IEUK Finance, Professional Services & Consulting Project

Xunqi Guo

2025-07-16

Task 1: Market Dynamics (Porter's Five Forces)

Threats to New Entrants (moderate)

Low entry barriers: The health and wellness beverage market (especially vitamins-in-water) has relatively low capital and technical barriers to entry. Many startups can enter with minimal investment.

Online DTC channels reduce dependency on retail shelf space, enabling new brands to enter without distributors. However, if VitaSip has exclusive retailer or subscription agreements, it increases barriers for newcomers.

VitaSip's success likely depends on strong branding, customer trust, and retail distribution, which new entrants may struggle to replicate quickly.

Power of suppliers (moderate)

The ingredients (e.g., vitamins, minerals, flavored water) are generally commodities, and there are likely many suppliers available. If VitaSip uses specialised ingredients or packaging, supplier power could be slightly higher, especially if switching is difficult. Large volume players like VitaSip can negotiate better prices with economies of scales, reducing supplier leverage.

Power of buyers (high)

Customers have plenty of choices many similar products (vitamin water, electrolyte drinks, functional beverages) exist in both physical and online channels. Brand loyalty is weak in the wellness beverage market unless there's a strong value proposition. Customers can easily compare prices and switch based on promotions or taste, high elasticity of demand. Retailers (e.g., supermarkets, pharmacies) also have power, they decide shelf space and can negotiate terms or replace products.

Threat of Substitutes (High)

As mentioned in power of buyers, Substitutes include: Traditional supplements (vitamin pills, powders), healthy juices or teas, water, Electrolyte tablets or energy drinks. Many of these are cheaper or more convenient. Some substitutes offer stronger health claims or greater personalisation.

Industry rivalry (Very High)

The health beverage industry is highly fragmented and saturated. Competitors include: global giants (e.g., Vitaminwater by Coca-Cola, Lucozade), functional startups. VitaSip may need constant innovation and strong marketing to maintain share.

Competition is based on: price, shelf space, taste, benefits

Task 2: Data Exploration

```r
#Import the transaction data
transaction <- read_excel("/Users/danielguo/Desktop/IEUK/vitasip_raw_data.xlsx")

# Data types
transaction_clean <- transaction %>%
  mutate(
    `Date of Sale` = as.Date(`Date of Sale`),
    Product = as.character(Product),
    Channel = tolower(as.character(Channel)),
    `Customer Type` = as.character(`Customer Type`),
    Quantity = as.numeric(Quantity),
    `Total Sale` = as.numeric(`Total Sale`),
    `Total Direct Cost` = as.numeric(`Total Direct Cost`),
    Location = as.character(Location)
  )

# Handle missing values
transaction_clean <- transaction_clean %>%
  drop_na()

# Remove duplicates
transaction_clean <- transaction_clean %>%
  distinct()

# Adding additional columns
transaction_clean <- transaction_clean %>%
  mutate(
    profit = `Total Sale` - `Total Direct Cost`,
    month = floor_date(`Date of Sale`, "month"),
    year = year(`Date of Sale`)
  )

# Preview clean dataset
head(transaction_clean)
```

```
## # A tibble: 6 x 11
##   `Date of Sale` Product   Channel         `Customer Type` Quantity `Total Sale`
##   <date>         <chr>     <chr>           <chr>              <dbl>        <dbl>
## 1 2024-01-01     FocusFizz boutique fitne~ New                    1            4
## 2 2024-01-01     FocusFizz boutique fitne~ New                    3           12
## 3 2024-01-01     FocusFizz boutique fitne~ New                    2            8
## 4 2024-01-02     FocusFizz boutique fitne~ New                    1            4
## 5 2024-01-02     FocusFizz boutique fitne~ New                    2            8
## 6 2024-01-02     FocusFizz boutique fitne~ New                    5           20
## # i 5 more variables: `Total Direct Cost` <dbl>, Location <chr>, profit <dbl>,
## #   month <date>, year <dbl>
```

```r
# Calculate total sales, costs, profit and gross profit margin per product, channel and customer
product_summary <- transaction_clean %>%
  group_by(Product) %>%
  summarise(
    total_revenue = sum(`Total Sale`, na.rm = TRUE),
```

```
    total_cost = sum(`Total Direct Cost`, na.rm = TRUE),
    total_profit = total_revenue - total_cost,
    gross_profit_margin = round((total_profit / total_revenue) * 100, 2)
  )

kable(product_summary, caption = "Sales Summary by Product")
```

Table 1: Sales Summary by Product

Product	total_revenue	total_cost	total_profit	gross_profit_margin
CalmKombu	9890.65	5162.45	4728.20	47.80
FocusFizz	11369.90	5611.12	5758.78	50.65
GlowTonic	10982.30	6278.61	4703.69	42.83
RestRefresher	5881.90	2382.40	3499.50	59.50

