Analysis on Cloudwalk Deliveries

Victor Reichenbach Requião

2023-05-30

Install/Run Packages for data Analysis(SETTING ENVIROMENT UP)

```r
library("tidyverse")
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```r
library("rmarkdown")
library("gridExtra")
```

```
##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##     combine
```

```r
library(knitr)
```

Import CSV and View data Columns and data type

```r
cloudwalk_dirty <- read_csv("logistic-case-v4.csv")
```

```
## Rows: 42710 Columns: 15
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): status, provider, state, city, supply_name
## dbl  (2): id, shipment_cost
## dttm (7): sales_order_created_at, device_order_created_at, processing_at, in...
## date (1): delivery_estimate_date
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

Data cleaning

Rows 14883, 14838 and 14715 seems to have the month and day in an incorrect order, since i cannot evaluate the accurate delivery time im removing them from the list to avoid bias in the results.

```
cloudwalk <- cloudwalk_dirty[-c(14883, 14838, 14715), ]
```

Observations:

- KPI for logistics are shipment cost, On time delivery(OTD) and delivery time
- Also is relevant to get data on customer experience

Data Analisys step-by-step

created a column calculating the time difference between sale order and delivery in days

```
cloudwalk$delivery_time <- difftime(cloudwalk$delivered_at, cloudwalk$device_order_created_at, units =
average_delivery_time <- mean(cloudwalk$delivery_time, na.rm = TRUE)
```

Adding the region column to the dataframe using the column “state” as a reference

```
region <- data.frame(
  state = c("SP", "RJ", "MG", "ES", "RS", "SC", "PR", "BA", "PE", "CE", "PA", "AM", "MT", "GO", "DF")
)
state_to_region <- c(
  "AC" = "Norte", "AP" = "Norte", "AM" = "Norte", "PA" = "Norte", "RO" = "Norte", "RR" = "Norte", "TO"
  "AL" = "Nordeste", "BA" = "Nordeste", "CE" = "Nordeste", "MA" = "Nordeste", "PB" = "Nordeste", "PE" =
  "DF" = "Centro-Oeste", "GO" = "Centro-Oeste", "MT" = "Centro-Oeste", "MS" = "Centro-Oeste",
  "ES" = "Sudeste", "MG" = "Sudeste", "RJ" = "Sudeste", "SP" = "Sudeste",
  "PR" = "Sul", "RS" = "Sul", "SC" = "Sul"
)
cloudwalk <- cloudwalk %>%
  mutate(region = state_to_region[state])
```

Calculating difference in delivery time per provider

```
provider_delivery_time <- cloudwalk %>%
  group_by(provider) %>%
  summarize(average_delivery_time = mean(delivery_time, na.rm = TRUE))
provider_delivery_time <- provider_delivery_time %>%
  arrange(desc(average_delivery_time))
```

Calculate the average, longest and shortest delivery time per provider

```r
avg_delivery_time <- cloudwalk %>%
  group_by(provider) %>%
  summarize(average_delivery_time = mean(delivery_time, na.rm = TRUE))
longest_delivery_time <- cloudwalk %>%
  group_by(provider) %>%
  summarize(max_delivery_time = max(delivery_time, na.rm = TRUE))
shortest_delivery_time <- cloudwalk %>%
  group_by(provider) %>%
  summarize(min_delivery_time = min(delivery_time, na.rm = TRUE))
delivery_time_summary <- left_join(avg_delivery_time, longest_delivery_time, by = "provider") %>%
  left_join(shortest_delivery_time, by = "provider")
```

Calculate the average, max and min shipping cost per provider

```r
provider_shipping_costs <- cloudwalk %>%
  group_by(provider) %>%
  summarize(max_shipping_cost = max(shipment_cost),
            min_shipping_cost = min(shipment_cost))
average_shipping_cost <- cloudwalk %>%
  group_by(provider) %>%
  summarize(average_cost = mean(shipment_cost, na.rm = TRUE))
combined_shipping_costs <- left_join(provider_shipping_costs, average_shipping_cost, by = "provider")
timeandcostsummary <-left_join(delivery_time_summary, combined_shipping_costs, by = "provider")
```

Maximum, Minimum and Average cost and time per provider

