

# Lab5Module

Meghana Mittameedi

2022-10-03

```
library(ggplot2)
library(completejourney)
library(dplyr)
library(lubridate)
library(stringr)
library(gridExtra)
library(viridis)
library(viridisLite)
```

## Weekly average quantity and average quantity per income versus department

```
transactions <- get_transactions()
demographics <- demographics
promotions <- get_promotions()

wkly_quantity_dept <- transactions %>%
  inner_join(products) %>%
  semi_join(promotions, by = "product_id") %>%
  group_by(week, department) %>%
  summarise(avg_week_qty = mean(sum(quantity)))

quantity_per_income <- transactions %>%
  left_join(products) %>%
  left_join(demographics, by = "household_id") %>%
  group_by(income, department) %>%
  summarise(avg_qty_perincome = mean(sum(quantity)))

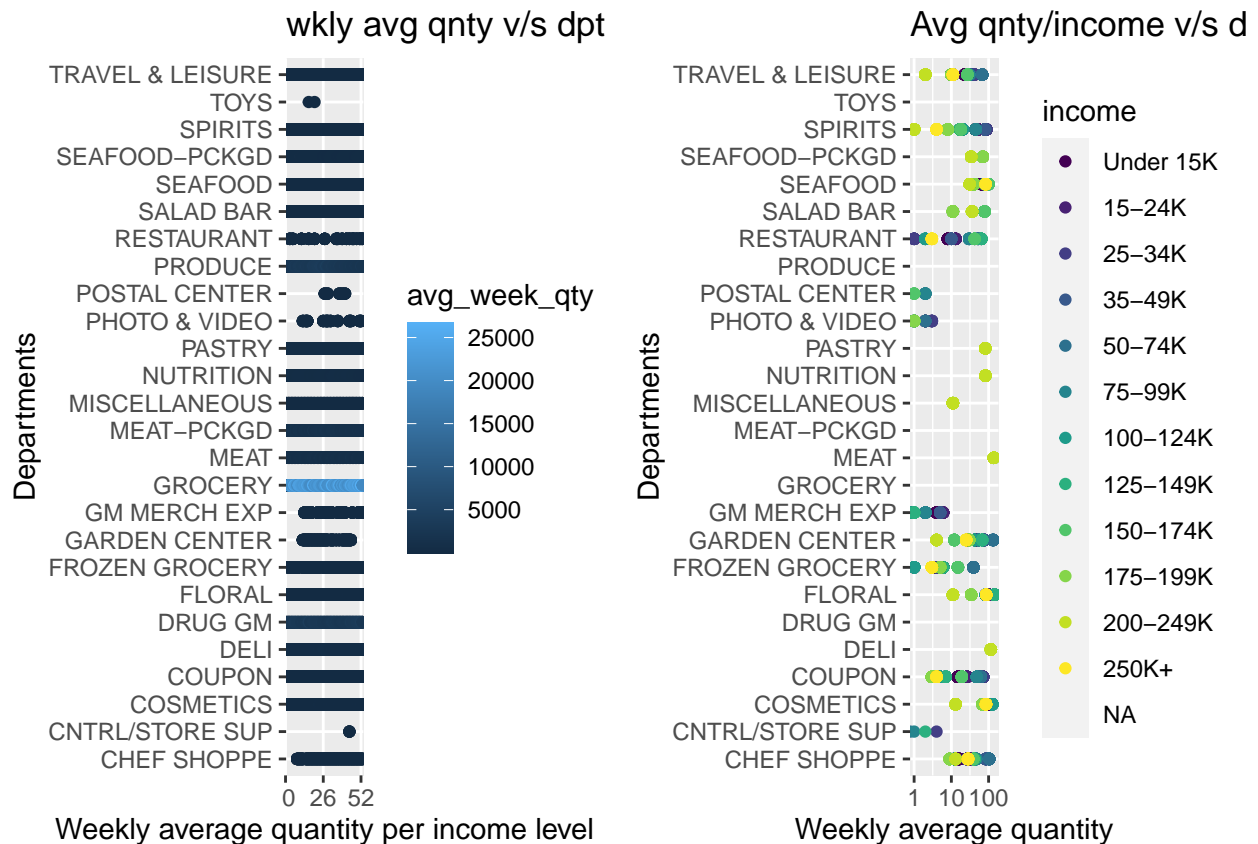
one <- quantity_per_income %>%
  inner_join(wkly_quantity_dept, by = "department")

p1 <- ggplot(one, aes(x = week , y = department, color = avg_week_qty )) +
  geom_point() +
  scale_x_discrete(limits = c(0,26,52)) +
  labs(title = "wkly avg qnty v/s dpt",
       x = "Weekly average quantity per income level",
       y = "Departments")

p2 <- ggplot(one, aes(x = avg_qty_perincome , y = department, color = income )) +
  geom_point() +
```

```
scale_x_log10(limits = c(1,144,400)) +
  labs(title = "Avg qnty/income v/s dpt",
       x = "Weekly average quantity",
       y = "Departments")

gridExtra::grid.arrange(p1, p2, nrow = 1)
```