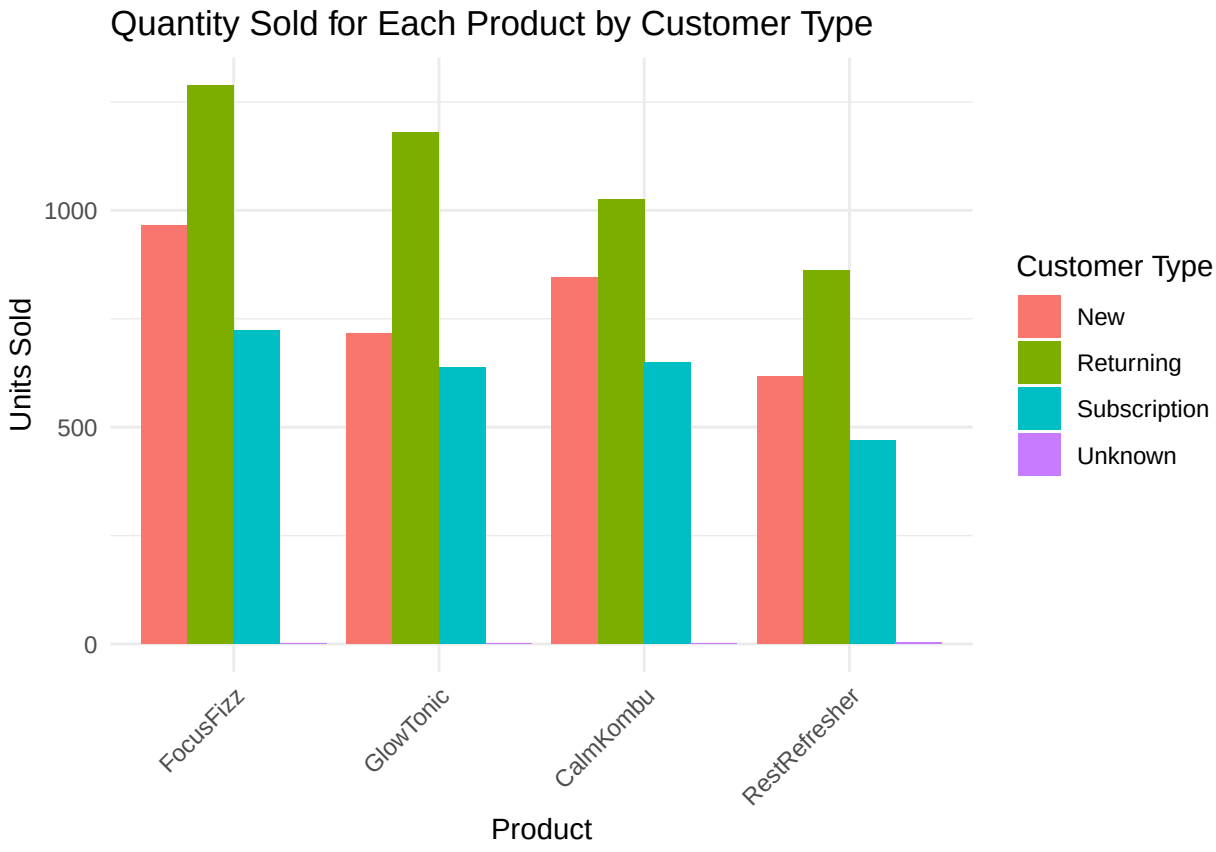
```
# Product analysis - number of sales, quantity sold and number of sales and quantity sold by customer ty
product_analysis <- transaction_clean %>%
  group_by(Product) %>%
  summarise(
    sales = n(),
    quantity_sold = sum(Quantity, na.rm = TRUE),

    no_sales_new = sum(`Customer Type` == "New", na.rm = TRUE),
    qty_sold_new = sum(Quantity[`Customer Type` == "New"], na.rm = TRUE),

    no_sales_returning = sum(`Customer Type` == "Returning", na.rm = TRUE),
    qty_sold_returning = sum(Quantity[`Customer Type` == "Returning"], na.rm = TRUE),

    no_sales_subscription = sum(`Customer Type` == "Subscription", na.rm = TRUE),
    qty_sold_subscription = sum(Quantity[`Customer Type` == "Subscription"], na.rm = TRUE),

    no_sales_unknown = sum(`Customer Type` == "Unknown", na.rm = TRUE),
    qty_sold_unknown = sum(Quantity[`Customer Type` == "Unknown"], na.rm = TRUE)
  ) %>%
  arrange(desc(sales))

kable(head(product_analysis, 10), caption = "Product-Level analysis by sales, quantity sold and Customer
```