```r
kable(timeandcostsummary)
```

provider	average_delivery_time	max_delivery_time	min_delivery_time	max_shipping_cost	min_shipping_cost	average_cost
provider 1	8.584015 days	65.81338 days	0.8386445 days	123.03	11.69	23.40006
provider 2	5.854187 days	47.34163 days	0.2079403 days	135.38	7.46	21.04981

Adding OTD column to data

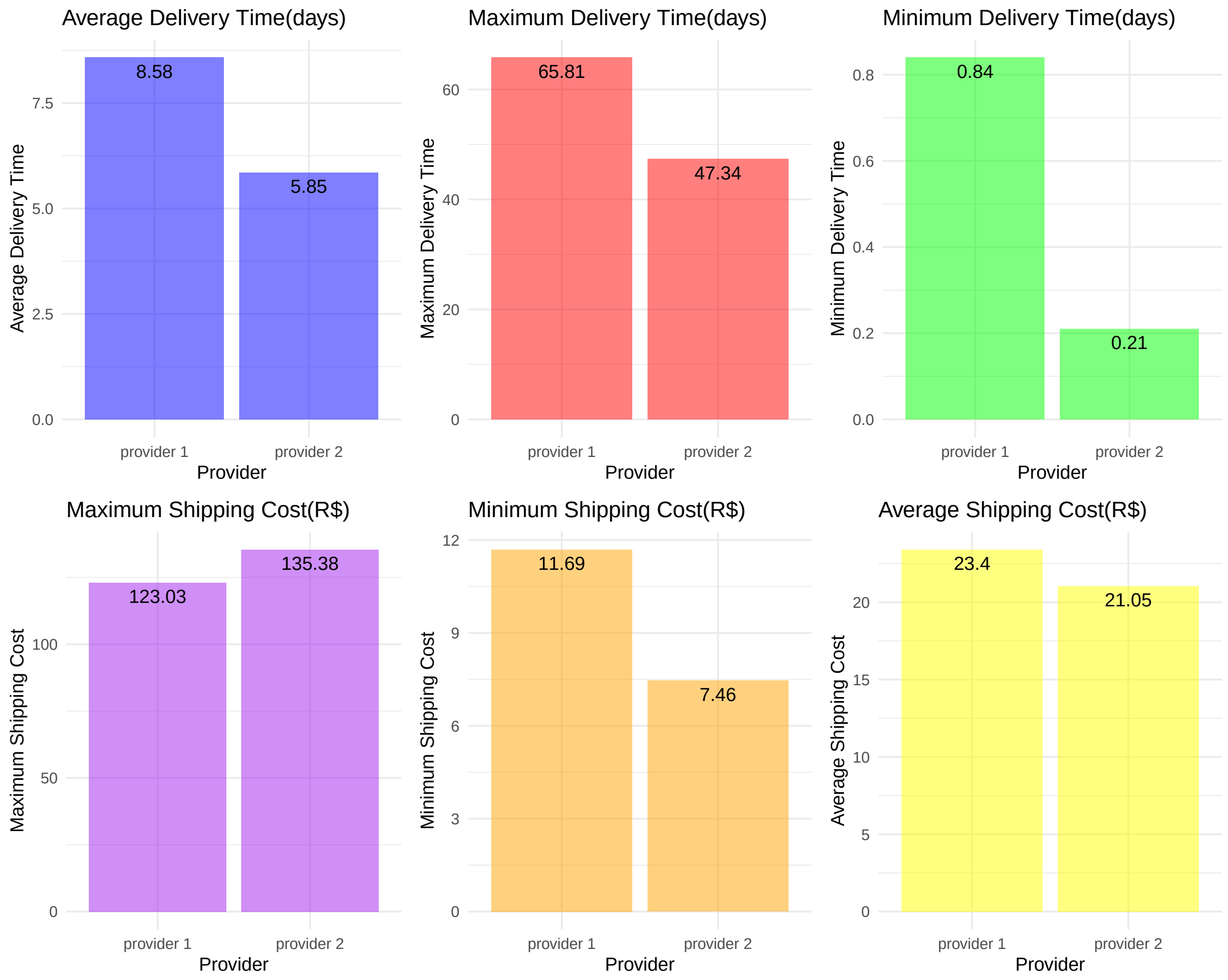
```r
cloudwalk$delivery_estimate_date <- as.Date(cloudwalk$delivery_estimate_date)
cloudwalk$delivered_at <- as.Date(cloudwalk$delivered_at)
cloudwalk <- cloudwalk %>%
  mutate(OTD = ifelse(delivered_at <= delivery_estimate_date, "On Time", "Delayed"))
cloudwalk <- cloudwalk %>%
  mutate(difference_otd = as.numeric(delivered_at - delivery_estimate_date))
```

Organizing data for plots

