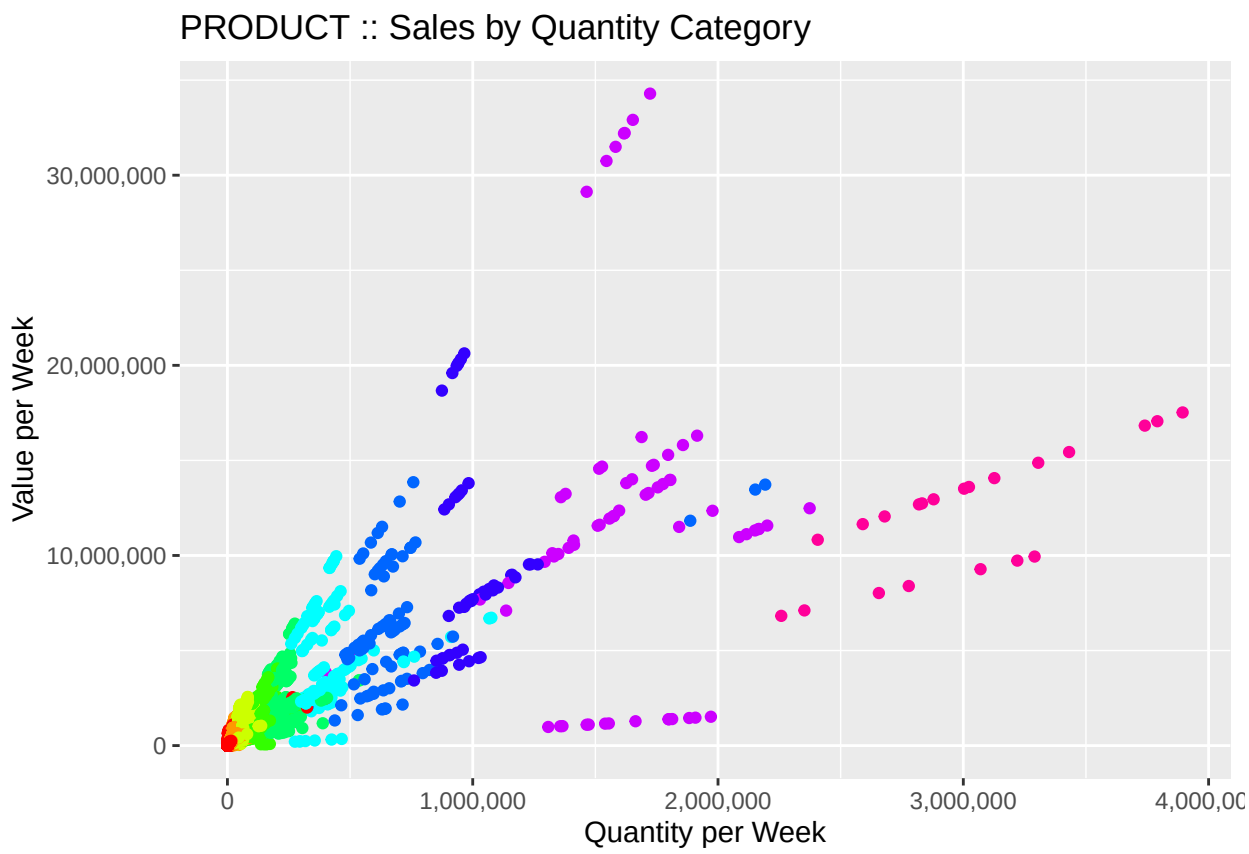



```
ggplot(df) +
  geom_point(aes(x = item_sum, y = value_sum), color = "steelblue") +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("PRODUCT :: Sales") +
  ylab("Value per Week") +
  xlab("Quantity per Week")
```

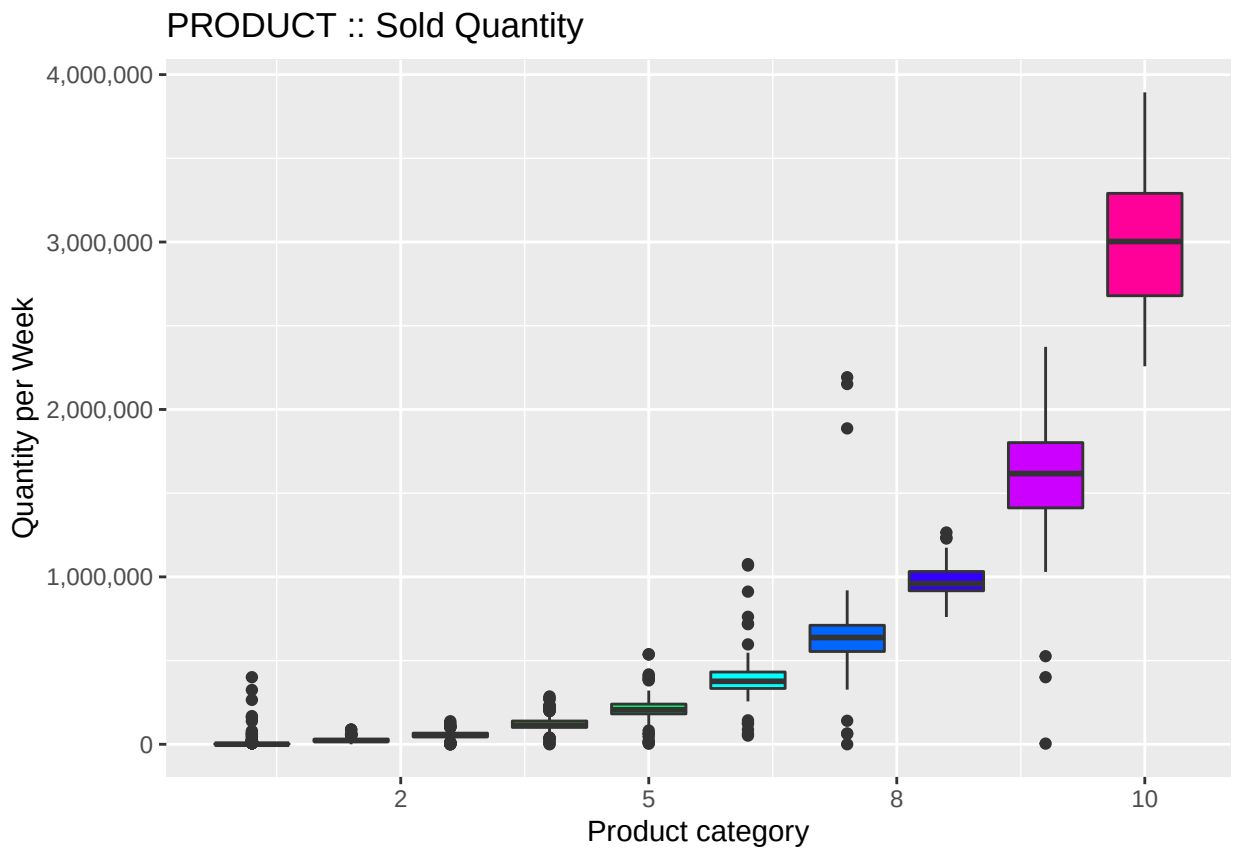



```
# Product category
df1 <- df[, list(item_median = median(item_sum),
                 value_median = median(value_sum)),
          by = Producto_ID]

cluster <- kmeans(df1[,item_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df1[,item_median], centers)
df1$product_item_category <- cluster$cluster

cluster <- kmeans(df1[,value_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df1[,value_median], centers)
