ANALISI DEL MERCATO IMMOBILIARE DEL TEXAS

Ente certificatore/Scuola : Profession AI

Corso: Statistica Descrittiva

Docente: Giuseppe Dejan Lucido

Studente: Luca Marletta

Data di riconsegna: 25/09/2024

Svolgimento: Ai fini della certificazione di fine corso di Statistica Descrittiva tenuto da Profession AI, mi è stato consegnato un dataset in formato .csv, nominato nel seguente modo "**realestate_texas.csv**". Ai fini di analisi dati del dataset in oggetto, oltre alle libreria installate in precedenza come “ggplot2, dplyr e moments” , ho importato anche il dataset da esaminare tramite riga di comando di ambiente R ed, attraverso l’interfaccia RStudio, ho generato di prassi il seguente codice per l’import:

```
#import delle librerie e del dataset
getwd()
library(dplyr)
library(ggplot2)
library(moments)
setwd(getwd())
dir()
data_texas_real_estate<- read.csv("realestate_texas.csv", sep=",")
```

Attraverso la funzione **class()** ed **str()** possiamo indagare la struttura del dataset salvato precedentemente nella variabile **data_texas_real_estate:**

```
#indagare la struttura dati
class(data_texas_real_estate)
str(data_texas_real_estate)
```

```
> str(data_texas_real_estate)
'data.frame':   240 obs. of  8 variables:
 $ city            : chr  "Beaumont" "Beaumont" "Beaumont" "Beaumont" ...
 $ year            : int  2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 ...
 $ month           : int  1 2 3 4 5 6 7 8 9 10 ...
 $ sales           : int  83 108 182 200 202 189 164 174 124 150 ...
 $ volume          : num  14.2 17.7 28.7 26.8 28.8 ...
 $ median_price    : num  163800 138200 122400 123200 123100 ...
 $ listings        : int  1533 1586 1689 1708 1771 1803 1857 1830 1829 1779 ...
 $ months_inventory: num  9.5 10 10.6 10.6 10.9 11.1 11.7 11.6 11.7 11.5 ...
```

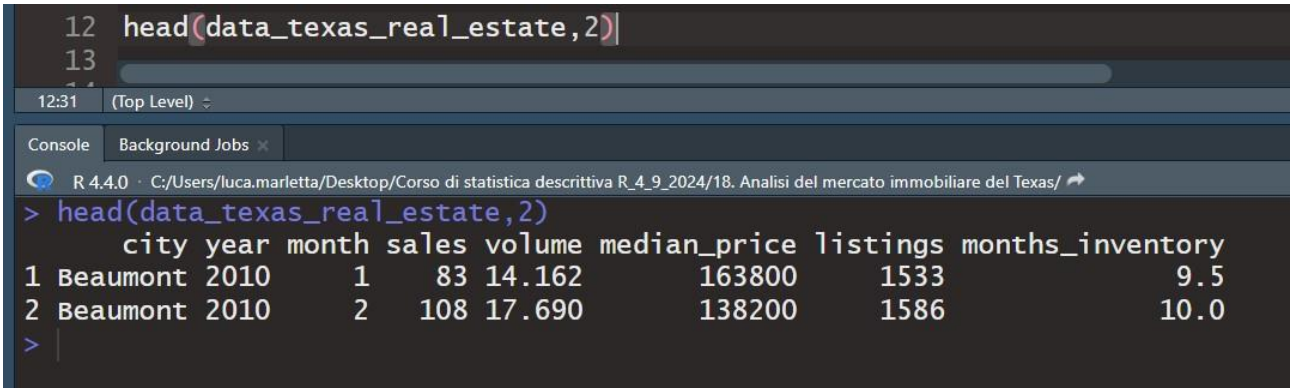
Dall’analisi risulta che il dataset sia composto da **8 variabili**, e **240 osservazioni** per ogni variabile. Tramite la funzione **colSums(is.na(data_texas_real_estate))** si evince che il dataset non presenta valori nulli al suo interno.

```
colSums(is.na(data_texas_real_estate))
```

```
> colSums(is.na(data_texas_real_estate))
          city           year          month          sales         volume
             0              0              0              0              0
  median_price       listings months_inventory
             0              0              0
>
```

ANALISI 1 – DESCRIZIONE DELLE VARIABILI DEL DATASET

Tramite la funzione di R **head(dataset,2),** è possibile ottenere un print in consolle R delle intestazioni delle colonne che contengono i nomi delle variabili, e le entries delle prime 2 righe:



Le variabili visualizzate in output della funzione hear(), descrivono quanto di seguito in elenco:

- **city**: qualitativa nominale, rappresenta la città di riferimento in cui è stata effettuata la misurazione statistica delle altre variabili continue.
- **year**: quantitativa continua, trattata come qualitativa ordinale per questo contesto specifico. Rappresenta l'anno di riferimento.
- **month**: Qualitativa nominale (ciclica) ma codificata in numeri, e trattata come fattore ordinato da gennaio a dicembre. Rappresenta il mese di riferimento in cui è stata effettuata la misurazione statistica delle altre variabili continue.
- **sales**: quantitativa continue, su scala di rapporti. Rappresenta il numero totale di vendite.
- **volume**: quantitativa continue, su scala di rapporti, e rappresenta il valore totale delle vendite in milioni di dollari
- **median_price**: quantitativa continue, su scala di rapporti e rappresenta prezzo mediano di vendita in dollari
- **listings**: La variabile listings, che rappresenta il numero totale di annunci attivi, è tecnicamente una variabile discreta, in quanto può assumere solo valori interi (non ha senso parlare di mezzo annuncio). Tuttavia, poiché il numero di possibili valori è ampio e ha un punto zero assoluto, in questa analisi statistica viene trattata come una variabile continua su scala di rapporti. Questo permette di applicare metodi statistici comuni per le variabili continue, come il calcolo della media, deviazione standard, e altre analisi.
- **months_inventory**: quantitativa continue, su scala di rapporti. Essa rappresenta nel dataset quantità di tempo necessaria per vendere tutte le inserzioni correnti al ritmo attuale delle vendite, espresso in mesi.

ANALISI 2. CALCOLA INDICI DI POSIZIONE, VARIABILITÀ E FORMA PER TUTTE LE VARIABILI PER LE QUALI HA SENSO FARLO, PER LE ALTRE CREA UNA DISTRIBUZIONE DI FREQUENZA. COMMENTA TUTTO BREVEMENTE.

Attraverso la funzione **dim(data_texas_real_estate)[1]** otteniamo il numero di osservazioni **n** (240 osservazioni) che ci tornerà utile per le successive analisi, mentre attraverso la funzione **attach(data_texas_real_estate)** otteniamo l'accesso rapido alle colonne dei vettori tramite il nome diretto della relativa proprietà del dataset. Per il calcolo degli **indici di posizione (media, mediana, 1° e 3° quartile)** delle variabili quantitative del dataset è stata utilizzata la funzione **summary(data_texas_real_estate)** che mi ha permesso in un'unica schermata per di avere la visione generale degli indici di posizione, oltre alle altre informazioni circa il range (min e max) della distribuzione delle variabili variabili quantitative:

```
25 #INDICI DI POSIZIONE
26 continue_var_summary <- summary(data_texas_real_estate[, c("sales", "volume", "median_price",
27                                                            "listings", "months_inventory")])
28 continue_var_summary
29
30
```

27:60 (Top Level) R Scrip

Console Background Jobs

R 4.4.0 · C:/Users/luca.marletta/Desktop/18. Progetto_da riconsegnare/Progetto_da riconsengare/

```
> continue_var_summary
     sales           volume         median_price       listings    months_inventory
 Min.   : 79.0   Min.   : 8.166   Min.   : 73800   Min.   : 743   Min.   : 3.400
 1st Qu.:127.0   1st Qu.:17.660   1st Qu.:117300   1st Qu.:1026   1st Qu.: 7.800
 Median :175.5   Median :27.062   Median :134500   Median :1618   Median : 8.950
 Mean   :192.3   Mean   :31.005   Mean   :132665   Mean   :1738   Mean   : 9.193
 3rd Qu.:247.0   3rd Qu.:40.893   3rd Qu.:150050   3rd Qu.:2056   3rd Qu.:10.950
 Max.   :423.0   Max.   :83.547   Max.   :180000   Max.   :3296   Max.   :14.900
> |
```

Insight indici di posizione:

- **Sales:** distribuzione relativamente ampia con un intervallo di 344 vendite, **media 192.3 maggiore** della relativa mediana che risulta essere pari a 175;
- **Volume**: distribuzione con intervallo di 75.381M$, **media** di 31.005M€ **maggiore della mediana** che risulta essere pari a 27.062M.
- **Median_price**: distribuzione con intervallo di 106200$, **media** 132662$ **minore della mediana** che risulta essere pari a 134500$
- **Listings**: distribuzione con intervallo 2553 annunci, con **media** di 1738 **maggiore della mediana** che risulta essere pari a 1618 annunci.
- **Months_inventory**: distribuzione con intervallo di 11.5 mesi, con media di 9.193 **maggiore della mediana** 8.95.