```r
summary_stats <- cloudwalk %>%
  group_by(provider) %>%
  summarize(
    average_delivery_time = mean(delivery_time, na.rm = TRUE),
    max_delivery_time = max(delivery_time, na.rm = TRUE),
    min_delivery_time = min(delivery_time, na.rm = TRUE),
    max_shipping_cost = max(shipment_cost),
    min_shipping_cost = min(shipment_cost),
    average_cost = mean(shipment_cost)
  )

summary_stats$average_delivery_time <- as.numeric(summary_stats$average_delivery_time)
summary_stats$max_delivery_time <- as.numeric(summary_stats$max_delivery_time)
summary_stats$min_delivery_time <- as.numeric(summary_stats$min_delivery_time)
summary_stats$max_shipping_cost <- as.numeric(summary_stats$max_shipping_cost)
summary_stats$min_shipping_cost <- as.numeric(summary_stats$min_shipping_cost)
summary_stats$average_cost <- as.numeric(summary_stats$average_cost)

summary_stats$average_delivery_time <- round(summary_stats$average_delivery_time, 2)
summary_stats$max_delivery_time <- round(summary_stats$max_delivery_time, 2)
summary_stats$min_delivery_time <- round(summary_stats$min_delivery_time, 2)
summary_stats$max_shipping_cost <- round(summary_stats$max_shipping_cost, 2)
summary_stats$min_shipping_cost <- round(summary_stats$min_shipping_cost, 2)
summary_stats$average_cost <- round(summary_stats$average_cost, 2)
```

Creating the bar plots

```r
plots <- list(
  ggplot(summary_stats, aes(x = provider, y = average_delivery_time)) +
    geom_col(fill = "blue", alpha = 0.5) +
    geom_text(aes(label = average_delivery_time), vjust = 1.5) +
    labs(title = "Average Delivery Time(days)", x = "Provider", y = "Average Delivery Time") +
    theme_minimal(),

  ggplot(summary_stats, aes(x = provider, y = max_delivery_time)) +
    geom_col(fill = "red", alpha = 0.5) +
    geom_text(aes(label = max_delivery_time), vjust = 1.5) +
    labs(title = "Maximum Delivery Time(days)", x = "Provider", y = "Maximum Delivery Time") +
    theme_minimal(),

  ggplot(summary_stats, aes(x = provider, y = min_delivery_time)) +
    geom_col(fill = "green", alpha = 0.5) +
    geom_text(aes(label = min_delivery_time), vjust = 1.5) +
    labs(title = "Minimum Delivery Time(days)", x = "Provider", y = "Minimum Delivery Time") +
    theme_minimal(),