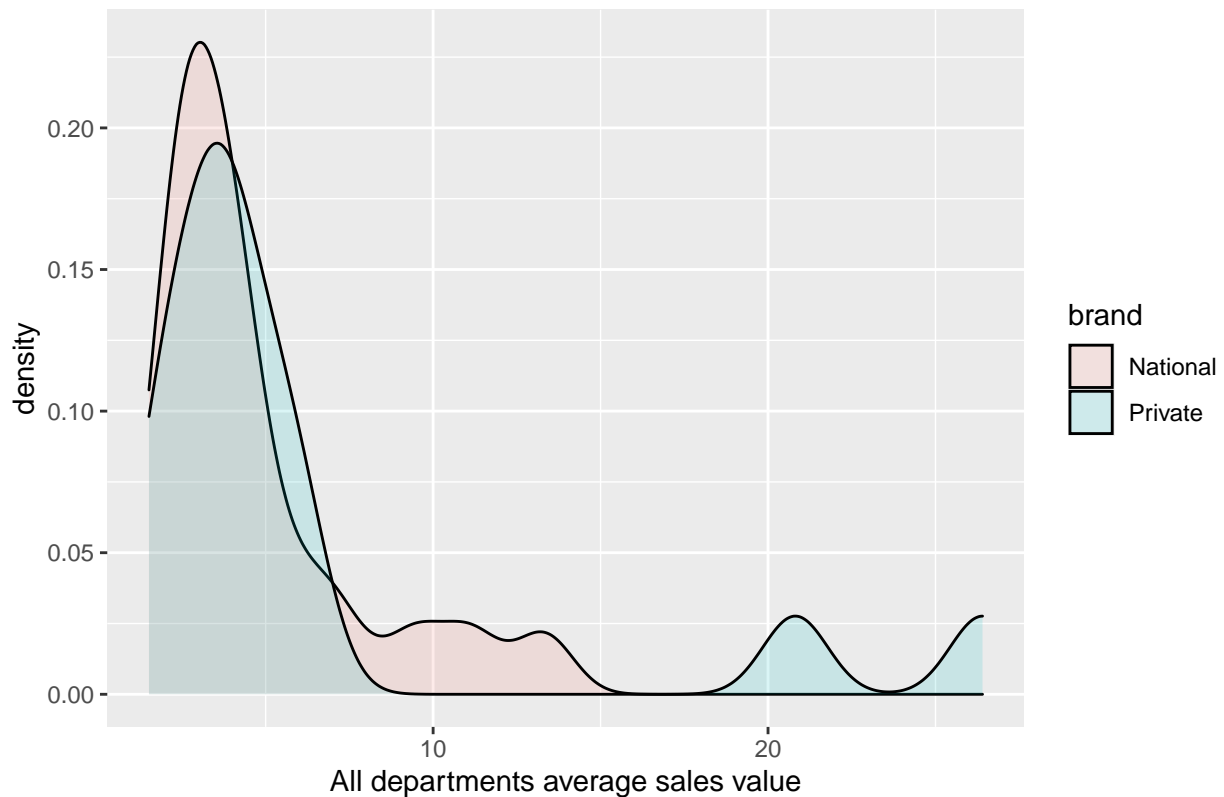


## Distribution of departments average sales value

```
transactionssample_product_demographics <- transactions_sample %>%
  inner_join(products) %>%
  inner_join(demographics) %>%
  group_by(department, brand) %>%
  summarise(n = mean(sales_value))

ggplot(transactionssample_product_demographics, aes(x = n, fill = brand)) +
  geom_density(alpha = .15) +
  labs(title = "Distribution of departments mean sales value",
       x = "All departments average sales value")
```

### Distribution of departments mean sales value



### Per day sales value of Meat

```
meat_products <- products %>%
  filter(
    str_detect(product_category, regex("MEAT", ignore_case = TRUE))) %>%
  inner_join(transactions) %>%
  mutate(
    Date = date(transaction_timestamp),
    Day = day(transaction_timestamp),
    Month = month(transaction_timestamp),
    Month_name = factor(month.abb[Month], levels = month.abb),
    week = week(transaction_timestamp),
    Weekday = wday(transaction_timestamp),
    Weekday_name = factor(Weekday,
                          levels = (1:7),
                          labels = (c("Su", "M", "Tu", "W", "Th", "F", "Sa")),
                          ordered = TRUE)
  )

meat_products <- meat_products %>%
  group_by(Date, Day, Month_name, week, Weekday_name, Month) %>%
  summarise(tot_meat_sales = sum(sales_value)) %>%
  group_by(Month_name) %>%
  mutate(MonthWeek = 1 + week - min(week))
```

```
ggplot(meat_products,aes(Weekday_name, MonthWeek, fill = tot_meat_sales)) +
  geom_tile(color = 'white') +
  scale_fill_viridis(direction = -1, name = "Meat Sales", option = "mako") +
  facet_wrap(~ Month_name, nrow = 3) +
  scale_y_reverse() +
  geom_text(aes(color = tot_meat_sales > 800, label = Day),size = 3) +
  scale_color_manual(guide = "none", values = c("black", "white")) +
  labs(title = "Per day sales value of Meat",
       subtitle = "Plotted using transaction, and products data from completejourney package.",
       x = "Day of the Week",
       y = "Week of the Month") +
  theme(strip.text = element_text(size = 13,
                                   face = "plain",
                                   color = "black"),
        panel.background = element_rect("white"))
```

## Per day sales value of Meat

Plotted using transaction, and products data from completejourney package.

