Lesson 3: Filtering to a Comparable Market Segment (CMS)

AUTHOR
John Fariss

PUBLISHED
April 3, 2025

Introduction

Welcome to Lesson 3! In this lesson, you will: - Filter the dataset to a relevant market segment - Create visualizations to analyze price trends

Step 1: Filtering Data

Before we being filtering data, we should start with a setup chunk. This is good practice for all Quarto documents. Instead of loading packages by chunk, we create a single setup chunk to load packages and set options. This helps with transparency and reproducibility.

```
# I always name this chunk: Setup
# load needed packages
pacman::p_load( # this temporarily loads pacman to load the other packages
              tidyverse, # this is a key package that loads ggplot2, dplyr, lubridate and more
              leaflet # this is a mapping package
              )

# turn scientific notation off
options(scipen = 999)
```

Now we are ready to filter transactions from our **ADF** to our **CMS**, those properties similar to the subject property. We'll go back a little over 1 year, and narrow to properties with GLA of 1200-1800 sf, assuming our subject is 1500 sf.

```
# Filter CMS based on time, size, and location
CMS <- ADF %>% filter( # make sure tidyverse or dplyr was loaded
  DateSale >= as.Date("2024-01-01"), # sets oldest date
  DateSale <= as.Date("2025-03-15"), # sets newest date
  GLA >= 1200, GLA <= 1800 # sets our GLA limits
)
```

Step 2: Visualizing the CMS

Let's see the results of our filter by creating a histogram.

Histogram of CMS

```
# Create histogram
ggplot() +
  geom_histogram(data = CMS,aes(PriceSale), # specify plot type, data & variable
                 binwidth=10000, # set size of bin widths
                 fill="blue", # set fill color as blue
                 color = "gray") + # set line color as gray
  labs(title = "Distribution of CMS Sale Prices",
       x = "Reported Sale Price") +
  theme_minimal() # add a theme to change the look of the plot
```

Note that the above plot was built very similarly to the histogram we built for the ADF in the prior lesson. However, there is 1 key difference...we specified the data and variable from within the plot type (**geom_histogram**) function. This is an option that allows you to layer graphics. Additionally, we introduced the **theme_** function that alters the appearance of your plots.

Let's continue exploring the CMS with a bar plot and a box plot to see how the subject fits within the CMS.

Plot of Bedroom Counts

```
# Create a bar plot of CMS showing volume by bedroom count
ggplot(CMS, aes(x = Bedrooms)) +
  geom_bar(na.rm = TRUE, fill = "orange") +
  labs(title = "Volume of Sales by Bedroom Count",
       x = "Bedroom Count",
       y = "Volume")
```

If the subject is a 3-bedroom home, then it fits well into this market segment.

Plot of GLA Distribution

```
#Create a box and whisker graph of GLA indicating subject position
ggplot(data = CMS) + # we specify the data
  geom_boxplot((aes(GLA,"")), # specify variable to plot and make y blank
               color = "gray20", # specify line color
               outlier.alpha = .5, # set any outliers to transparent
               fill = "lightblue") + # set fill color
  labs(title = "Living Area", # create our labels
       y = "", # prevents a y from appearing
       x = "Square Feet") +
  theme_bw() + # set a different theme
  annotate("point", x = 1500, # add the subject's GLA as a point
           y = "", color = "darkblue", size = 2) +
  annotate("text", x = 1500, # sets a label over the point above
           y = "", label = "Subject", vjust = -1, color = "darkblue")
```

We can see that the subject's GLA is above the median, but well within the predominant (25%-75%) range.

Distribution of Sale Prices by Pool Amenity

```
# Boxplot of Price per Square Foot
ggplot(CMS, aes(x = as.factor(PoolType), y = PriceSale,
                fill = as.factor(PoolType))) + # this will color plots by amenity type
  geom_boxplot() + # specify plot type
  labs(title = "Price by Pool/Spa Amenity",  # add labels
       x = "",
       y = "Sale Price") +
  theme_minimal() + # set our theme
  theme(legend.position = "none") # turn off the legend
```

These box plots show the range of sale prices for those sales with the specified pool amenity. Could this be good support for an adjustment using simple methods?

Step 3: Analyze Market Conditions

Now that we know the subject fits well in the CMS, let's determine market conditions on the subject's competing market segment.

Price Indexing of CMS

```
# Create a scatter plot of price per square foot for CMS
ggplot(CMS, aes(DateSale, PriceSale)) + #specify data and variables
      geom_point(na.rm=TRUE) +  # specify plot and that NAs may exist
      geom_smooth(method = lm, # add our linear method smoother
                  formula = y ~ x, # specify y varies with x
                  se = FALSE, # turn standard error shading off
                  na.rm=TRUE, # indicate NAs may exist
                  color="green") + # set line color
      geom_smooth(method = lm, # add 2nd linear method smoother
                  formula = y ~ poly(x,3), # add 3rd order poly formula
                  se = FALSE, na.rm=TRUE, # set se and na options
                  color="blue") + # set color different from prior line
      labs(title = "Price Index of Competing Market Segment", # add labels
           x = "Reported Sale Date",
           y = "Reported Sale Price") +
      theme_light() # set our theme
```

The 3rd order polynomial helps us confirm the linear trend, and helps our eyes visualize the trend. The 2 lines should have contrasting colors.

Now we can determine the daily adjustment using a few lines of code.

```
# Specify linear model and obtain coefficents (slope[2]) for time adjustment by sale price
DailyTrend <- lm(PriceSale ~ DateSale, data = CMS)
# store result as an object
```

```
DailyAdjust <- coef(DailyTrend) [[2]] # stores just the adjustment

DailyAdjust
```

To calculate the adjustment, subtract the sales' date from the effective date for the number of days differences. Multiply the result by **DailyAdjust** for the total adjustment. You can use similar methods to plot price per square foot and determine a trend for price per square foot.

In the next lesson, we will build a **table of the CMS and map** it.