  ggplot(summary_stats, aes(x = provider, y = max_shipping_cost)) +
    geom_col(fill = "purple", alpha = 0.5) +
    geom_text(aes(label = max_shipping_cost), vjust = 1.5) +
    labs(title = "Maximum Shipping Cost(R$)", x = "Provider", y = "Maximum Shipping Cost") +
    theme_minimal(),

  ggplot(summary_stats, aes(x = provider, y = min_shipping_cost)) +
    geom_col(fill = "orange", alpha = 0.5) +
    geom_text(aes(label = min_shipping_cost), vjust = 1.5) +
    labs(title = "Minimum Shipping Cost(R$)", x = "Provider", y = "Minimum Shipping Cost") +
    theme_minimal(),

  ggplot(summary_stats, aes(x = provider, y = average_cost)) +
    geom_col(fill = "yellow", alpha = 0.5) +
    geom_text(aes(label = average_cost), vjust = 1.5) +
    labs(title = "Average Shipping Cost(R$)", x = "Provider", y = "Average Shipping Cost") +
    theme_minimal()
)

grid.arrange(grobs = plots, nrow = 2, ncol = 3)
```

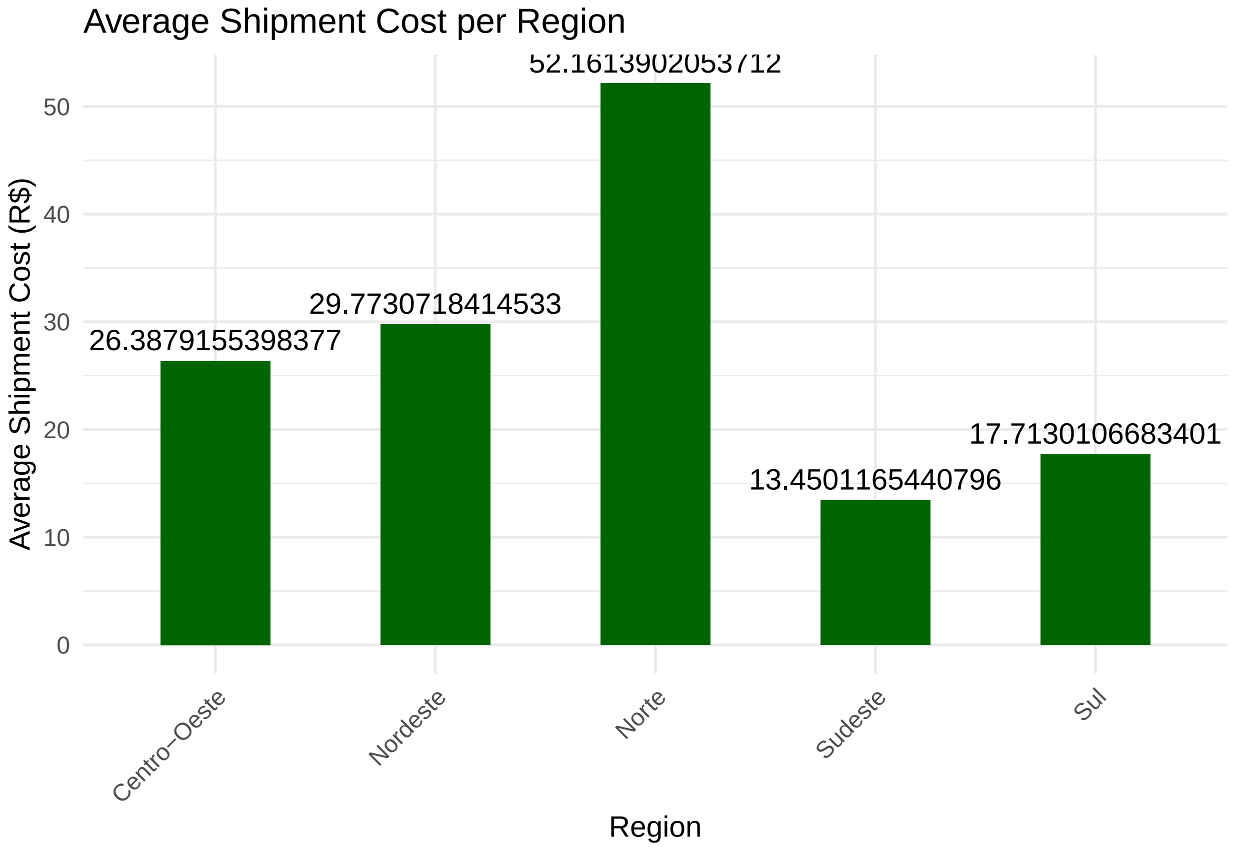


Shipment cost and time per region

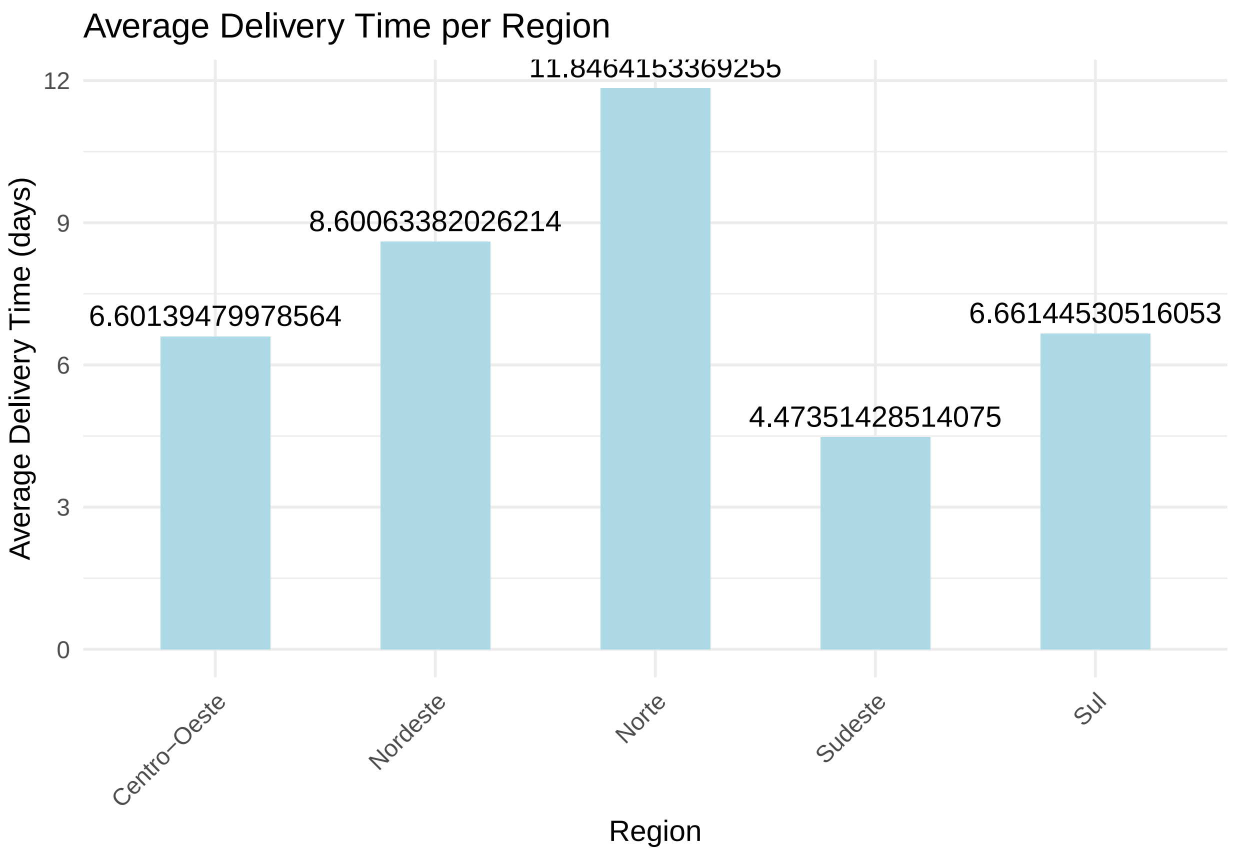
```
avg_delivery_shipment <- cloudwalk %>%
  group_by(region) %>%
  summarise(
    avg_delivery_time = mean(as.numeric(delivery_time), na.rm = TRUE),
    avg_shipment_cost = mean(as.numeric(shipment_cost), na.rm = TRUE)
  )
View(avg_delivery_shipment)
```

Creating graphics