Table 2: Product-Level analysis by sales, quantity sold and Customer Type

Product	sales	quantity_sold	no_sales_new	qty_sold_new	no_sales_returning	qty_sold_returning	no_sales_subscription	qty_sold_subscription	no_sales_unknown	qty_sold_unknown
FocusFizz	2984	2979	320	966	429	1288	234	723	1	2
CalmKombu	2440	2522	284	845	343	1025	212	650	1	2
GlowTonic	2329	2532	246	715	383	1179	200	638	0	0
RestRefresher	1660	1951	210	616	292	861	157	470	1	4

```
product_qty_long <- product_analysis %>%
  select(Product,
         qty_sold_new,
         qty_sold_returning,
```

```
        qty_sold_subscription,
        qty_sold_unknown) %>%
  pivot_longer(
    cols = starts_with("qty_sold"),
    names_to = "customer_type",
    values_to = "quantity_sold"
  ) %>%
  mutate(
    customer_type = case_when(
      customer_type == "qty_sold_new" ~ "New",
      customer_type == "qty_sold_returning" ~ "Returning",
      customer_type == "qty_sold_subscription" ~ "Subscription",
      customer_type == "qty_sold_unknown" ~ "Unknown"
    )
  )
ggplot(product_qty_long, aes(x = reorder(Product, -quantity_sold), y = quantity_sold, fill = customer_ty
  geom_col(position = "dodge") +
  labs(
    title = "Quantity Sold for Each Product by Customer Type",
    x = "Product",
    y = "Units Sold",
    fill = "Customer Type"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

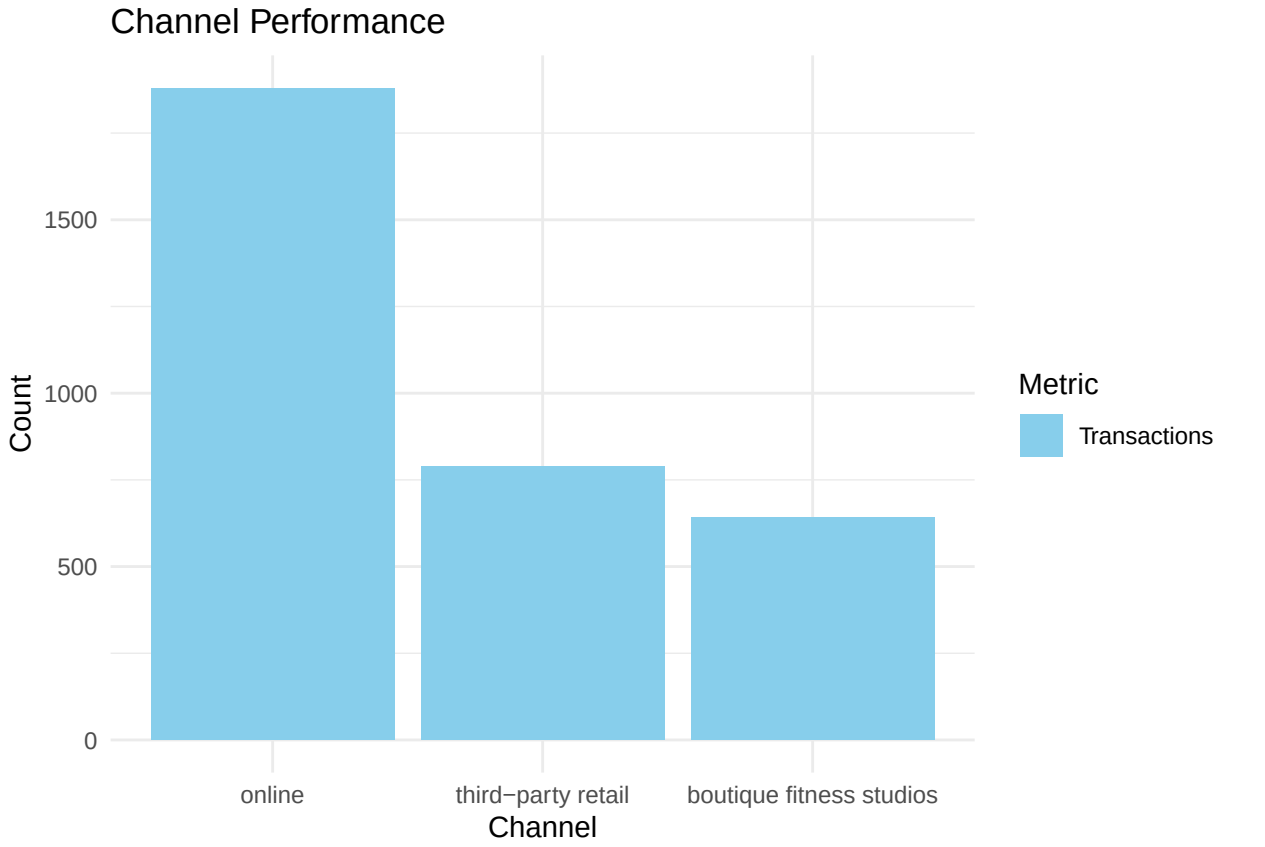


```r
channel_summary <- transaction_clean %>%
  group_by(Channel) %>%
  summarise(
    sales = n(),
    quantity_sold = sum(Quantity, na.rm = TRUE),
    total_revenue = sum(`Total Sale`, na.rm = TRUE),
    total_cost = sum(`Total Direct Cost`, na.rm = TRUE),
    total_profit = total_revenue - total_cost,
    gross_profit_margin = round((total_profit / total_revenue) * 100, 2)
  )

kable(channel_summary, caption = "Sales Summary by Channel")
```

Table 3: Sales Summary by Channel

Channel	sales	quantity_sold	total_revenue	total_cost	total_profit	gross_profit_margin
boutique fitness studios	643	1903	7318.50	3824.48	3494.02	47.74
online	1880	5784	21076.75	10489.00	10587.75	50.23
third-party retail	790	2297	9729.50	5121.10	4608.40	47.37

