Sales Mehod Review

Sarim Rizvi

2025-06-19

Project Overview

This project analyzes the efficiency of three sales strategies — **Email**, **Call**, and **Email + Call** — using real-world product sales data from a new office stationery line. The goal is to recommend the most scalable, cost-effective method based on revenue and effort..

Data Summary

```r
sales_clean <- read_csv("data/product_sales_clean.csv")
glimpse(sales_clean)
```

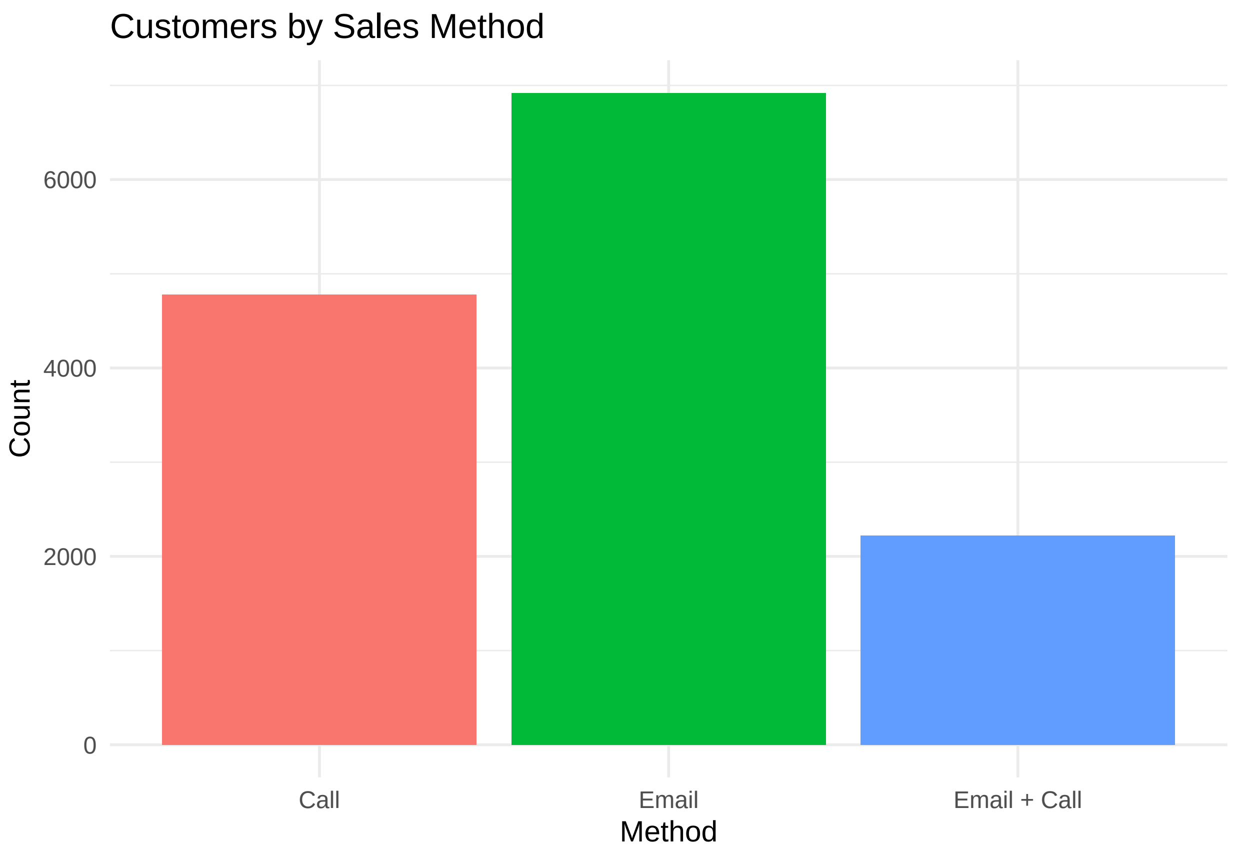
```
## Rows: 13,924
## Columns: 8
## $ week              <dbl> 3, 2, 2, 2, 5, 5, 2, 1, 4, 1, 6, 4, 3, 2, 2, 3, 4, 4~
## $ sales_method      <chr> "Email", "Call", "Email", "Email", "Email", "Email +~
## $ customer_id       <chr> "10e6d446-10a5-42e5-8210-1b5438f70922", "3729ca37-5c~
## $ nb_sold           <dbl> 9, 8, 10, 10, 10, 13, 9, 8, 10, 9, 13, 10, 9, 9, 10,~
## $ revenue           <dbl> 90.49, 42.43, 102.22, 101.22, 103.87, 188.87, 44.30,~
## $ years_as_customer <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
## $ nb_site_visits    <dbl> 28, 21, 28, 24, 22, 26, 24, 23, 27, 25, 26, 25, 27, ~
## $ state             <chr> "Illinois", "New York", "Pennsylvania", "Illinois", ~
```