df1$product_value_category <- cluster$cluster

remove(cluster)
remove(centers)

df <- merge(x = df, y = df1[,list(Producto_ID, product_item_category, product_value_category)])

remove(df1)

ggplot(df) +
  geom_point(aes(x = item_sum, y = value_sum, color = product_item_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
```

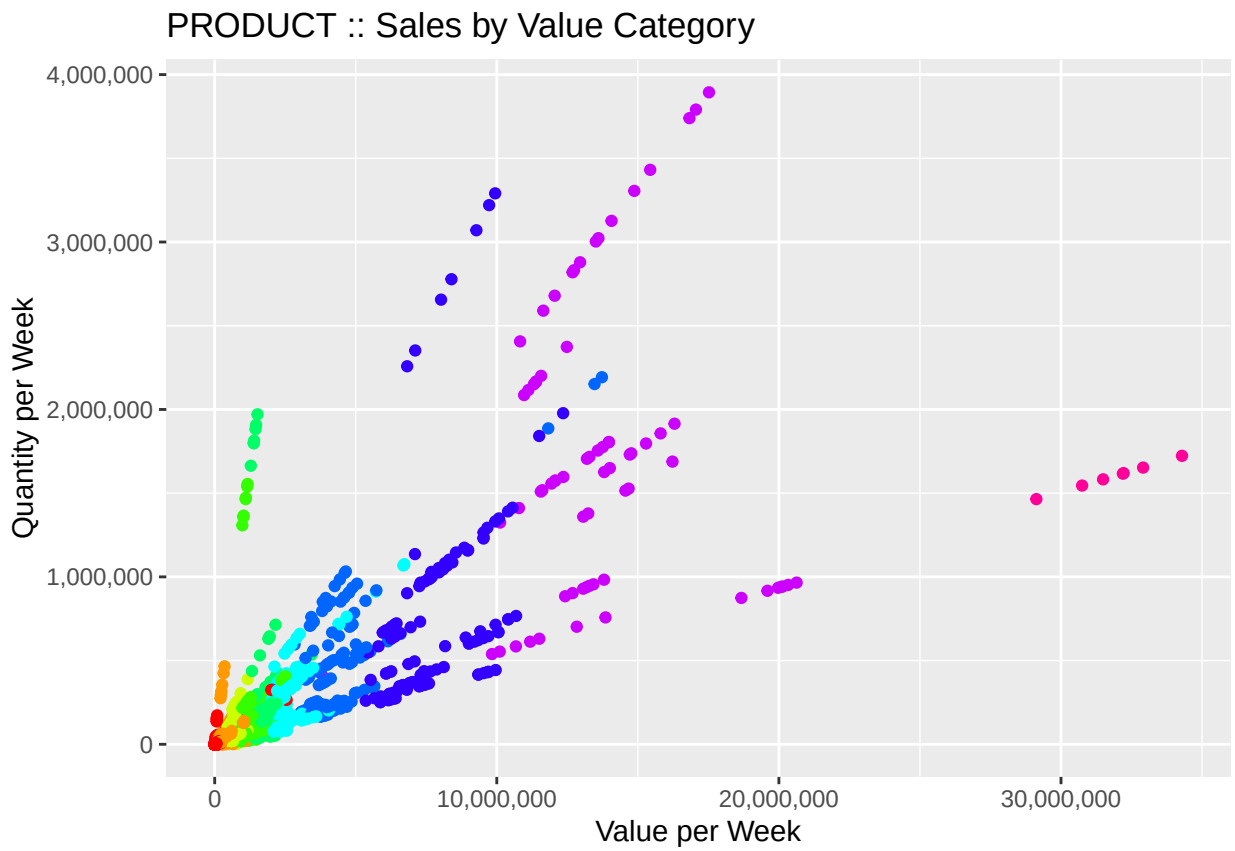
```
  ggtitle("PRODUCT :: Sales by Quantity Category") +
  ylab("Value per Week") +
  xlab("Quantity per Week") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```



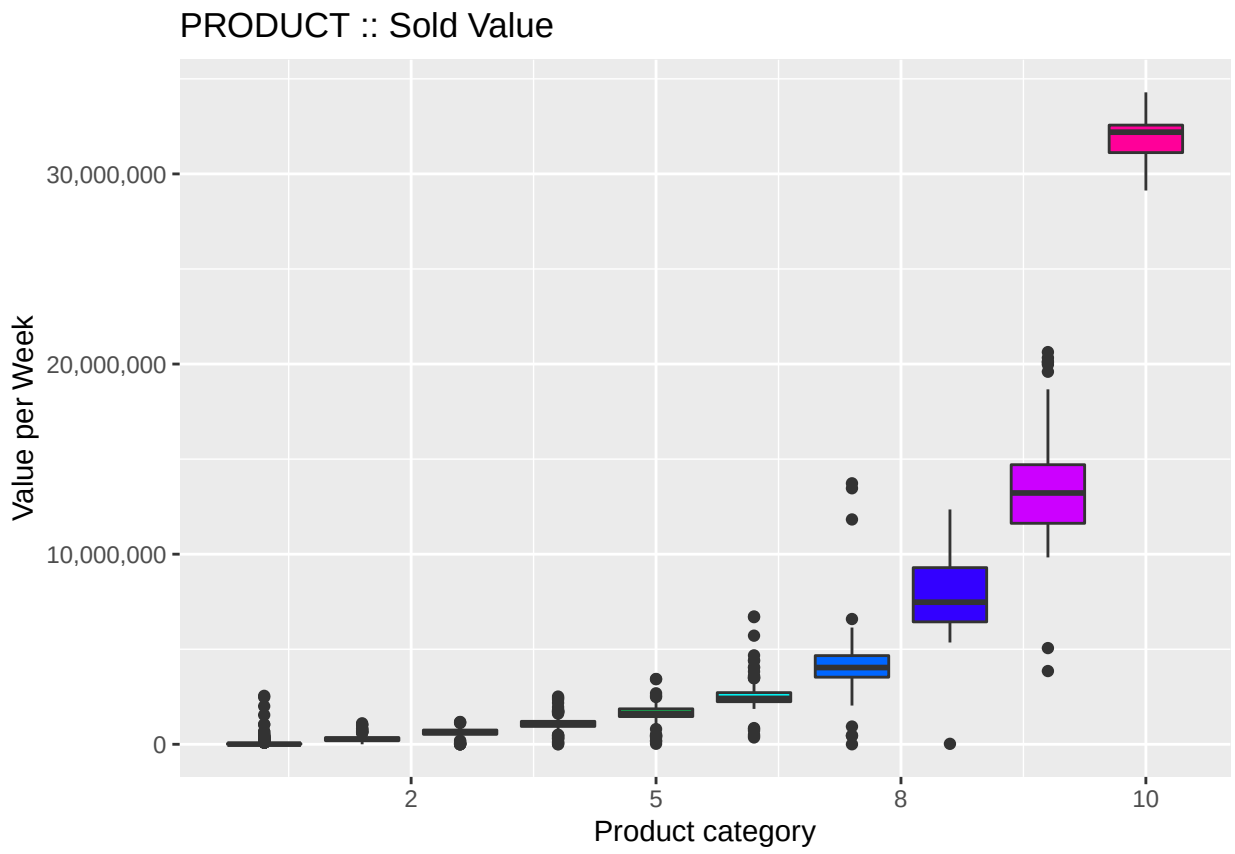

```
ggplot(df) +
  geom_boxplot(aes(x = product_item_category, y = item_sum, group = product_item_category, fill = produ