```r
ggplot(channel_summary, aes(x = reorder(Channel, -sales))) +
  geom_col(aes(y = sales, fill = "Transactions"), position = "dodge") +
  labs(title = "Channel Performance", x = "Channel", y = "Count") +
  scale_fill_manual(name = "Metric", values = c("Transactions" = "skyblue")) +
  theme_minimal()
```



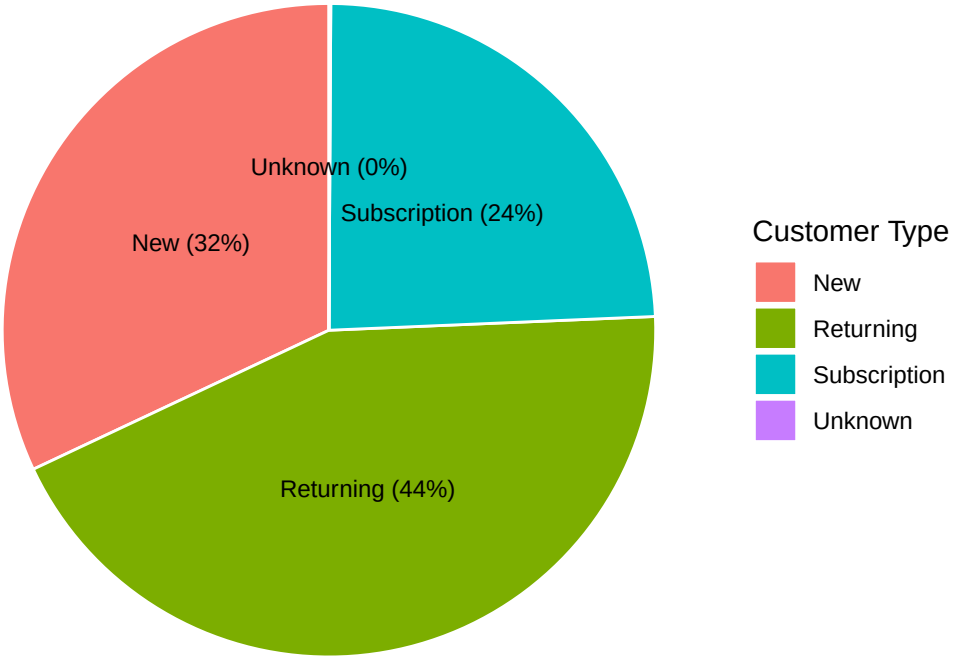
```r
customer_summary <- transaction_clean %>%
  group_by(`Customer Type`) %>%
  summarise(
    sales = n(),
    quantity_sold = sum(Quantity, na.rm = TRUE),
    total_revenue = sum(`Total Sale`, na.rm = TRUE),
    total_cost = sum(`Total Direct Cost`, na.rm = TRUE),
    total_profit = total_revenue - total_cost,
    gross_profit_margin = round((total_profit / total_revenue) * 100, 2)
  )

kable(customer_summary, caption = "Sales Summary by Customer Type")
```

Table 4: Sales Summary by Customer Type

Customer Type	sales	quantity_sold	total_revenue	total_cost	total_profit	gross_profit_margin
New	1060	3142	12942.50	5877.98	7064.52	54.58
Returning	1447	4353	16553.50	9025.32	7528.18	45.48
Subscription	803	2481	8601.75	4519.48	4082.27	47.46
Unknown	3	8	27.00	11.80	15.20	56.30

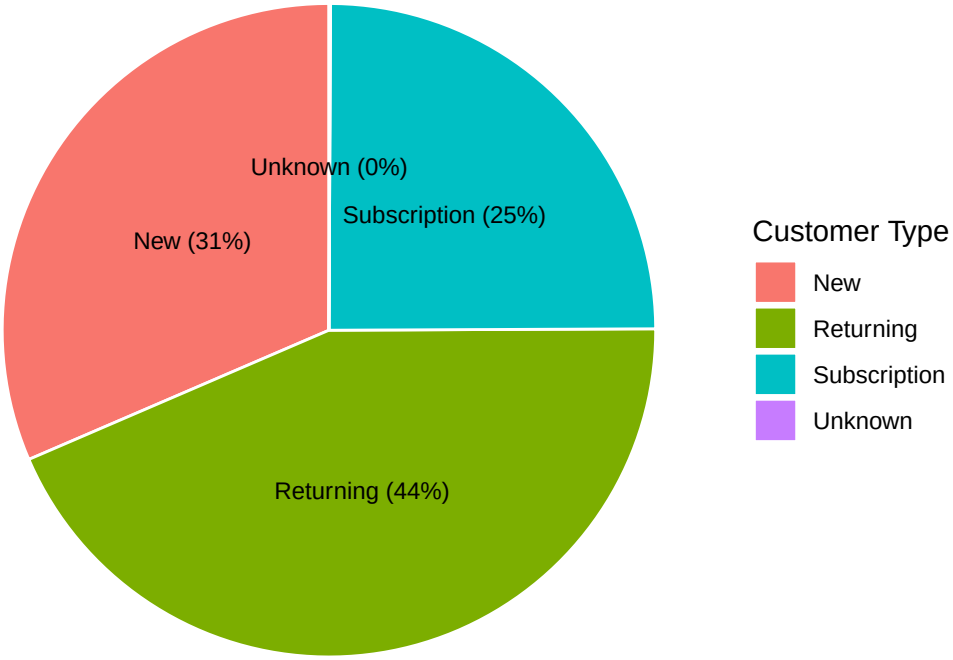
```r
# Create percentages
customer_summary <- customer_summary %>%
  mutate(
    pct_sales = sales / sum(sales),
    pct_revenue = total_revenue / sum(total_revenue),
    pct_quantity = quantity_sold / sum(quantity_sold),
    label_sales = paste0(`Customer Type`, " (", round(pct_sales * 100), "%)"),
    label_revenue = paste0(`Customer Type`, " (", round(pct_revenue * 100), "%)"),
    label_quantity = paste0(`Customer Type`, " (", round(pct_quantity * 100), "%)")
  )