```
ggplot(avg_delivery_shipment, aes(x = region, y = avg_shipment_cost)) +
  geom_bar(stat = "identity", fill = "darkgreen", width = 0.5) +
  geom_text(aes(label = avg_shipment_cost), vjust = -0.5, color = "black") +
  labs(x = "Region", y = "Average Shipment Cost (R$)", title = "Average Shipment Cost per Region") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



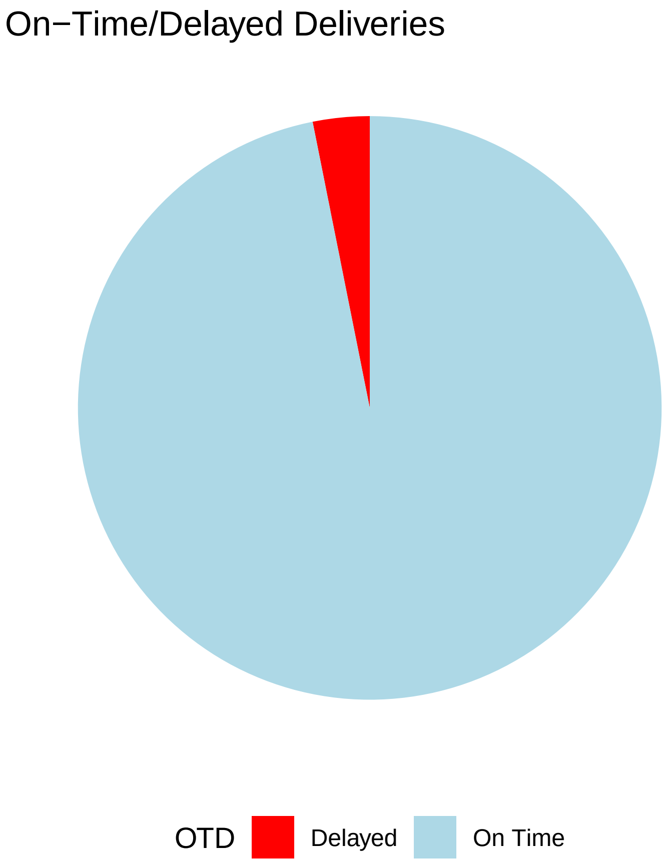
```
ggplot(avg_delivery_shipment, aes(x = region, y = avg_delivery_time)) +
  geom_bar(stat = "identity", fill = "lightblue", width = 0.5) +
  geom_text(aes(label = avg_delivery_time), vjust = -0.5, color = "black") +
  labs(x = "Region", y = "Average Delivery Time (days)", title = "Average Delivery Time per Region") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



OTD percentagem graphic

The on-time deliveries represent 95% of all shipments

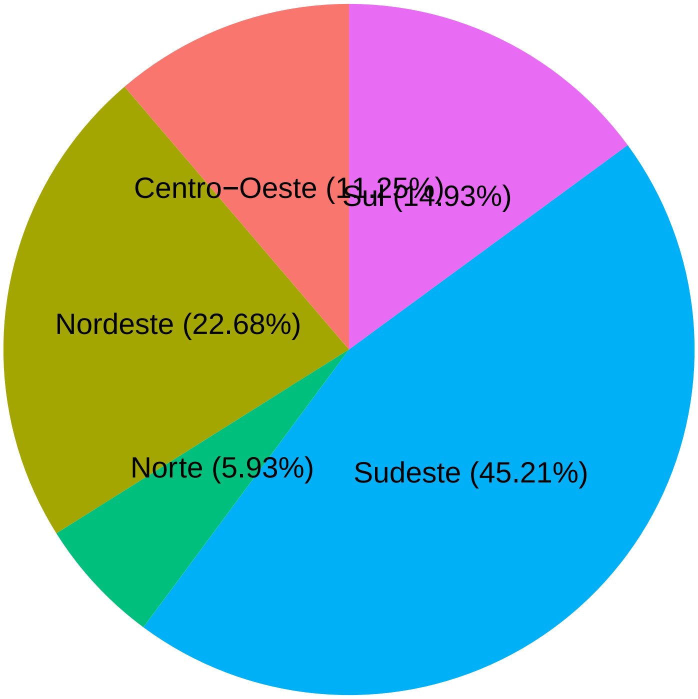
```
otd_percentage <- cloudwalk %>%
  group_by(OTD) %>%
  summarise(count = n()) %>%
  mutate(percentage = count / sum(count) * 100)