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("PRODUCT :: Sold Quantity") +
  ylab("Quantity per Week") +
  xlab("Product category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df) +
  geom_point(aes(x = value_sum, y = item_sum, color = product_value_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("PRODUCT :: Sales by Value Category") +
  ylab("Quantity per Week") +
  xlab("Value per Week") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df) +
  geom_boxplot(aes(x = product_value_category, y = value_sum, group = product_value_category, fill = pr
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("PRODUCT :: Sold Value") +
  ylab("Value per Week") +
  xlab("Product category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
# Saving the product categories into train dataset
# train <- merge(x = train, y = df[,c("Producto_ID","product_value_category", "product_item_category")]
fwrite(x = df, file = v_c_output_product_sum)

remove(df)
gc()
```

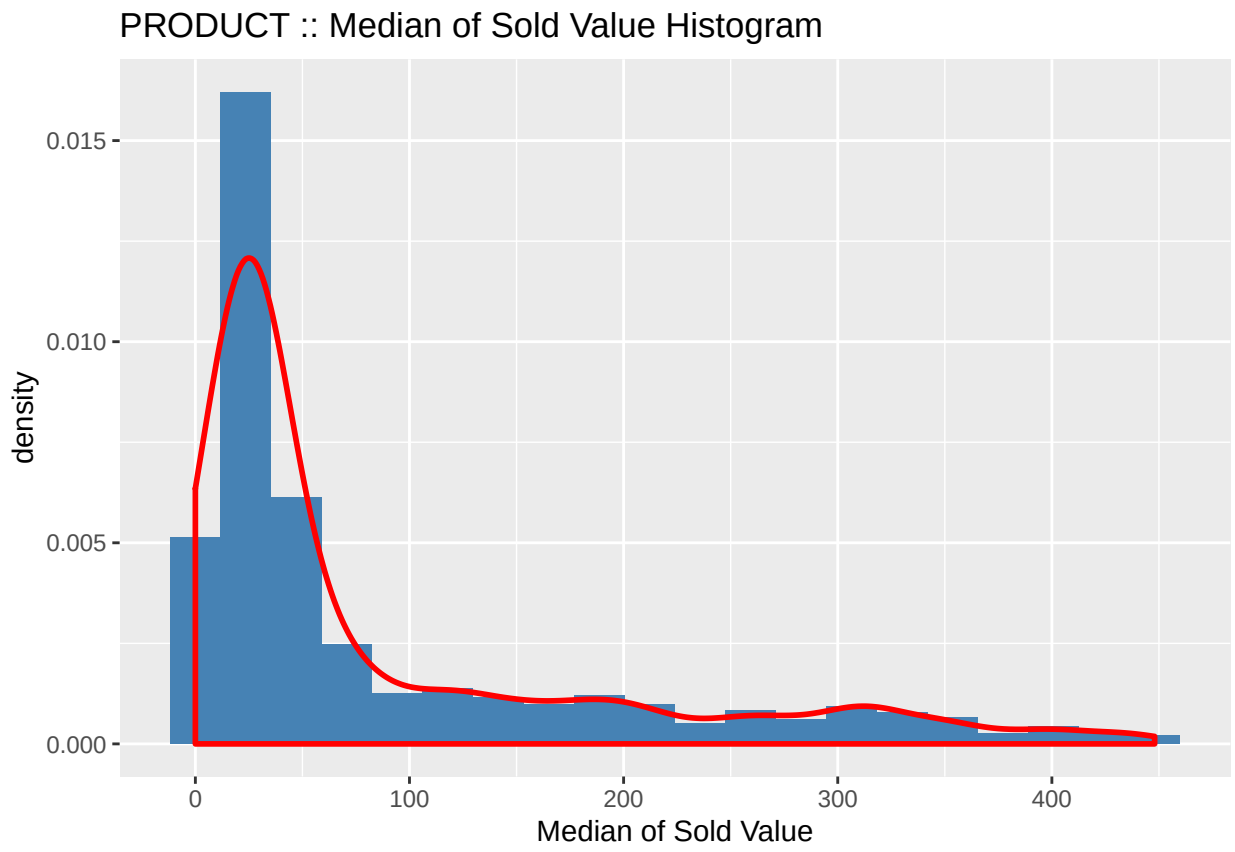
```
##             used   (Mb) gc trigger   (Mb)  max used   (Mb)
## Ncells   1877251  100.3    3273447  174.9   2631004  140.6
## Vcells 537101928 4097.8  847872332 6468.8 847872332 6468.8
```

```
# Unit sales median
df <- train[, list(item_median = median(Venta_uni_hoy),
                   value_median = median(Venta_hoy)),
            by = Producto_ID]