# Donut chart
ggplot(customer_summary, aes(x = "", y = pct_sales, fill = `Customer Type`)) +
  geom_col(width = 1, color = "white") +
  coord_polar(theta = "y") +
  labs(title = "Donut Chart: Transaction Share by Customer Type") +
  theme_void() +
  geom_text(aes(label = label_sales), position = position_stack(vjust = 0.5), size = 3)
```

Donut Chart: Transaction Share by Customer Type



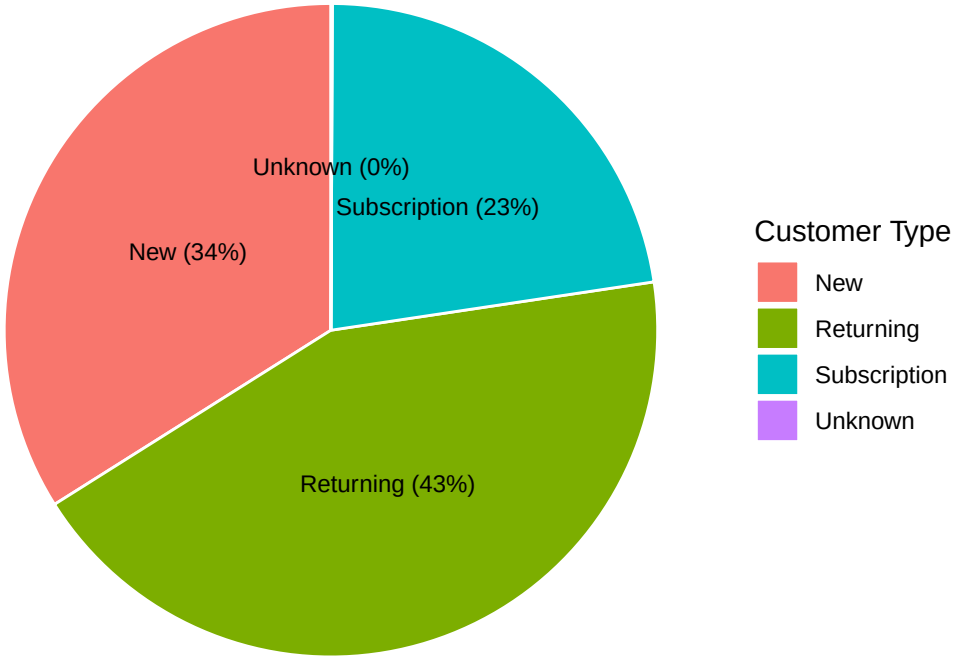

```
ggplot(customer_summary, aes(x = "", y = pct_quantity, fill = `Customer Type`)) +
  geom_col(width = 1, color = "white") +
  coord_polar(theta = "y") +
  labs(title = "Donut Chart: Quantity sold by Customer Type") +
  theme_void() +
  geom_text(aes(label = label_quantity), position = position_stack(vjust = 0.5), size = 3)
```

Donut Chart: Quantity sold by Customer Type

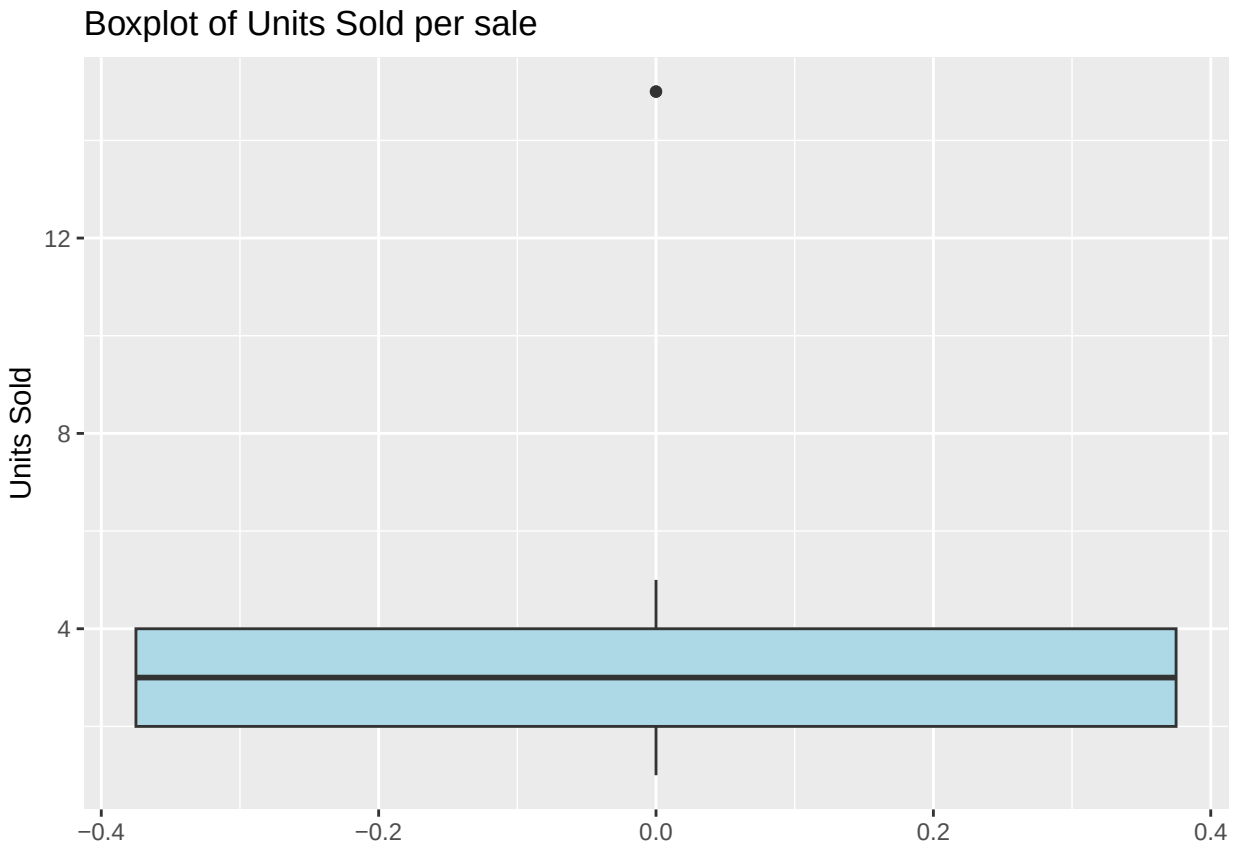