otd_percentage <- otd_percentage[complete.cases(otd_percentage), ]

ggplot(otd_percentage, aes(x = "", y = percentage, fill = OTD)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y", start = 0) +
  theme_void() +
  theme(legend.position = "bottom", legend.box = "horizontal") +
  guides(fill = guide_legend(title = "OTD")) +
  labs(title = "On-Time/Delayed Deliveries") +
  scale_fill_manual(values = c("On Time" = "lightblue", "Delayed" = "red")) +
  theme(legend.direction = "horizontal",
        legend.box.just = "center",
        legend.title.align = 0.5,
        legend.margin = margin(t = 10))
```



Creating graphs for some more information

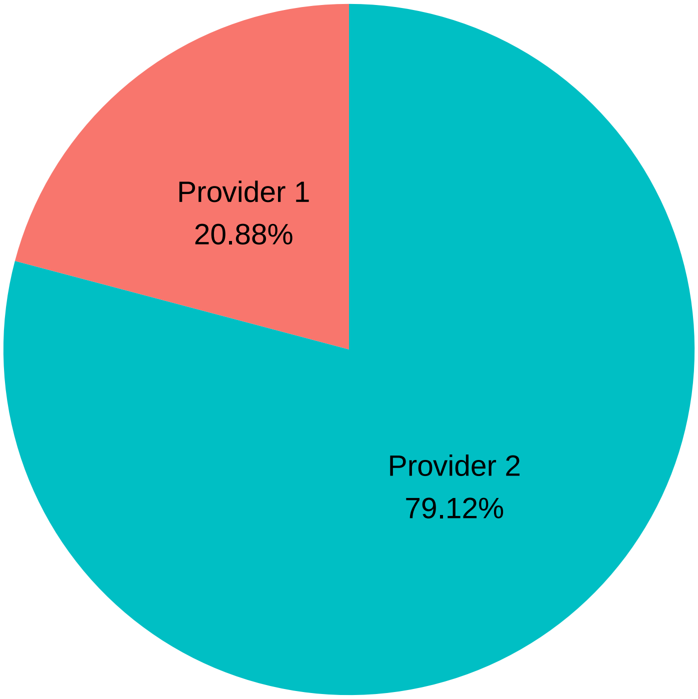
GRAPH FOR REPRESENTING THE PERCENT OF SHIPPING FOR EACH REGION