# What is the PESOS median of each product?
summary(df[,value_median])
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##      0.00     23.34     52.80    820.40    316.10 141657.60
```

```
ggplot(df[value_median >= 0 & value_median <= quantile(x = df[,value_median], probs = 0.8)]) +
  scale_y_continuous() +
```

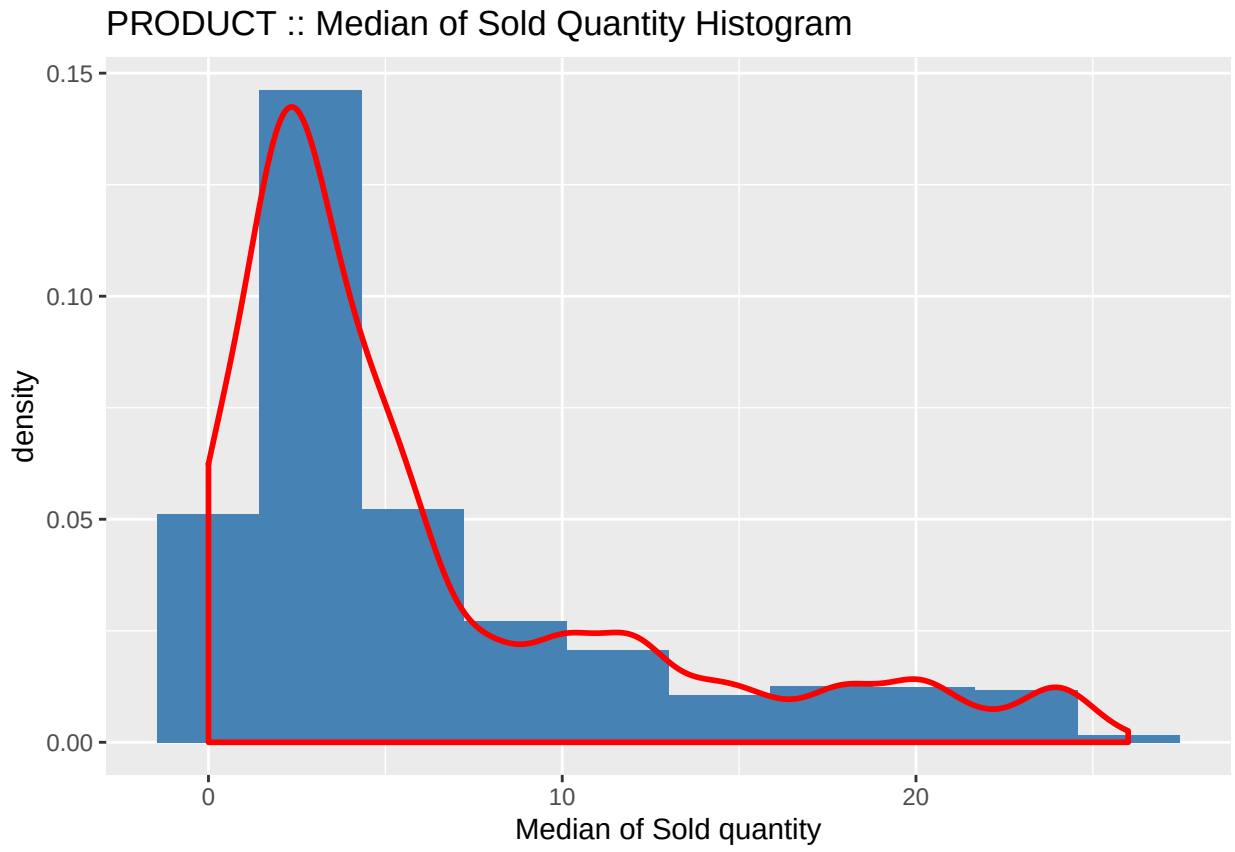
```
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = value_median, y = ..density..), bins = 20, fill = "steelblue") +
  geom_density(aes(x = value_median), color = "red", size = 1) +
  ggtitle("PRODUCT :: Median of Sold Value Histogram") +
  xlab("Median of Sold Value")
```



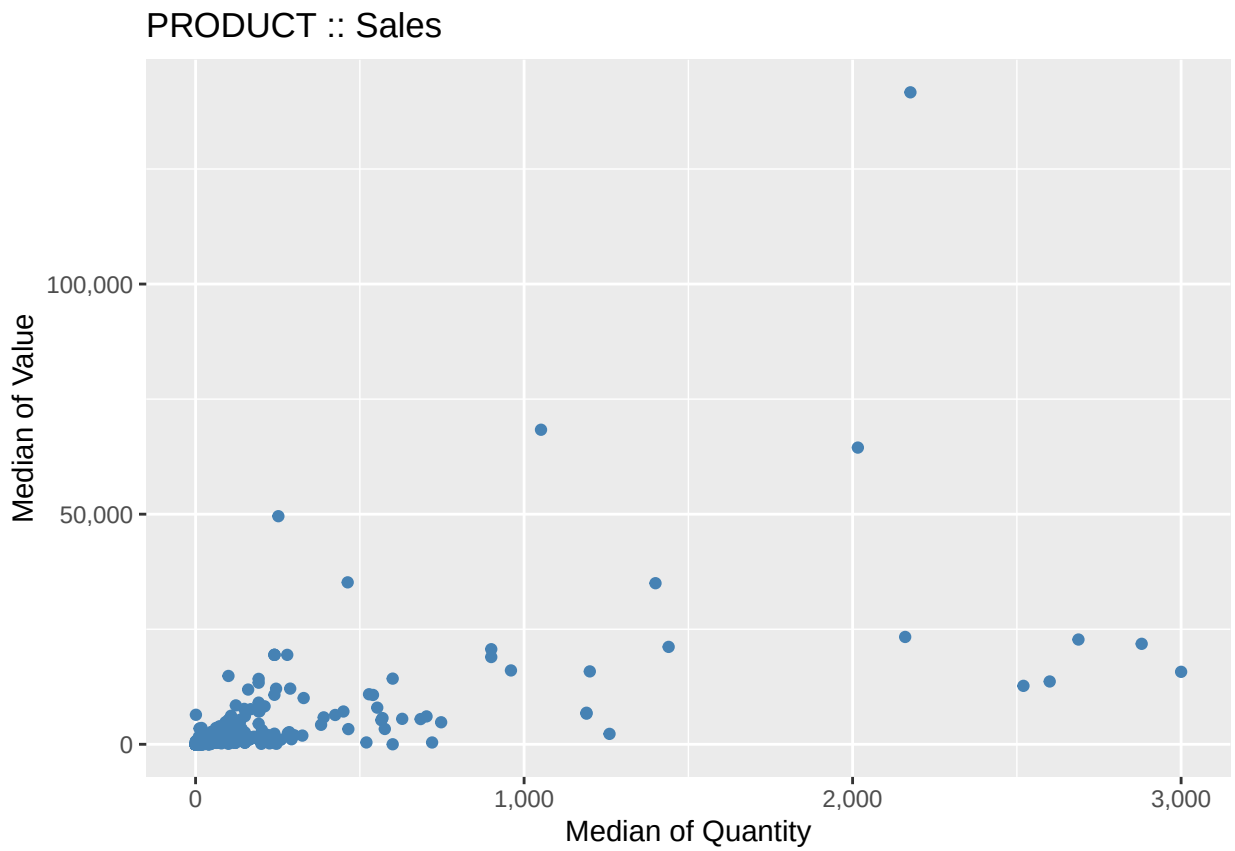