```
ggplot(customer_summary, aes(x = "", y = pct_revenue, fill = `Customer Type`)) +
  geom_col(width = 1, color = "white") +
  coord_polar(theta = "y") +
  labs(title = "Donut Chart: Revenue Share by Customer Type") +
  theme_void() +
  geom_text(aes(label = label_revenue), position = position_stack(vjust = 0.5), size = 3)
```

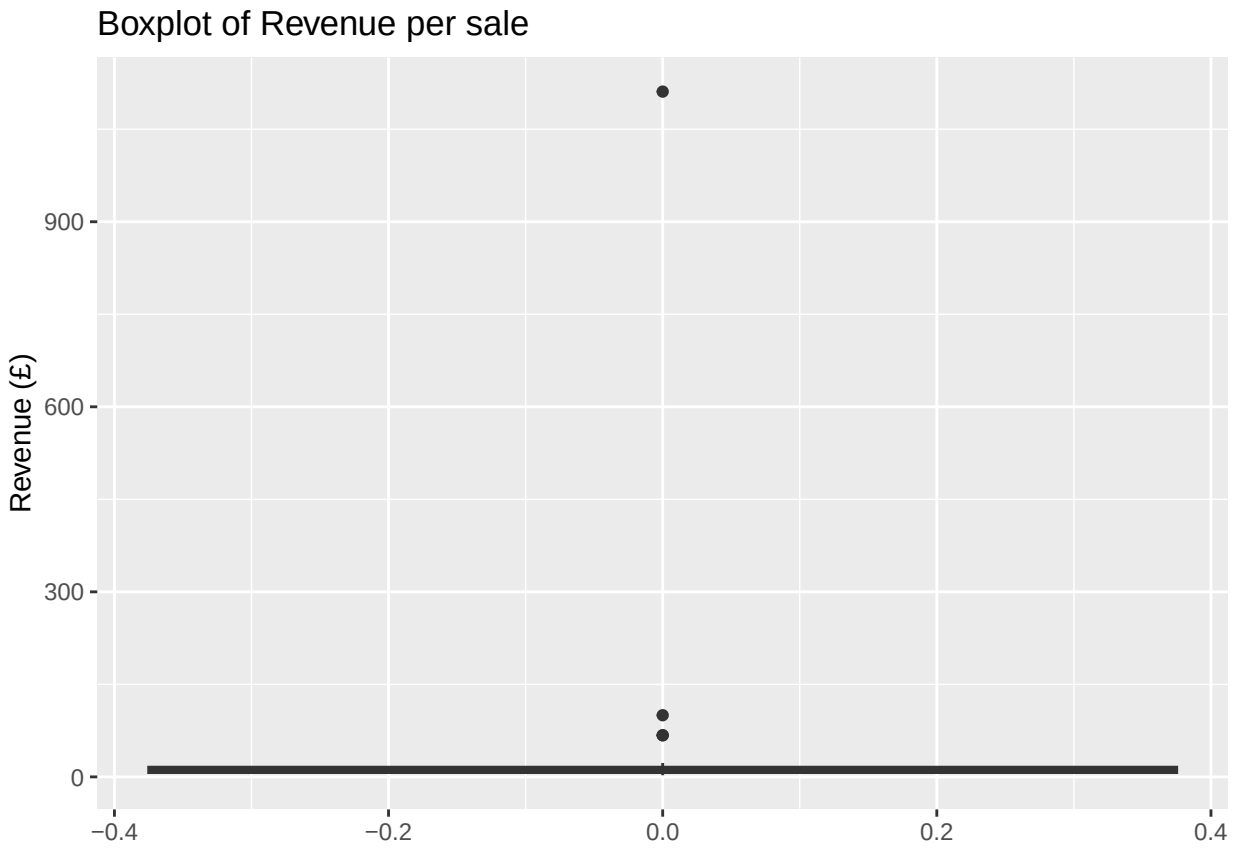
Donut Chart: Revenue Share by Customer Type