- Raw data had information about 150000 customers.
- Removed 1074 rows due to missing revenue, and 2 rows due to invalid customer tenure.
- Total rows after cleaning: 13924

Exploratory Data Analysis

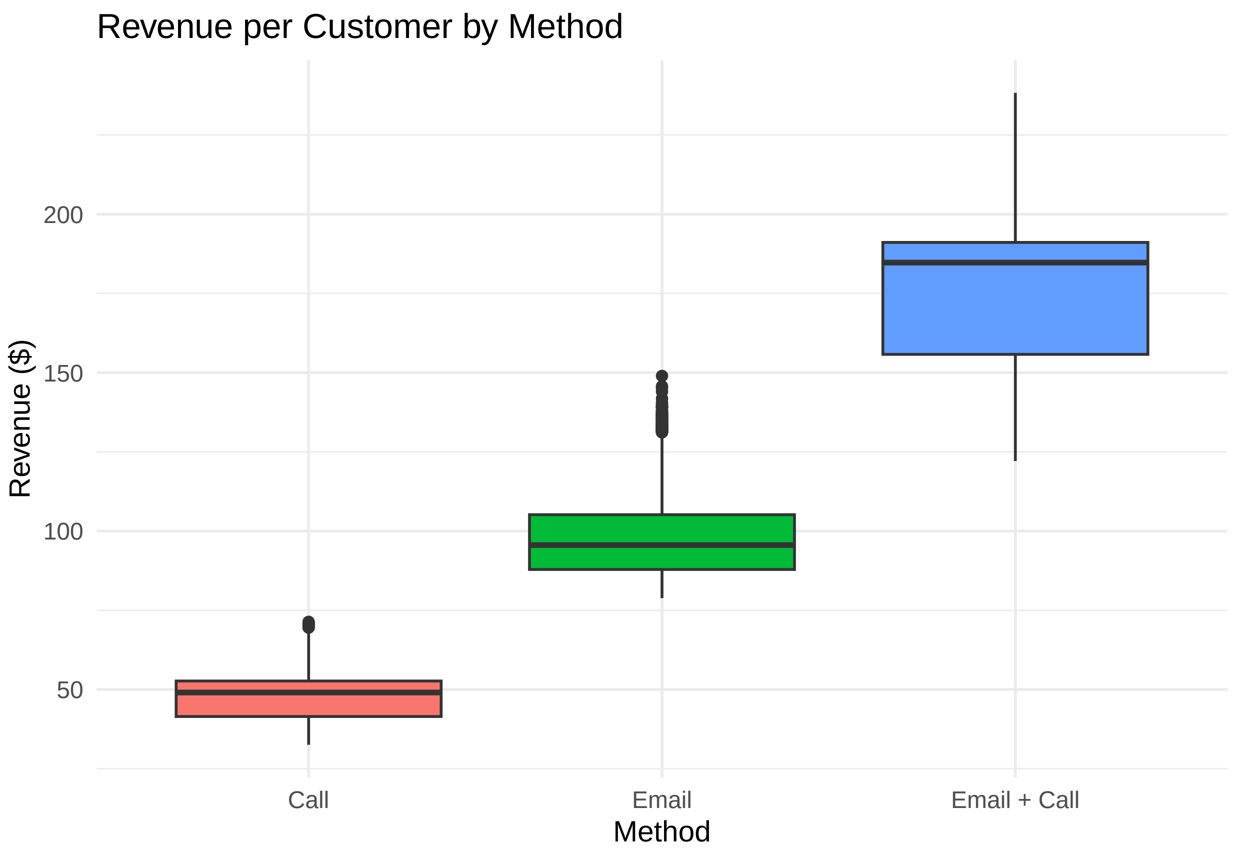
Sales Method Distribution

```r
sales_clean %>%
  count(sales_method) %>%
  ggplot(aes(x = sales_method, y = n, fill = sales_method)) +
  geom_col() +
  labs(title = "Customers by Sales Method", x = "Method", y = "Count") +
  theme(legend.position = "none")
```