```
# What is the ITEMS median of each product?
summary(df[,item_median])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     2.0     5.0    44.8    20.0  3000.0
```

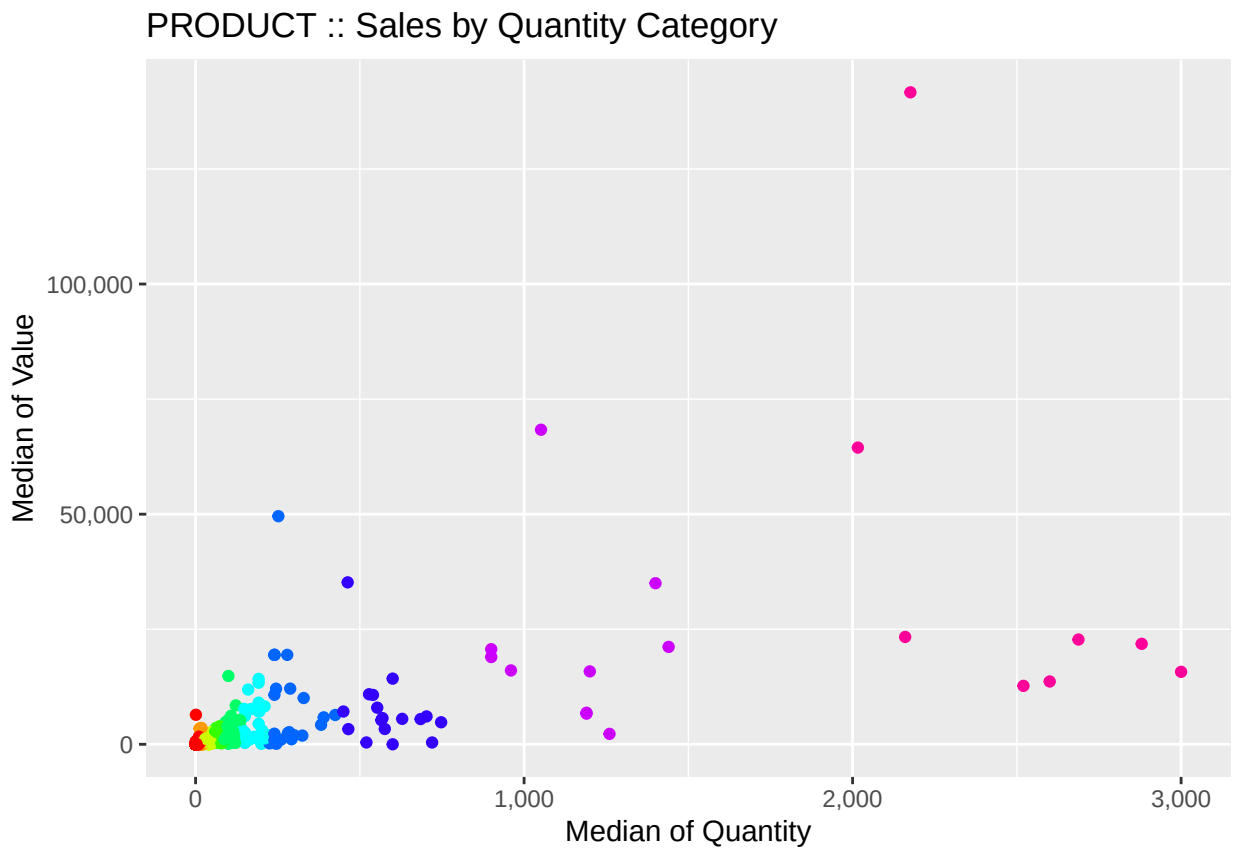
```
ggplot(df[item_median >= 0 & item_median <= quantile(x = df[,item_median], probs = 0.8)]) +
  scale_y_continuous() +
  scale_x_continuous(labels = comma) +
  geom_histogram(aes(x = item_median, y = ..density..), bins = 10, fill = "steelblue") +
  geom_density(aes(x = item_median), color = "red", size = 1) +
  ggtitle("PRODUCT :: Median of Sold Quantity Histogram") +
  xlab("Median of Sold quantity")
```



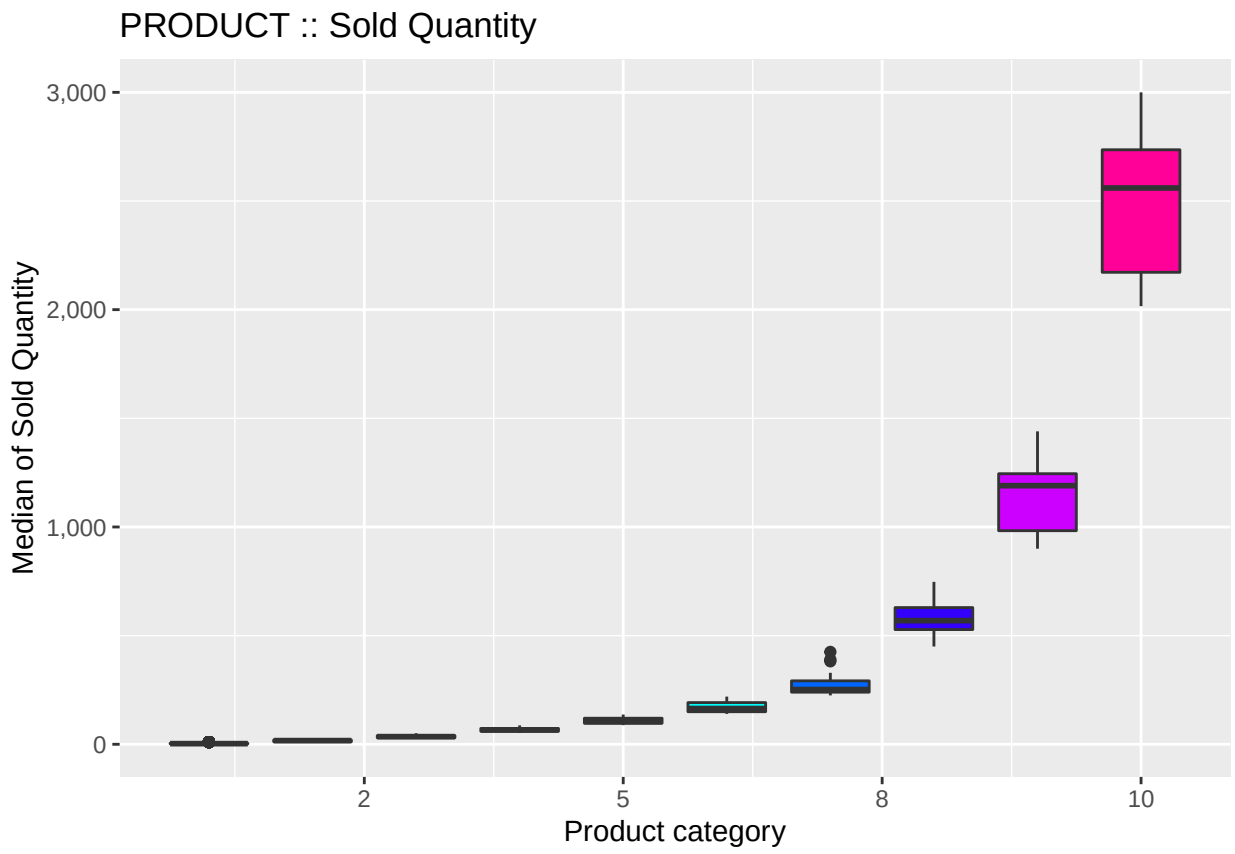

```
ggplot(df) +
  geom_point(aes(x = item_median, y = value_median), color = "steelblue") +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("PRODUCT :: Sales") +
  ylab("Median of Value") +
  xlab("Median of Quantity")
```



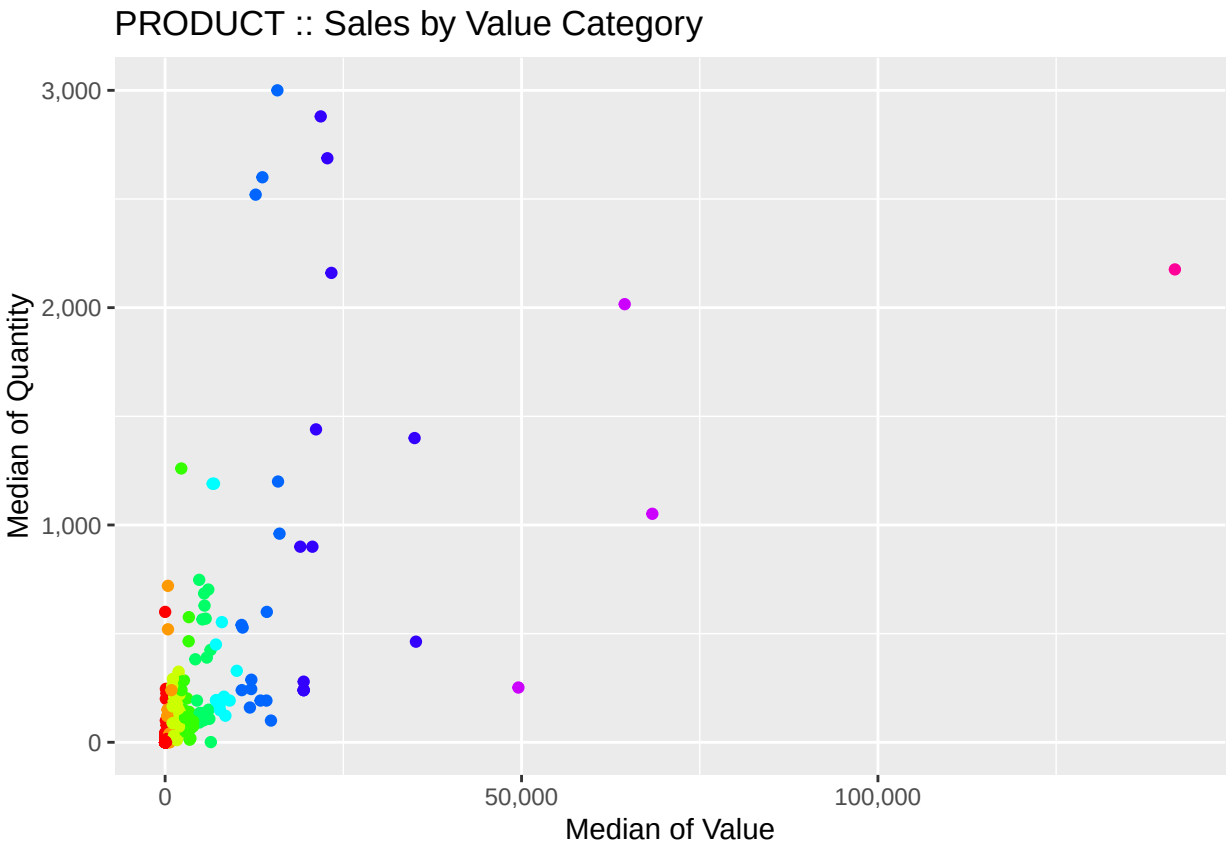

```
# Product category
cluster <- kmeans(df[,item_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df[,item_median], centers)
df$product_item_median_category <- cluster$cluster