```
# Visualisation of outliers
ggplot(transaction_clean, aes(y = Quantity)) +
  geom_boxplot(fill = "lightblue") +
  labs(title = "Boxplot of Units Sold per sale", y = "Units Sold")
```




```
ggplot(transaction_clean, aes(y = `Total Sale`)) +
  geom_boxplot(fill = "salmon") +
  labs(title = "Boxplot of Revenue per sale", y = "Revenue (£)")
```



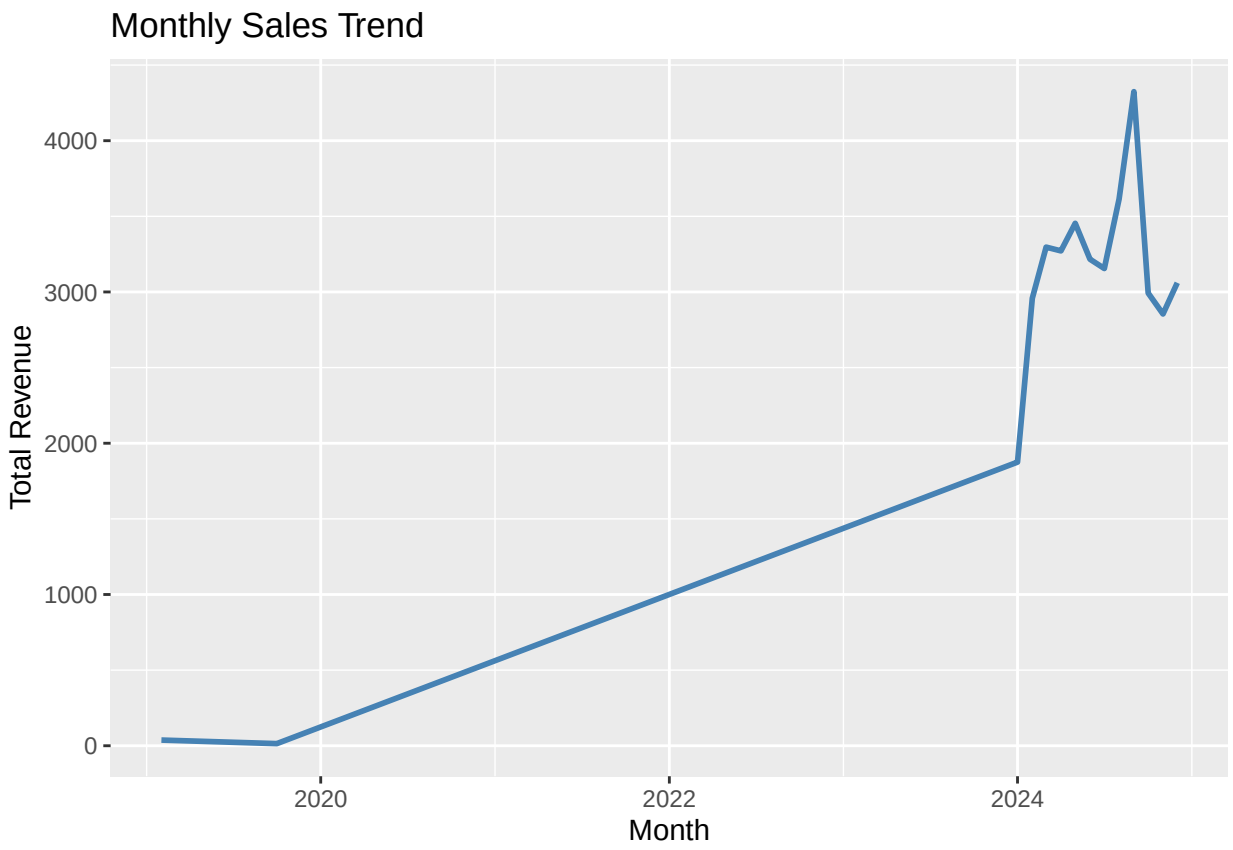

```
# Function to flag outliers based on IQR
flag_outliers <- function(x) {
  Q1 <- quantile(x, 0.25, na.rm = TRUE)
  Q3 <- quantile(x, 0.75, na.rm = TRUE)
  IQR <- Q3 - Q1
  lower_bound <- Q1 - 1.5 * IQR
  upper_bound <- Q3 + 1.5 * IQR
  which(x < lower_bound | x > upper_bound)
}
outlier_units <- flag_outliers(transaction_clean$Quantity)
outlier_revenue <- flag_outliers(transaction_clean$`Total Sale`)