Per le variabili city, year e month ho proceduto in modo diverso:

- **City**: ho codificato il vettore in livelli, considerando il vettore city con un costrutto di fattori, tramite la funzione as.factor (city), in modo da capire quanti livelli (in questo caso città) vengono considerate nello studio statistico all'interno del dataset principale.

```
city <- as.factor(city)
levels(city)
table(city)
```

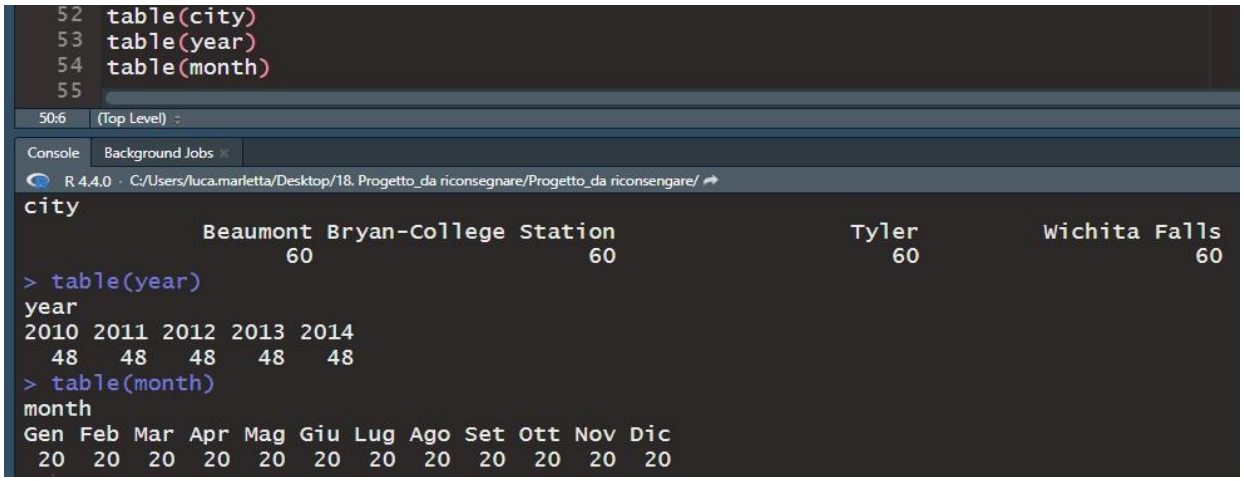
- **Year**: ho codificato in fattori la varibile year (tramite **as.factor(year, ordered=TRUE))**, trattandola come una qualitativa ordinale (codificata in numeri), ordinata secondo un ordine crescente.

```
year <- factor(year, ordered = TRUE)
levels(year)
year_freq<-data.frame(table(year))
```

- **Months**: ho codificato in 12 livelli la variabile di month, ordinando in mesi in base all'etichette ed ordine crescente da Gennaio (month 1) a Dicembre (month 12):

```
month <- factor(month,
                levels = 1:12,
                labels = c("Gen", "Feb", "Mar", "Apr", "Mag", "Giu", "Lug", "Ago", "Set", "Ott", "Nov", "Dic"),
                ordered = TRUE)
```

A questo punto ho proceduto a riassumere le **frequenze di ogni osservazione per i vettori city, year, e month**, tramite la funzione **table(),** e riassunti nel seguente output della consolle di R:



Si evince dalle **frequenze di osservazione** di ogni proprietà indagate tramite la funzione table(), che le variabili city, year e month risultano essere **equidistribuite, infatti andando per ordine:**

- **City**: le città analizzate sono **4**, **Beaumont, Bryan-College Station, Tyler e Wichita Falls, e** per la variabile city sono presenti 60 osservazioni per ogni città (240 osservazioni totali).

- **Year**: i periodi in cui sono state fatte le rilevazioni sono 5 anni e vanno dal 2010 al 2014 (inclusi). Per ogni anno sono state fatte 48 osservazioni, cioè 1 ogni mese dell'anno considerato, per 4 Città diverse per 12 mesi nell'anno (1*4*12=**48 osservazioni**/anno considerato). La visione è più chiara se si costruisce una tabella di contigenza **table(year,city):**

```
> table(year,city)
      city
year   Beaumont Bryan-College Station Tyler Wichita Falls
  2010       12                    12    12            12
  2011       12                    12    12            12
  2012       12                    12    12            12
  2013       12                    12    12            12
  2014       12                    12    12            12
```

- **Month**: il numero sottointende il numero ordinale dei mesi in cui sono state effettute le rilevazioni, che va da Gennaio (1) a Dicembre (12). Ogni mese nel dataset compare 20 volte in quanto viene preso il meso di riferimento ed indagato per ogni città (4 città) per 5 anni (dal 2010 al 2014 inclusi), quindi 1*4*5=**20 osservazioni.**

2.1 DISTIBUZIONI DI FREQUENZA PER LE VARIABILI CONTINUE (SALES, VOLUME)

VARIABILE SALES: la distribuzione è stata creata dividendo in intervalli secondo la regola di Sturges:

Nbins= $1 + \log_2(n)$ con n=numero di osservazioni del dataset (240)

Attraverso questa relazione è possibile trovare il numero di intervalli su cui basare gli intervalli dell'istogrammi di frequenza per la distribuzione sales. Dividendo, quindi il range del vettore sales in Nbins, otterremo **l'ampiezza di ogni intervallo** per la variabile sales. **E' stata identificata la classe modale in (115,153] vendite**

```
62 #sales
63 table(sales)
64 Nbins<-(1+log2(n))
65 bins_amplitude_sales<-(((max(sales))-min(sales))/Nbins)
66 sales_classes=cut(sales,(seq((min(sales)-3),(max(sales)+3),bins_amplitude_sales)))
67 ni <- table(sales_classes)
68 fi <- ni/n
69 Ni <- cumsum(table(sales_classes))
70 Fi <- Ni/n
71 distr_sales <- as.data.frame(cbind(ni,fi,Ni,Fi))
72 distr_sales <- round(distr_sales,2)
73 distr_sales
74
```

```
71:49 (Untitled)
Console  Background Jobs
R 4.4.0 · C:/Users/luca.marletta/Desktop/Corso di statistica descrittiva R_16_9_2024/18. Analisi del mercato immobiliare del Texas/Progetto/
> distr_sales
          ni   fi  Ni   Fi
(76,115]  42 0.17  42 0.17
(115,153] 53 0.22  95 0.40
(153,192] 44 0.18 139 0.58
(192,230] 29 0.12 168 0.70
(230,269] 23 0.10 191 0.80
(269,308] 26 0.11 217 0.90
(308,346] 10 0.04 227 0.95
(346,385]  9 0.04 236 0.98
(385,424]  4 0.02 240 1.00
```

VARIABILE VOLUME:

con lo stesso metodo ho identificato il valore della **classe modale** per la variabile volume, **ovvero quella di classe (13.60,22.1] Milioni di €.**

```
#volume
table(volume)
bins_amplitude_volume<-(((max(volume))-min(volume))/Nbins)
volume_classes=cut(volume,(seq((min(volume)-3),(max(volume)+3),bins_amplitude_volume)))
ni <- table(volume_classes)
fi <- ni/n
Ni <- cumsum(ni)
Fi <- Ni/n
distr_volume <- as.data.frame(cbind(ni,fi,Ni,Fi))
distr_volume <- round(distr_volume,2)
distr_volume
```

85:13 (Untitled)

Console Background Jobs

R 4.4.0 · C:/Users/luca.marletta/Desktop/Corso di statistica descrittiva R_16_9_2024/18. Analisi del mercato immobiliare del Texas/Progetto/

```
> distr_volume
            ni   fi  Ni   Fi
(5.17,13.6] 31 0.13  31 0.13
(13.6,22.1] 55 0.23  86 0.36
(22.1,30.6] 51 0.21 137 0.57
(30.6,39]   37 0.15 174 0.72
(39,47.5]   23 0.10 197 0.82
(47.5,55.9] 22 0.09 219 0.91
(55.9,64.4]  9 0.04 228 0.95
(64.4,72.9]  8 0.03 236 0.98
(72.9,81.3]  3 0.01 239 1.00
```