```
regions <- c("Sudeste", "Nordeste", "Sul", "Centro-Oeste", "Norte")
percentages <- c(45.21, 22.68, 14.93, 11.25, 5.93)
df <- data.frame(regions, percentages)
df$cumulative <- cumsum(df$percentages)
ggplot(df, aes(x = "", y = percentages, fill = regions)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar(theta = "y") +
  geom_text(aes(label = paste0(regions, " (", round(percentages, 2), "%)")),
            position = position_stack(vjust = 0.5)) +
  theme_void() +
  theme(legend.position = "none")
```



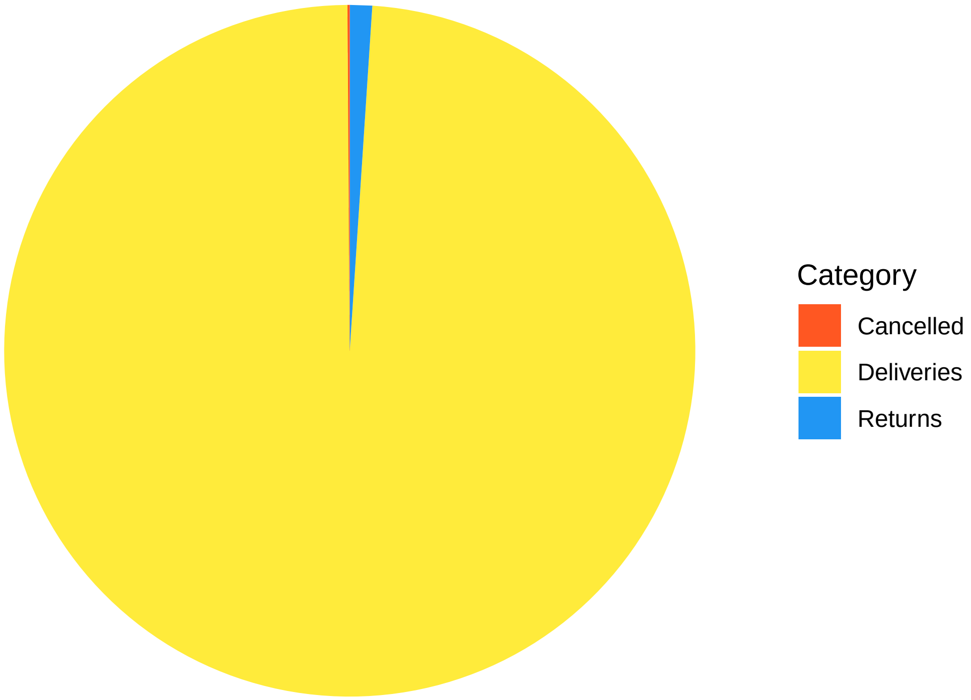
DATA FOR REPRESENTING SHIPMENTS PER PROVIDER

```
providers <- c("Provider 1", "Provider 2")
transactions <- c(8917, 33793)
total_transactions <- sum(transactions)
percentages <- transactions / total_transactions * 100
df <- data.frame(providers, percentages)
ggplot(df, aes(x = "", y = percentages, fill = providers)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar(theta = "y") +
  geom_text(aes(label = paste0(providers, "\n", round(percentages, 2), "%")),
            position = position_stack(vjust = 0.5)) +
  theme_void() +
  theme(legend.position = "none")
```



deliverys - returns - cancelled

```
deliveries <- 42220
returns <- 445
cancelled <- 45
total_orders <- deliveries + returns + cancelled
delivery_percentage <- deliveries / total_orders * 100
return_percentage <- returns / total_orders * 100
cancel_percentage <- cancelled / total_orders * 100
df <- data.frame(
  Category = c("Deliveries", "Returns", "Cancelled"),
  Percentage = c(delivery_percentage, return_percentage, cancel_percentage)
)
colors <- c("#FF5722", "#FFEB3B", "#2196F3")
ggplot(df, aes(x = "", y = Percentage, fill = Category)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar(theta = "y") +
    scale_fill_manual(values = colors) +
  theme_void()
```


Category
Cancelled
Deliveries
Returns