cluster <- kmeans(df[,value_median], 10)
centers <- sort(cluster$centers)
cluster <- kmeans(df[,value_median], centers)
df$product_value_median_category <- cluster$cluster

remove(cluster)
remove(centers)

ggplot(df) +
  geom_point(aes(x = item_median, y = value_median, color = product_item_median_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("PRODUCT :: Sales by Quantity Category") +
  ylab("Median of Value") +
  xlab("Median of Quantity") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```



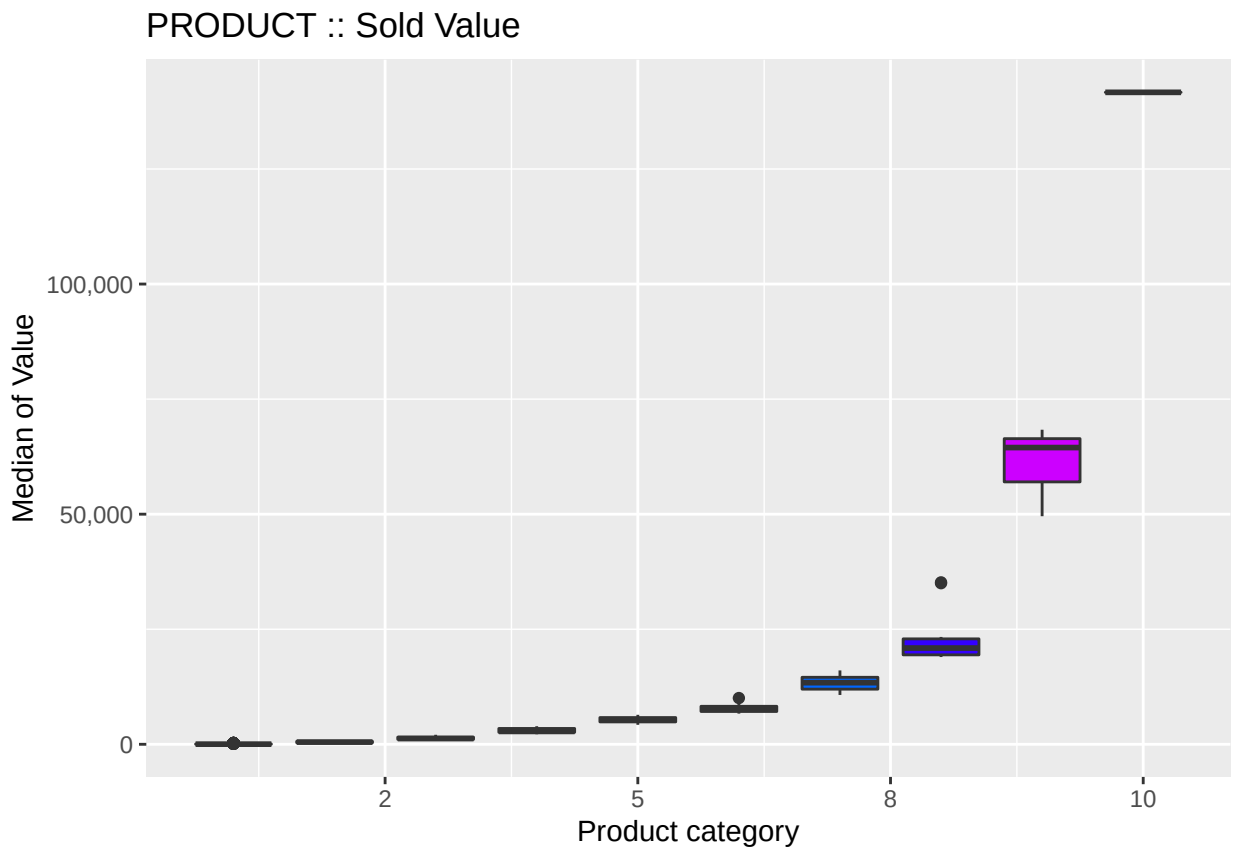

```
ggplot(df) +
  geom_boxplot(aes(x = product_item_median_category, y = item_median, group = product_item_median_categ
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("PRODUCT :: Sold Quantity") +
  ylab("Median of Sold Quantity") +
  xlab("Product category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df) +
  geom_point(aes(x = value_median, y = item_median, color = product_value_median_category)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("PRODUCT :: Sales by Value Category") +
  ylab("Median of Quantity") +
  xlab("Median of Value") +
  scale_color_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
ggplot(df) +
  geom_boxplot(aes(x = product_value_median_category, y = value_median, group = product_value_median_ca
  scale_y_continuous(labels = comma) +
  scale_x_continuous(labels = comma) +
  ggtitle("PRODUCT :: Sold Value") +
  ylab("Median of Value") +
  xlab("Product category") +
  scale_fill_gradientn(colours = rainbow(n = 10)) +
  theme(legend.position="none")
```




```
# train <- merge(x = train, y = df[,c("Producto_ID","product_value_median_category", "product_item_medi
fwrite(x = df, file = v_c_output_product_median)

remove(df)
remove(train)
remove(v_c_output_warehouse_median)
remove(v_c_output_warehouse_sum)
remove(v_c_output_customer_median)
remove(v_c_output_customer_sum)
remove(v_c_output_route_median)
remove(v_c_output_route_sum)
remove(v_c_output_product_median)
remove(v_c_output_product_sum)
gc()
```

```
##            used  (Mb) gc trigger   (Mb)  max used   (Mb)
## Ncells  2015474 107.7    3273447  174.9   3273447  174.9
## Vcells 55216443 421.3  678297866 5175.1 847872332 6468.8
```