2.2 CALCOLO DEGLI INDICI DI VARIABILITÀ E DI FORMA DELLE SEGUENTI PROPRIETÀ QUANTITATIVE CONTINUE DEL DATASET:

- sales,
- volume,
- median_price,
- listings
- months_inventory

ho provveduto al calcolo della varianza, dev.standard,range ed IQR **(indici di variabilità)** e delle skewness e curtosi **(indici di forma, rispettivamente momenti 3° e 4° della distribuzione)** di ogni variabile sopra elencata. Il calcolo del range è stato effettuato tramite il calcolo della differenza dei valori min() e max() delle distribuzioni di ogni singola proprietà, mentre gli indici di forma sono stati calcolati con le opportuni funzioni skewness() e kurtosis() della libreria "moments". Ho provveduto a condensare tutte queste misure di variabilità, oltre al valore della medie, all'interno di un singolo dataframe che ho chiamato **"variability_shape_dataframe".** A seguire il codice R, e i suoi risultati:

```r
#Indice di variabilità e forma:
sales_range<-max(sales)-min(sales)
volume_range<-max(volume)-min(volume)
median_price_range<-max(median_price)-min(median_price)
listings_range<-max(listings)-min(listings)
months_inventory_range<-max(months_inventory)-min(months_inventory)

CV <- function(x)
{return(sd(x)/mean(x)*100)}
variability_shape_dataframe<- data.frame(
  Variabile=c("sales", "volume", "median_price", "listings", "months_inventory"),
  Range=c(sales_range, volume_range,median_price_range,listings_range,months_inventory_range),
  media=c(mean(sales), mean(volume),mean(median_price),mean(listings),mean(months_inventory)),
  SD=c(sd(sales), sd(volume),sd(median_price),sd(listings),sd(months_inventory)),
  CV=c(CV(sales), CV(volume),CV(median_price),CV(listings),CV(months_inventory)),
  IQR=c(IQR(sales), IQR(volume),IQR(median_price),IQR(listings),IQR(months_inventory)),
  skewness=c(skewness(sales), skewness(volume),skewness(median_price),skewness(listings),skewness(months_inventory)),
  kurtosis=c((kurtosis(sales)-3), (kurtosis(volume)-3),(kurtosis(median_price)-3),(kurtosis(listings)-3),(kurtosis(months_inventory)-3)))

variability_shape_dataframe<- variability_shape_dataframe %>%
  mutate_if(is.numeric, round, digits = 2)
variability_shape_dataframe
```

```
> variability_shape_dataframe
         Variabile     Range      media        SD    CV      IQR skewness kurtosis
1            sales    344.00     192.29     79.65 41.42   120.00     0.72    -0.31
2           volume     75.38      31.01     16.65 53.71    23.23     0.88     0.18
3     median_price 106200.00 132665.42 22662.15 17.08 32750.00    -0.36    -0.62
4         listings   2553.00    1738.02    752.71 43.31  1029.50     0.65    -0.79
5 months_inventory     11.50       9.19      2.30 25.06     3.15     0.04    -0.17
```

- La varianza di una variabile statistica fornisce una misura della variabilità dei valori assunti della variabile, ovvero la misura di quanto essi si discostano dalla rispettiva media. Valori alti della varianza indicano alti valori di dispersione dei dati intorno alla media della distribuzione di appartenenza. Di più immediata comprensione vi è la deviazione standard (radice quadrata della varianza), in quanto la deviazione standard ha le stesse unità di misura delle variabili. Per esempio per la variabile sales abbiamo la seguente sintesi statistica:

192.3 +- 79.65 sales (media sales +- dev. Std sales)

- Per confrontare le variabilità fra diversi vettori si può paragonare il valore dei coefficiente di variazione fra le variabili**,** cioè la deviazione standard che presenta la variabile rispetto alla propria media, ed esprimere il risultato il % (Coefficiente di variazione %). La variabile con la CV% più alta nel dataset presenterà la variabilità più alta.

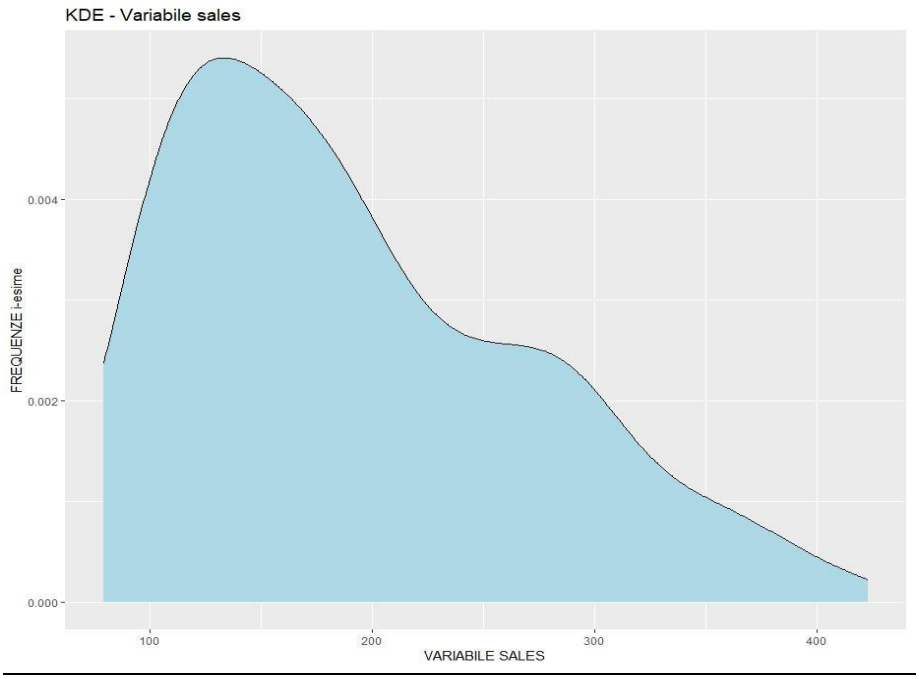
- Per indagare la forma delle distribuzioni e, nello specifico la deviazione dalla simmetria di una distribuzione di dati viene eseguito il calcolo delle skewness. Questo parametro descrittivo indica una deviazione della simmetria della distribuzione quando è diverso da 0. Quando la skewsness è diversa da zero, infatti, la moda della distribuzione non coincide più con media aritmetica dei suoi valori e può assumere skewsness>0, quando la media dei valori > moda e la coda della distruzione è allungata verso i valori più elevati a distribuzione. Ne consegue una coda allungata verso destra. Quando invece la skewness<0, la media<moda e la coda della distribuzione è allungata verso sinistra.

- Un altro parametro di forma che elenchiamo in questo studio, e che descrive un allontanamento dalla normalità distributiva, è la curtosi (kurtosis in inglese); anche questa funzione può assumere valori diversi da 0 quando la normalità non è rispetta, e nello specifico:

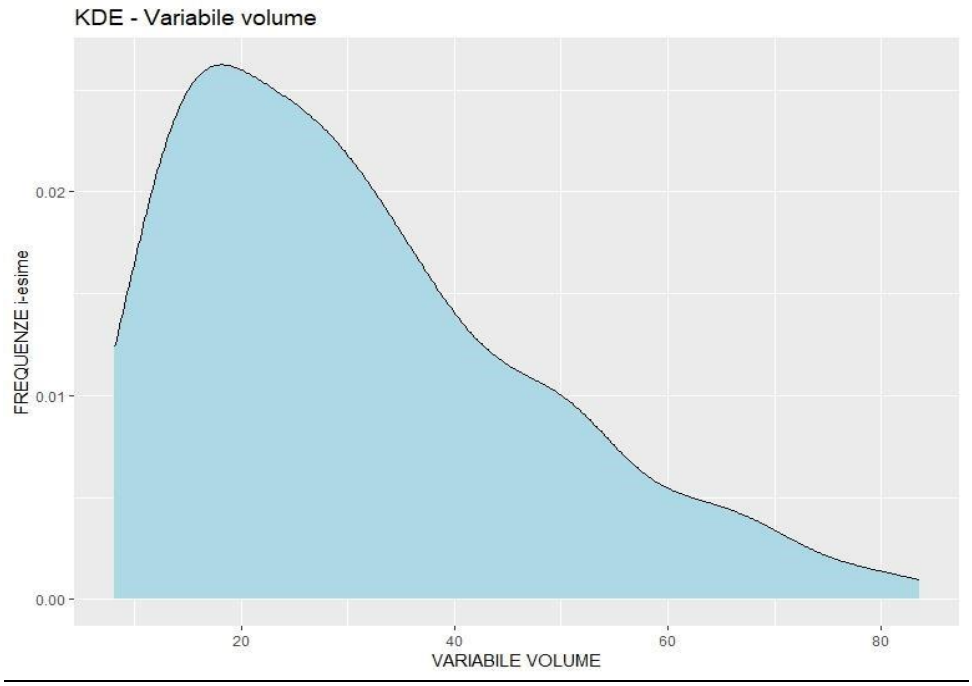
  ○ Curtosi =0 (normocurtica, curva distribuita normalmente ○
  Curtosi>0 (leptocurtica, curva che assumente forme appuntitie) ○
  Curtosi<0 (platicurtica, curva che assume forme schiacciate)