transaction_clean[outlier_units, ] %>%
  head()
```

```
## # A tibble: 2 x 11
##   `Date of Sale` Product   Channel `Customer Type` Quantity `Total Sale`
##   <date>         <chr>     <chr>   <chr>              <dbl>        <dbl>
## 1 2024-06-26     GlowTonic online  New                   15         67.5
## 2 2024-09-05     GlowTonic online  Returning             15         67.5
## # i 5 more variables: `Total Direct Cost` <dbl>, Location <chr>, profit <dbl>,
## #   month <date>, year <dbl>
```

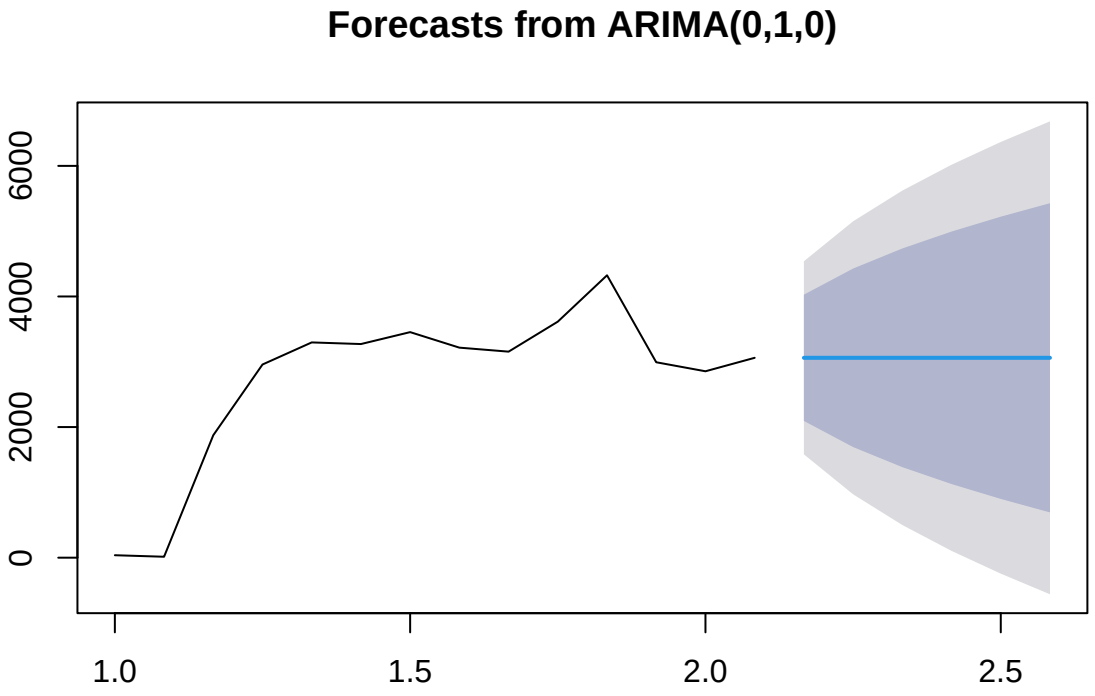
```
transaction_clean %>%
  mutate(Month = floor_date(`Date of Sale`, "month")) %>%
  group_by(Month) %>%
  summarise(TotalRevenue = sum(`Total Sale`, na.rm = TRUE)) %>%
  ggplot(aes(x = Month, y = TotalRevenue)) +
```

```
  geom_line(color = "steelblue", size = 1) +
  labs(title = "Monthly Sales Trend", y = "Total Revenue", x = "Month")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```




```
monthly_sales <- transaction_clean %>%
  mutate(month = floor_date(`Date of Sale`, "month")) %>%
  group_by(month) %>%
  summarise(sales = sum(`Total Sale`, na.rm = TRUE))

ts_data <- ts(monthly_sales$sales, frequency = 12)
fit <- auto.arima(ts_data)

forecasted <- forecast(fit, h = 6)
plot(forecasted)
```



Task 3: Strategic Interpretation

Sales Volume & Reach: VitaSip has a diverse product portfolio with some products performing significantly better than others in both transaction count and quantity sold.

Customer Base Composition: The customer base is composed of four types: New, Returning, Subscription, and Unknown. A large proportion of transactions come from New and Returning customers, suggesting strong acquisition but potentially moderate retention.

Revenue Drivers: Subscription customers may make fewer transactions overall, but they contribute a higher average order value, suggesting higher value per sale.

Channel Usage: Certain sales channels (e.g., direct-to-consumer or online) may be overperforming in volume or revenue, indicating more effective marketing or a better user experience there.

New customers account for a large number of transactions but may purchase in smaller quantities. Subscription customers show higher quantity per transaction and revenue, indicating loyalty and value. Returning customers sustain ongoing sales and help stabilize demand.

A few top-performing products (FocusFizz) dominate in both sales and quantity sold. Some products (RestRefresher) underperform, either due to limited appeal or poor positioning.

If your time series plots show monthly trends, there is a peak but followed by a decrease in total revenue.

Task 4: Looking Forward

Convert New Customers into Subscription Customers:

New customers generate high transaction volume but lower revenue per transaction. Subscription customers, though fewer, spend more per transaction and likely generate more stable revenue.

Strategic Action: Launch an automated follow-up campaign offering: Discounts or exclusive flavours for subscribing within 7 days of a first purchase

Rationalize Product Portfolio to Focus on High-Performing Products

A few products account for most sales and revenue. Several products contribute little to performance but consume shelf space, marketing, and inventory cost.

Strategic Action: Phase out underperforming SKUs and reinvest resources in high-margin products and seasonal or bundle-ready variants. Create bundles or multi-pack promotions around top products