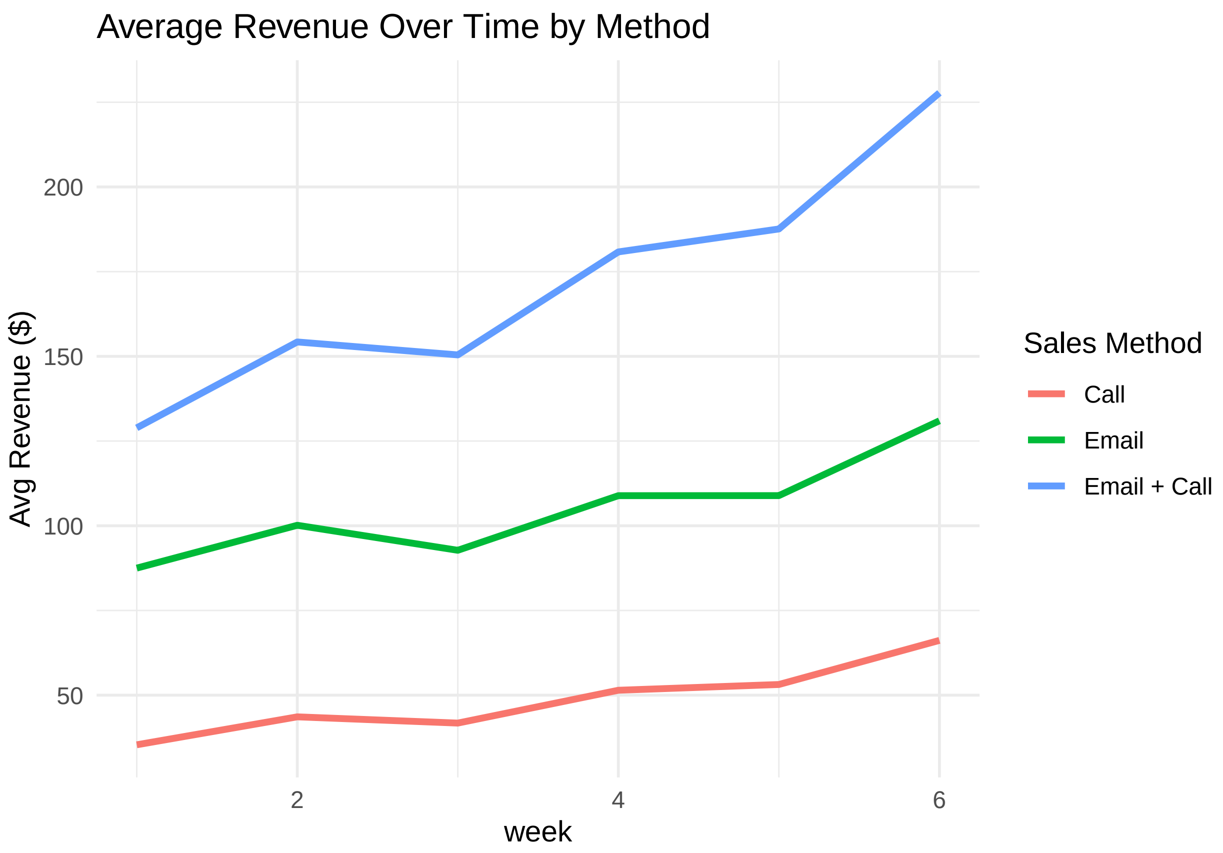
Revenue Spread over Sales Method

```
ggplot(sales_clean, aes(x = sales_method, y = revenue, fill = sales_method)) +
  geom_boxplot() +
  labs(title = "Revenue per Customer by Method", x = "Method", y = "Revenue ($)") +
  theme(legend.position = "none")
```



Revenue Over Time across Sales Method

```r
sales_clean %>%
  group_by(week, sales_method) %>%
  summarise(avg_revenue = mean(revenue), .groups = "drop") %>%
  ggplot(aes(x = week, y = avg_revenue, color = sales_method)) +
  geom_line(size = 1.2) +
  labs(title = "Average Revenue Over Time by Method", y = "Avg Revenue ($)", color = 'Sales Method')
```



Metric: Revenue per Minute

```
sales_metrics <- sales_clean %>%
  mutate(
    estimated_minutes = case_when(
      sales_method == "Email" ~ 1,
      sales_method == "Call" ~ 30,
      sales_method == "Email + Call" ~ 11
    ),
    revenue_per_unit = revenue / nb_sold,
    revenue_per_minute = revenue / estimated_minutes
  )

metric_summary <- sales_metrics %>%
  group_by(sales_method) %>%
  summarise(
    avg_revenue = mean(revenue),
    avg_units = mean(nb_sold),
    avg_rev_per_unit = mean(revenue_per_unit),
    avg_rev_per_min = mean(revenue_per_minute),
    avg_effort = mean(estimated_minutes),
    customers = n(),
    .groups = "drop"
  )
```

```
print(metric_summary)
```

```
## # A tibble: 3 x 7
##   sales_method avg_revenue avg_units avg_rev_per_unit avg_rev_per_min avg_effort
##   <chr>              <dbl>     <dbl>            <dbl>           <dbl>      <dbl>
## 1 Call                47.6      9.50             5.01            1.59         30
## 2 Email               97.1      9.72            10.0            97.1           1
## 3 Email + Call       184.      12.2             15.1            16.7          11
## # i 1 more variable: customers <int>
```

Key Insights

- **Email** yields high revenue per minute (~$97), best for scale.
- **Email + Call** brings the highest revenue per customer (~$184), ideal for high-value leads.
- **Call-only** performs poorly in both revenue and efficiency.

Recommendation

- Use **Email** as **default method** for all campaigns.
- Use **Email + Call** for **top-tier accounts** based on past spending.
- Avoid Call-only unless targeting niche, high-stakes clients.
- Monitor **revenue per minute** weekly by method.