2.3 INSIGHT su INDICI DI VARIABILITÀ E FORMA

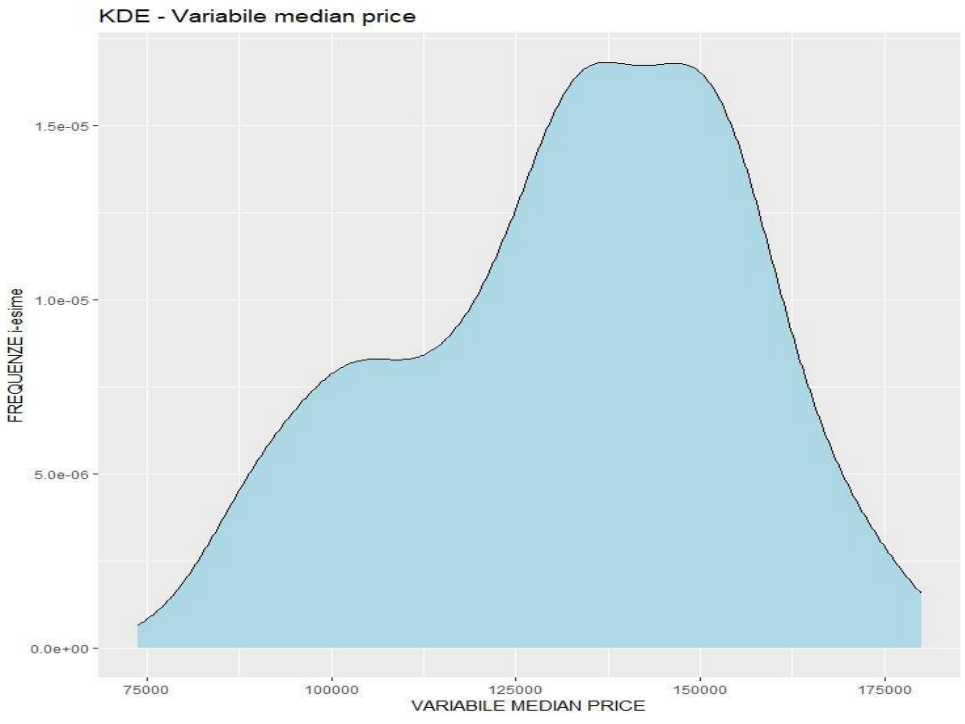
- **Sales** ○ Range =distribuzione relativamente ampia con un intervallo di 344 vendite,
  - misura variabilità CV%= la dev.std è il 42% rispetto alla propria media (variabilità abbastanza elevata)
  - Skewness **=** 0.72, asimmetria positiva, con la coda della distribuzione allungata verso i valori più elevati della distribuzione
  - Curtosi = -0.31 (curva platicurtica, forma leggermente forma schiacciata)



- **Volume**
  - Range**:** distribuzione con intervallo di 75.381M$;
  - misura variabilità CV% = la dev.std è il 53.71% rispetto alla propria media (variabilià abbastanza elevata)
  - skwewness: 0.88, forte asimmetria positiva, con la coda della distribuzione allungata verso i valori più elevati della distribuzione ○ curtosi**:** 0.18 (curva leptocurtica, forma leggermente appuntita)

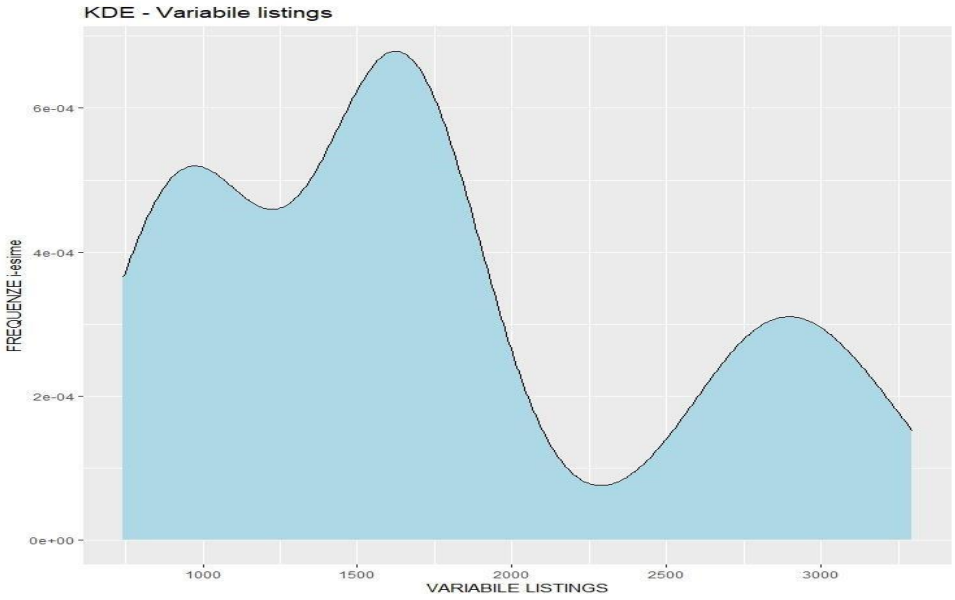


- **Median_price**:
  - range= ampio intervallo di distribuzione pari a 106200$ ○ misura variabilità CV% = la dev.std è 17.08% rispetto alla media indicando una variabilità contenuta
  - skwewness: -0.36, indicando asimmetria negativa con curva allungata verso i valori più bassi della distribuzione
  - curtosi: -0.62 (curva platicurtica, forma tendenzialmente schiacciata)

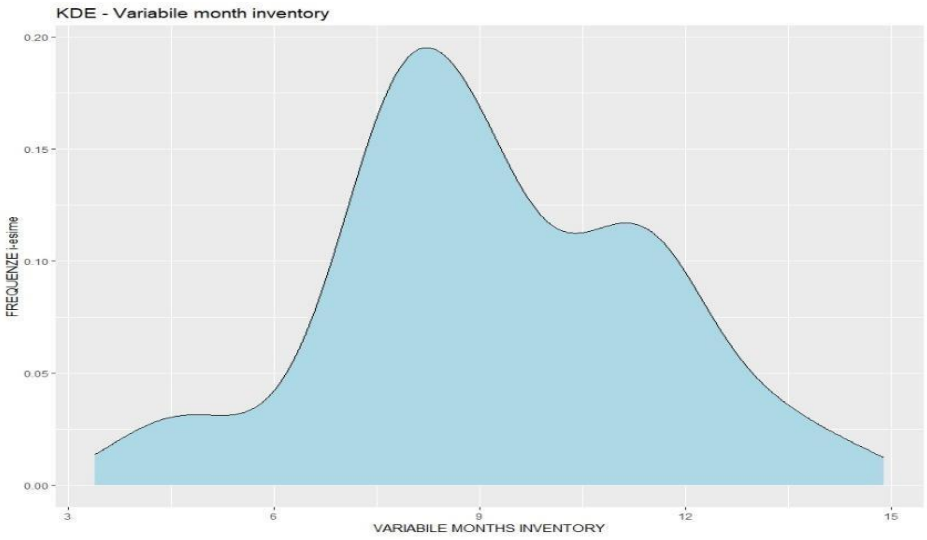


- **Listings**:

- range= ampio intervallo di distribuzione pari a 2553 ○ misura variabilità CV% = la dev.std è 43.31% rispetto alla media indicando una variabilità medio elevata
- skwewness: 0.65, indicando asimmetria positiva con curva allungata verso i valori più maggiori della distribuzione ○ curtosi: -0.79 (curva platicurtica, forma tendenzialmente schiacciata)



- **month_inventory**:
  - range= intervallo di distribuzione pari a 11.50 ○ misura variabilità CV% = la dev.std è 25.06% rispetto alla media indicato una variabilità contenuta
  - skwewness: 0.06, indicando una curva verosimilmente simmetrica ○ curtosi: -0.17 (curva platicurtica, forma leggermente schiacciata)

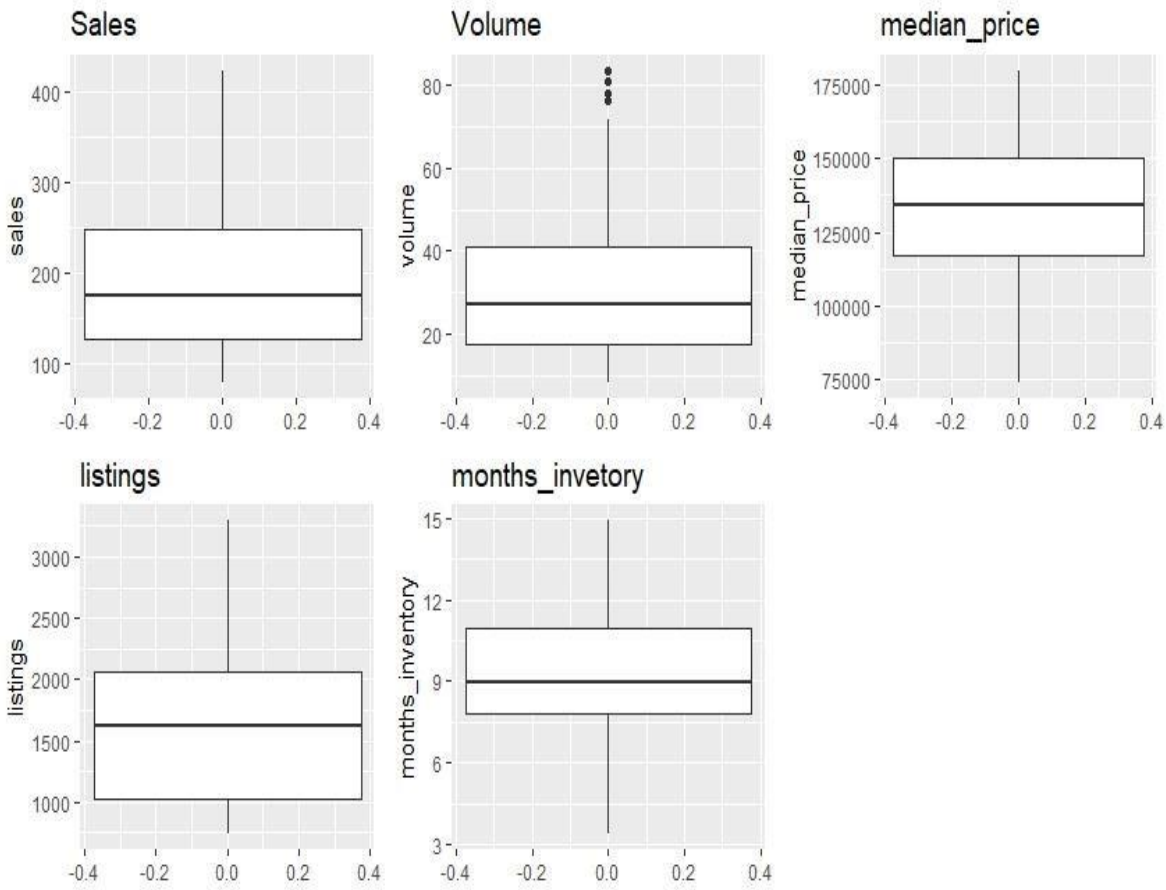


Graficamente è possibile utilizzare la versione affiancata dei boxplots per avere un'idea grafica della presenza di **outliers** nel dataset, tramite un'unica schermata di output di ggplot con la funzione **grid.arrange()** della libreria **"gridExtra":**

```r
# Creo i boxplot separati per ogni variabile
city_box<- ggplot(data_texas_real_estate, aes(y = city)) + geom_boxplot() + labs(title = "city")
year_box<- ggplot(data_texas_real_estate, aes(y = year)) + geom_boxplot() + labs(title = "Year")
month_box<- ggplot(data_texas_real_estate, aes(y = month)) + geom_boxplot() + labs(title = "month")
sales_box<- ggplot(data_texas_real_estate, aes(y = sales)) + geom_boxplot() + labs(title = "Sales")
volume_box<- ggplot(data_texas_real_estate, aes(y = volume)) + geom_boxplot() + labs(title = "Volume")
medianprice_box<- ggplot(data_texas_real_estate, aes(y = median_price)) + geom_boxplot() + labs(title = "median_price")
listings_box<- ggplot(data_texas_real_estate, aes(y = listings)) + geom_boxplot() + labs(title = "listings")
months_inv_box<- ggplot(data_texas_real_estate, aes(y = months_inventory)) + geom_boxplot() + labs(title = "months_invetory")

#grid.arrange per visualizzarli insieme
library(gridExtra)
grid.arrange(city_box,year_box,
             month_box,
             sales_box,
             volume_box,
             medianprice_box,
             listings_box,
             months_inv_box,
             ncol = 3)
```

I seguenti boxplots rappresentano l'output grafico di R, tramite l'utilizzo combinato delle librerie gglot2-gridExtra:



Si nota che la **variabile volume presenta outliers,** relativi a dei suoi valori che risultano essere maggiori all'estremo estremo superiore dell'IQR della distribuzione, motivo per cui ho proceduto a identificare gli outliers corrispondenti tramite libreria **dyplr** e utilizzo del **filter:**

```
#indentificare outliers in Volume:
Q1 <- quantile(volume, 0.25)
Q3 <- quantile(volume, 0.75)
IQR <- Q3 - Q1
lower_bound <- Q1 - 1.5 * IQR
upper_bound <- Q3 + 1.5 * IQR
outliers <- data_texas_real_estate%>%
  filter(volume < lower_bound | volume > upper_bound)
outliers
```

278:9 (Untitled)

Console Background Jobs

R 4.4.0 · C:/Users/luca.marletta/Desktop/Corso di statistica descrittiva R_16_9_2024/18. Analisi del mercato immobiliare del Texas/Progetto/

```
> outliers
                  city year month sales volume median_price listings months_inventory
1 Bryan-College Station 2013     7   402 76.116       161000     1385              6.1
2 Bryan-College Station 2014     6   377 77.983       169600     1152              4.5
3 Bryan-College Station 2014     7   403 83.547       172600     1041              4.1
4                Tyler 2014     6   423 80.814       155700     2855              9.3
```

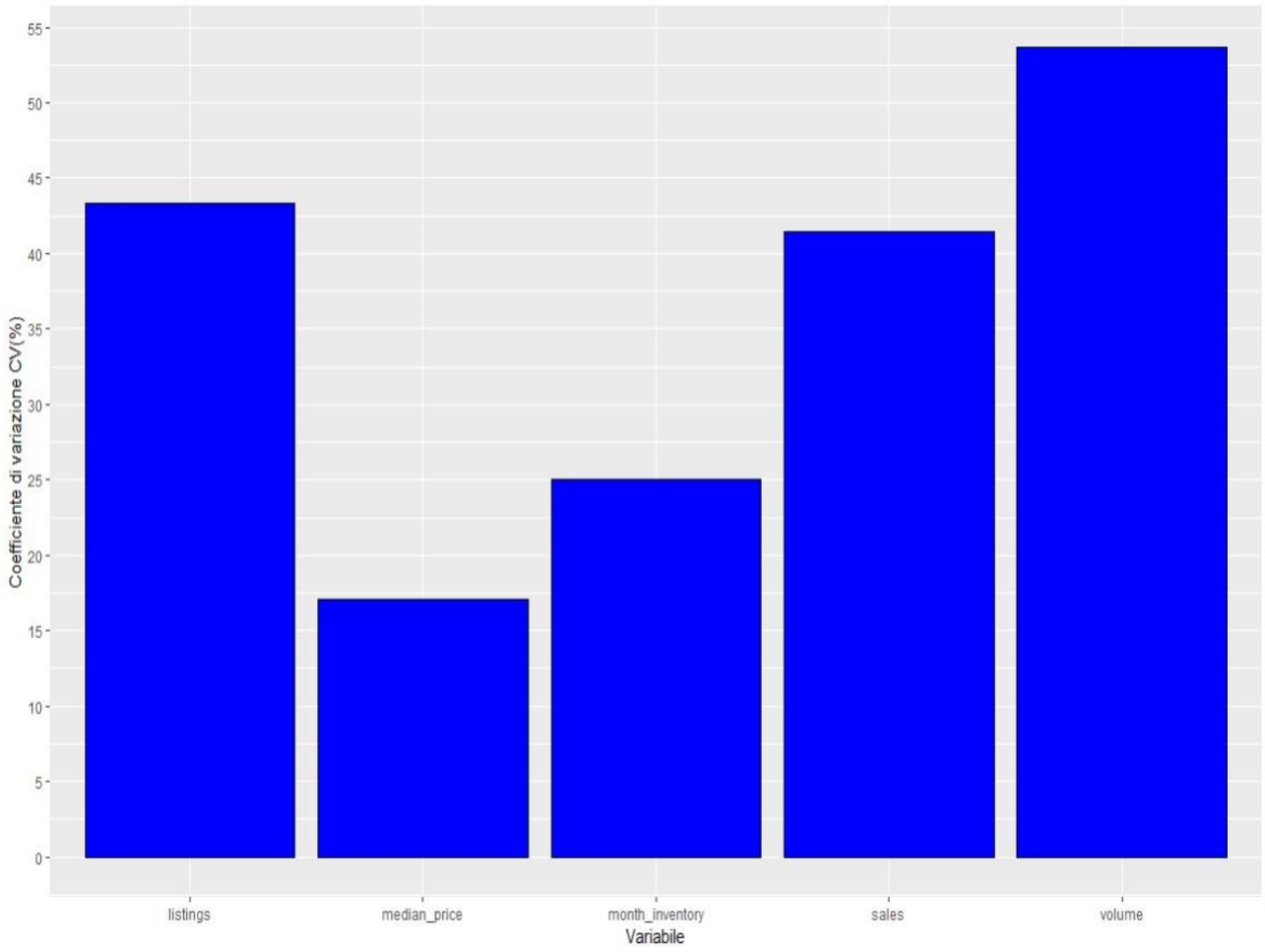
3. Qual è la variabile con variabilità più elevata? Come ci sei arrivato? E quale quella più asimmetrica?

3.1 VARIABILE DEL DATASET CON LA VARIABILITÀ PIÙ ELEVATA:

Per l'indagine sulla variabilità più elevata**, è necessaria confrontare la deviazione standard che presenta la variabile nel dataset rispetto alla propria media, ed esprimere il risultato il % (Coefficiente di variazione).** Sulla base dei valori ottenuti, la variabile che presenta il maggior valore del coefficiente di variazione %, sarà la variabile del dataset con la variabilità più elevata.

Grazie al dataframe **"variability_shape_dataframe**" e alle librerie **ggplot** è possibile visualizzare la variabile del dataset che presenta la variabilità più elevata osservando l'altezza delle colonne che rappresentano il CV% di ogni singola variabile messa in comparazione. Segue codice R, ed output delle librerie grafiche di ggplot2:

```
ggplot(data=variability_shape_dataframe)+
  geom_col(aes(x=Variabile,y=CV),fill="blue", col="black")+
  scale_y_continuous(breaks =seq(0,60,5))+
  labs(x="", y="Coefficiente di variazione CV(%)")
```



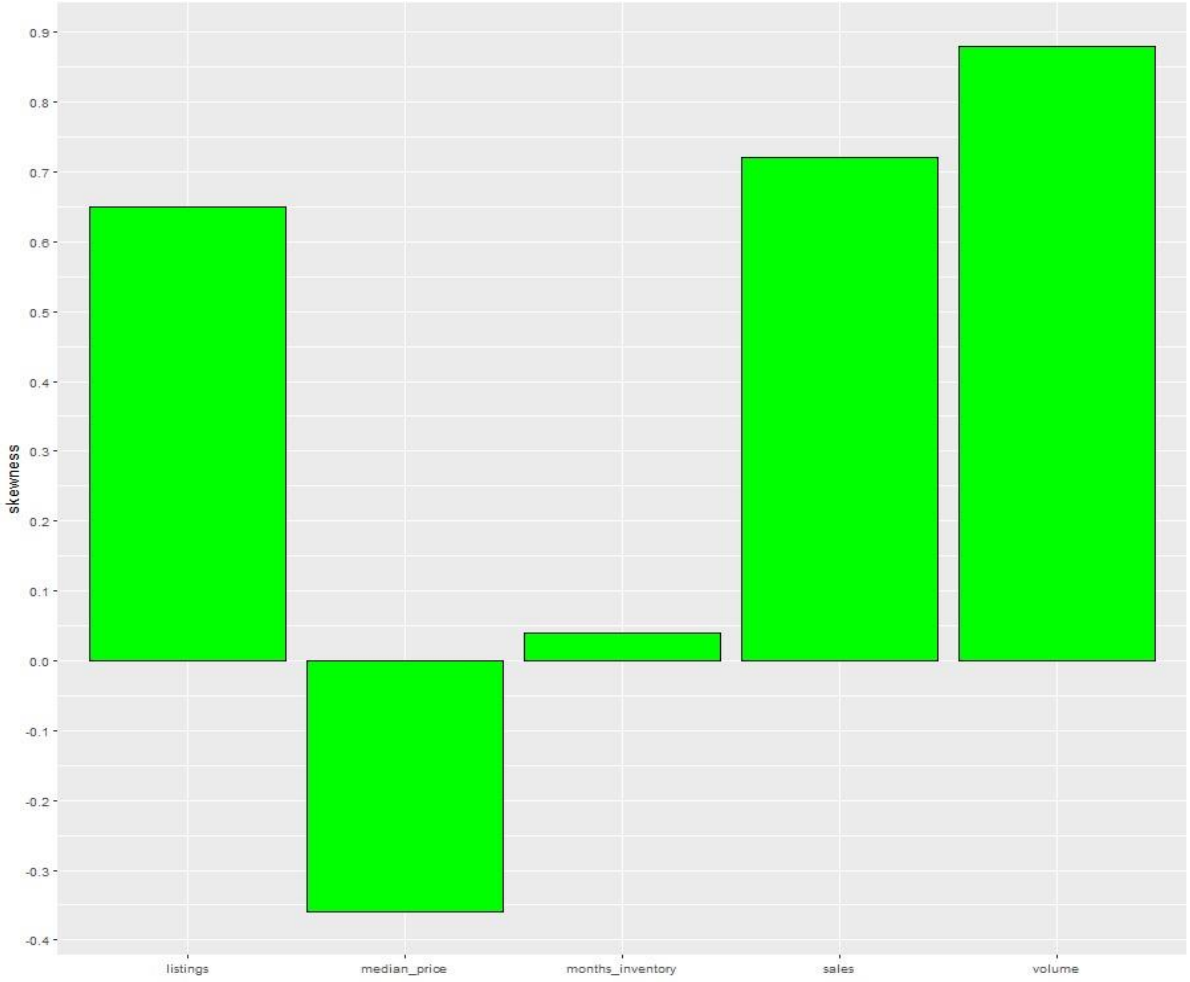
RISULTATO: La colonna blu più alta rappresenta la variabilità della proprietà più **elevata**, che nel dataset di partenza corrisponde alla variabile **volume**:

1. **CV(volume)=53.71%**
2. CV(listings)=43.31%
3. CV(sales)=41.42%
4. CV(month_inventory)=25.06%
5. CV(median_price)=17.06%

3.2 VARIABILE DEL DATASET CON LA PIÙ ELEVATA ASIMMETRIA:

anche in questo caso ho utilizzato il dataframe "variability_shape_dataframe" e la libreria ggplot2 per indicare la variabile più asimmetria, che in questo caso è rappresentata dalla variabile con la skewness maggiore con il più alto valore positivo (asimmetria positiva) , o e con la skewness minore con il piu basso valore negativo (asimmetria negativa) . In termini grafici l'asimmetria della variabile è rappresentata dall'altezza delle barre verdi (in senso positivo o negativo). Quanto è più lunga la barra, quanta maggiore è l'asimmetria. Segue codice R, e output di ggplot2:

```
ggplot(data=variability_shape_dataframe)+
  geom_col(aes(x=Variabile,y=skewness),fill="green", col="black")+
  scale_y_continuous(breaks =seq(-1,1,0.1))+
  labs(x="", y="skewness")
```



RISULTATO: La colonna verde più "lunga" rappresenta la **variabile più asimmetrica**, che in questo dataset è rappresentata ancora una volta dalla variabile volume, con **skewness pari a +0.88 (forte asimmetria positiva)**

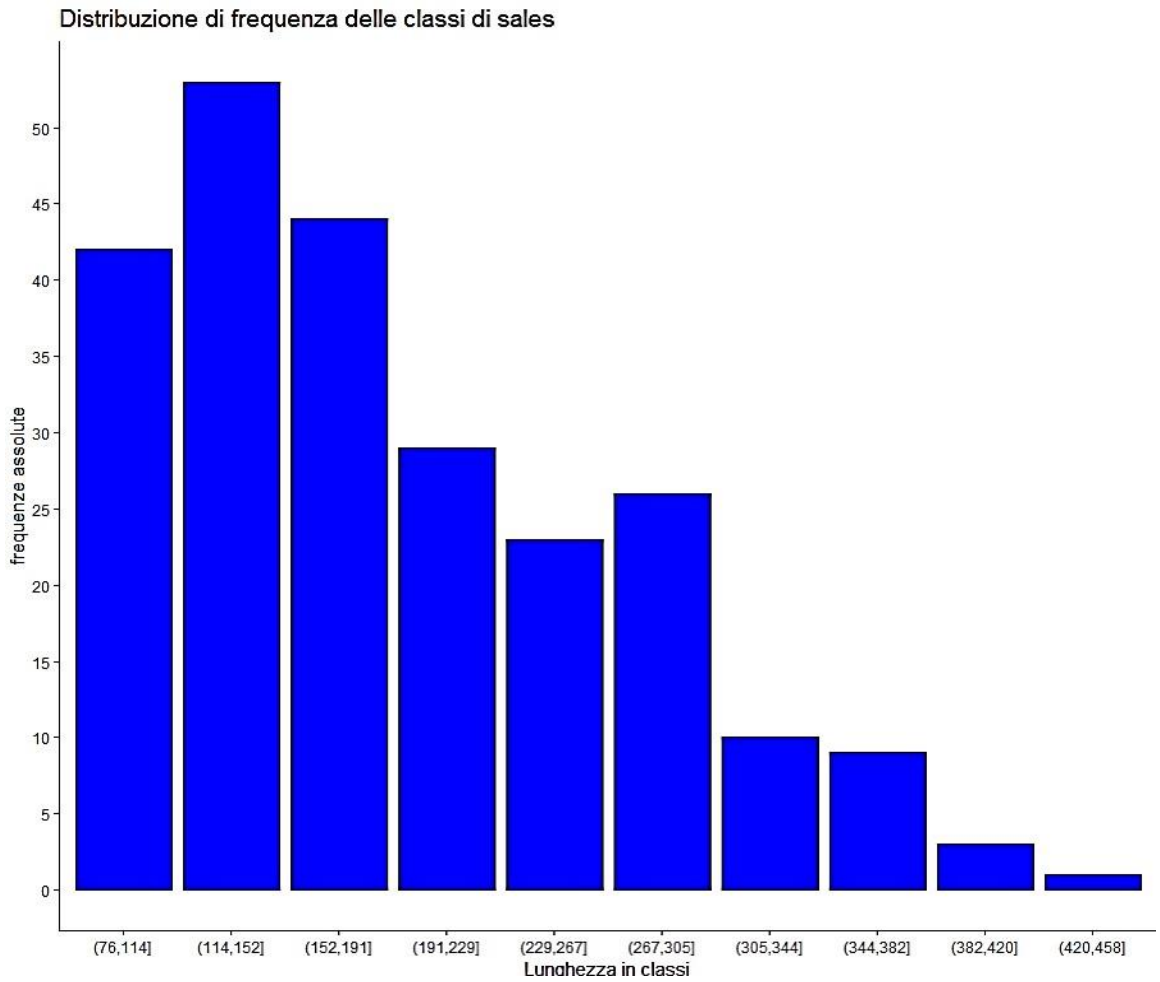
4. **Dividi una delle variabili quantitative in classi, scegli tu quale e come, costruisci la distribuzione di frequenze, il grafico a barre corrispondente e infine calcola l'indice di Gini.**

- Suddivido la variabile **sales** in "**classes**" di ampiezza **bins_amp**
- Creo un dataframe **distr_freq** con le distribuzioni di frequenza
- Creo il grafico a barre con ggplot

```
############################################################
#INDICE DI ETEROGENEITA' per una variabile quantitativa (prendo sales):
#Distribuzione di Frequenza Variabile median_price
#a divisione in classi
Nbins<-round((1+log2(n)))
Nbins
bins_amp<-((((max(sales))-min(sales))/Nbins))
bins_amp
classes=cut(sales,(seq((min(sales)-3),(max(sales)+bins_amp),bins_amp)))
#distribuzione di frequenza
ni <- table(classes)
fi <- ni/n
Ni <- cumsum(ni)
Fi <- cumsum(ni)/n
distr_freq<- as.data.frame(cbind(ni,fi,Ni,Fi))
round(distr_freq,2)
#grafico a barre
ggplot()+
  geom_bar(aes(x=classes),col="black", fill="blue")+
  labs(title="Distribuzione di frequenza delle classi di sales",
       x="Lunghezza in classi",
       y="frequenze assolute")+
  scale_y_continuous(breaks = seq(0,50,5))+
  theme_classic()
```

Segue la l'output della console R del dataframe delle distribuzione di frequenza distr_freq e il plot grafico a barre relative allo stesso dataset:

	ni	fi	Ni	Fi
(76,114]	42	0.17	42	0.17
(114,152]	53	0.22	95	0.40
(152,191]	44	0.18	139	0.58
(191,229]	29	0.12	168	0.70
(229,267]	23	0.10	191	0.80
(267,305]	26	0.11	217	0.90
(305,344]	10	0.04	227	0.95
(344,382]	9	0.04	236	0.98
(382,420]	3	0.01	239	1.00
(420,458]	1	0.00	240	1.00



COMMENTO: ho suddiviso il vettore della variabile sales in 10 classi di ampiezza 38 vendite, e si nota subito dal grafico che le classi con un numero di vendite inferiori sono le più frequenti, per poi diminuire in frequenza assolute man mano che ci spingiamo verso destra fra le classi con un numero di vendite superiori. Dalla tabella dataframe distr_freq appena illustrata, notiamo che le prime 7 classi che vanno da 76 a 344 vendite restituiscono il 95% percento della frequenza cumulata, con la classe modale identificata in (114,152] vendite con una frequenza assoluta di 53 vendite (22% della frequenza relativa di classe). La presenza di classi con vendite > uguali a 267 vendite, e una classe modale molto inferiore, fa si che la forma della distribuzione sia asimmetria positiva presentando lunghe code alla destra della classe modale (verso le classi a più alto valore di vendite). Questo fatto era già stato confermato dagli studi precedenti delle skewness delle variabili continue. Infatti la skewness per sales presentava +0.72. Infine, ho calcolato **l'indice di Eterogeneità di Gini** per la variabile sales, che può assumere valori da 0 a 1.L'indice di eterogeneità di Gini misura la propensione di una variabile qualitativa (o quantitativa numerica divisa in classi di frequenza) ad assumere le sue diverse modalità, andando quindi a considerare la distribuzione di frequenze.
Valori alti dell'indice indicano una distribuzione abbastanza omogenea, con il valore 1 che corrisponde alla pura equidistribuzione (massima eterogeneità). Valori bassi dell'indice, invece, indicano una distribuzione più diseguale, con il valore 0 che corrisponde alla massima concentrazione (minima eterogeneità). Per il calcolo della dell'indice Gini ho creato una funzione chiamata **gini.index, e lanciato tale funzione sulle classi del dataframe "distr_freq" ("classes" in R) costruite in precedenza sulla variabile sales.**

Segue il codice e il commento per la variabile sales e relativi risultati:

```
217 cat("L'indice di gini per la distribuzione di freqenza in classi di sales è pari a: ", round(gini.index(classes),2))
218
219
217:117 (Untitled)
Console Terminal Background Jobs
R 4.4.1 · C:/Users/lucam/Desktop/
> cat("L'indice di gini per la distribuzione di freqenza in classi di sales è pari a: ", round(gini.index(classes),2))
L'indice di gini per la distribuzione di freqenza in classi di sales è pari a:  0.94
>
```

5. **"INDOVINA" L'INDICE DI GINI PER LA VARIABILE CITY.**

   Ricordando la funzione table() applicata al vettore della variabile city:

```
> #indovina l'indice di Gini per la variabile city
> table(city)
city
             Beaumont Bryan-College Station                 Tyler
                   60                   60                    60
        Wichita Falls
                   60
```

- Otteniamo che il vettore della variabile city acquisisce 4 modalità, **in modalità equidistribuita per ogni modalità (nome delle citta),** quindi abbiamo una massima eterogeneità (o equidistribuzione) della variabile city. **Ne consegue che il valore dell'indice di eterogeneità di Gini (G') è pari al suo massimo teorico, ovvero G'=1.**

6. **QUAL È LA PROBABILITÀ CHE PRESA UNA RIGA A CASO DI QUESTO DATASET ESSA RIPORTI LA CITTÀ "BEAUMONT"? E LA PROBABILITÀ CHE RIPORTI IL MESE DI LUGLIO? E LA PROBABILITÀ CHE RIPORTI IL MESE DI DICEMBRE 2012?**

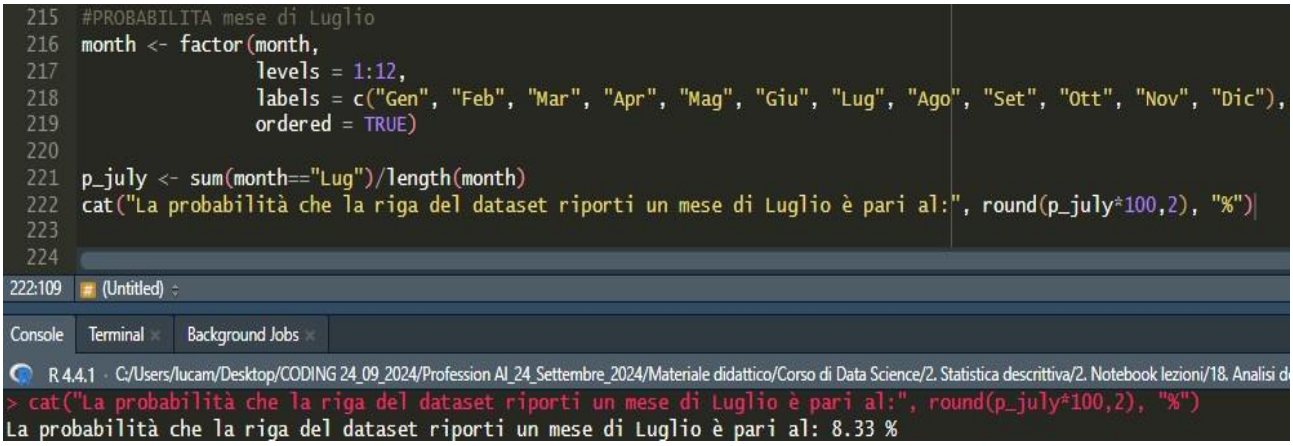
6.1 Probabilità Città Beumont: per calcolare la probabilità che presa una riga a caso del dataset (che ricordiano essere lungo 240 osservazioni per ogni variabile) si peschi la città di Beaumont, è bene ricordare la **definizione di probabilità classica**, ovvero: p(x) % = (numero caso favorevoli (che esca x) / numero di casi possibili) *100

quindi per ottenere il numero di volte che si ripete la città di Beaumont sul dataset e lungo la colonna del vettore city è data dalla funzione sum(city= "Beaumont") che conta il numero di casi favorevoli in modo cumulativo lungo la colonna del vettore. Se dividiamo la somma ottenuta per la lunghezza del vettore city, tramite lenght(city), otteniamo il numero di casi possibili (lunga la colonna del vettore). Moltiplicando x 100 otteniamo probabilità di Beumont in %. Segue codice R, ed output della console di Rstudio:

```
212 #PROBABILITA city BEAUMONT
213 p_beaumont <- sum(city=="Beaumont")/length(city)
214 cat("La probabilità che la riga del dataset riporti Beaumont è pari al:", p_beaumont*100, "%")
215
214:54 (Untitled)
Console Terminal Background Jobs
R 4.4.1 · C:/Users/lucam/Desktop/CODING 24_09_2024/Profession AI_24_Settembre_2024/Materiale didattico/Corso di Data Science/2. Statistica descrittiva/2. Noteb
> cat("La probabilità che la riga del dataset riporti beaumont è pari al:", p_beaumont*100, "%")
La probabilità che la riga del dataset riporti beaumont è pari al: 25 %
>
```

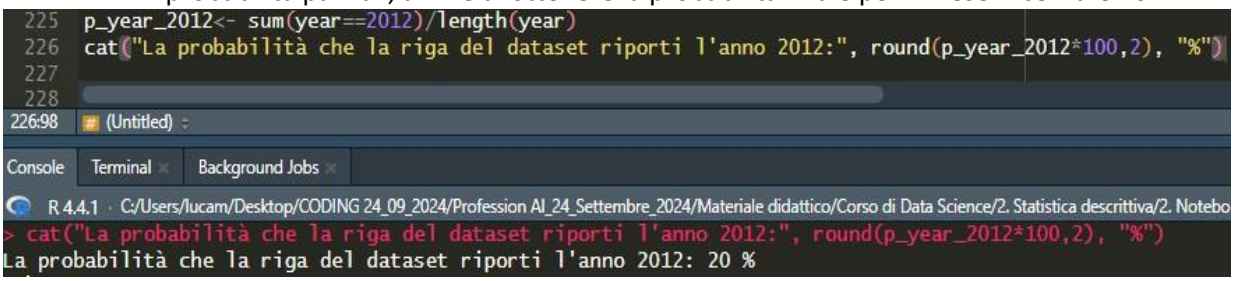
6.2 Probabilità mese di Luglio ("Lug"):

- stesso ragionamento si applica al caso della ricerca del mese di Luglio nel dataset. In questo caso la somma cumulativa è stata eseguita lunga la colonna del vettore month, ricercando come casi favorevoli la stringa "Lug", ovvero il mese di Luglio che inizialmente era il mese n.7, oppurtamente codificato in livelli tramite la funzione factor(). Segue la riga di codice e il print della consolle con il risultato della probabilità richiesta:

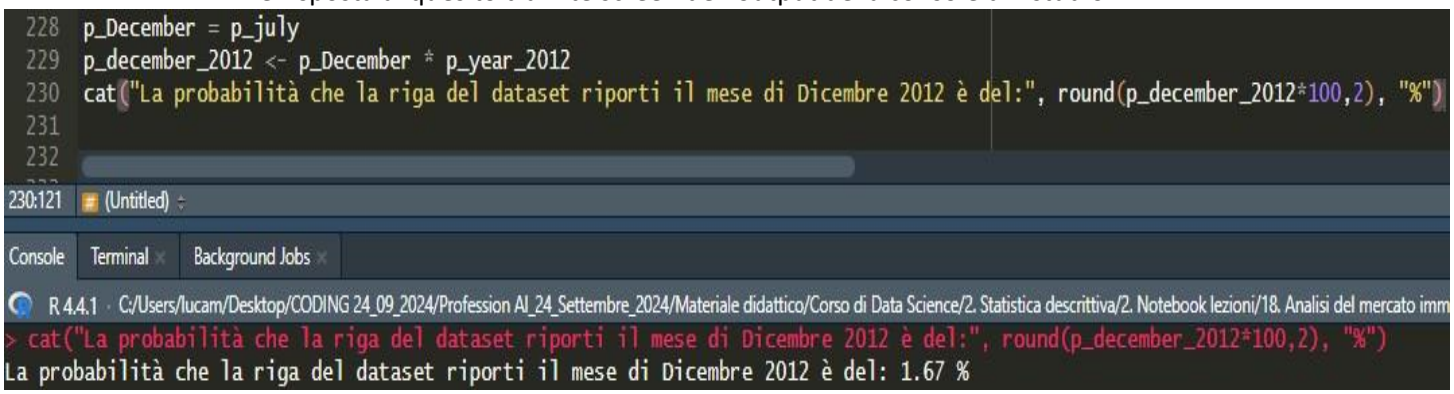


Probabilità mese di Dicembre 2012:

- Per calcolare la probabilità combinata di trovare Dicembre 2012 fra le righe del dataset, si rende necessario dapprima calcolare la probabilità di trovare un anno 2012 nel vettore year, poi quella di Dicembre nel vettore month, e poi combinare tramite il prodotto delle probabilità parziali, al fine di ottenere la probabilità finale per il mese Dicembre 2012:



Dato che la variabile month risulta essere equidistribuita lungo il dataset (informazione ottenuta dalla funzione table(month)) sappiamo, quindi, anche che la probabilità ottenuta dall'esercizio precedente (per il mese di luglio) sia uguale per tutti i mesi del dataset, quindi anche per il mese di dicembre generico. P (dicembre) = 8.33%. A questo punto combinando le probabilità parziali, otterremo la probabilità per il mese di Dicembre 2012. Segue codice R e risposta al quesito tramite screen dell'output della console di Rstudio:



7. **ESISTE UNA COLONNA COL PREZZO MEDIANO, CREANE UNA CHE INDICA INVECE IL PREZZO MEDIO, UTILIZZANDO LE ALTRE VARIABILI CHE HAI A DISPOSIZIONE**

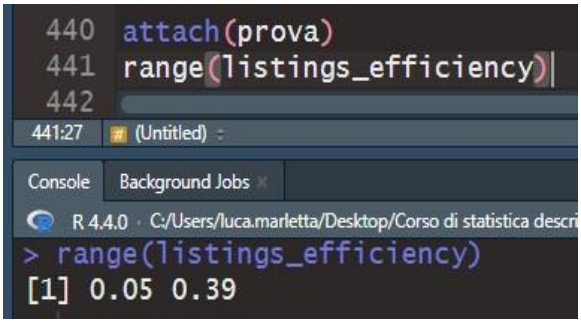
Al fine di creare una nuova colonna del prezzo medio ed inserirla nell'ultima colonna del dataset di partenza, ho proceduto ad effettuare il rapporto fra i vettori volume e sales del database di partenza, moltiplicando infine per 1'000'000 al fine di riportare i valori in tabella in kdollari. Quindi ho proceduto a creare il primo **dataframe "prezzo_medio_kdollari**", dal **rapporto** matematico dei vettori **volume** e **sales** del dataset iniziale, ed integrare il nuovo dataframe creato all'interno del dataset originale tramite la funzione cbind. La funzione cbind mi permette di affiancare alle colonne del dataset iniziale anche il mean_price creato in questa fase del progetto: Segue il codice e il particolare in tabella:

```
mean_price_USdollars<-(volume/sales)*1000000
prezzo_medio_kdollari <- data.frame(mean_price_USdollars)
data_texas_real_estate<-cbind(data_texas_real_estate,prezzo_medio_kdollari)
```

city	year	month	sales	volume	median_price	listings	months_inventory	mean_price_USdollars
Beaumont	2010	1	83	14.162	163800	1533	9.5	170626.51
Beaumont	2010	2	108	17.690	138200	1586	10.0	163796.30
Beaumont	2010	3	182	28.701	122400	1689	10.6	157697.80
Beaumont	2010	4	200	26.819	123200	1708	10.6	134095.00
Beaumont	2010	5	202	28.833	123100	1771	10.9	142737.62
Beaumont	2010	6	189	27.219	122800	1803	11.1	144015.87
Beaumont	2010	7	164	22.706	124300	1857	11.7	138451.22

8. **PROVA A CREARE UN'ALTRA COLONNA CHE DIA UN'IDEA DI "EFFICACIA" DEGLI ANNUNCI DI VENDITA. RIESCI A FARE QUALCHE CONSIDERAZIONE?**

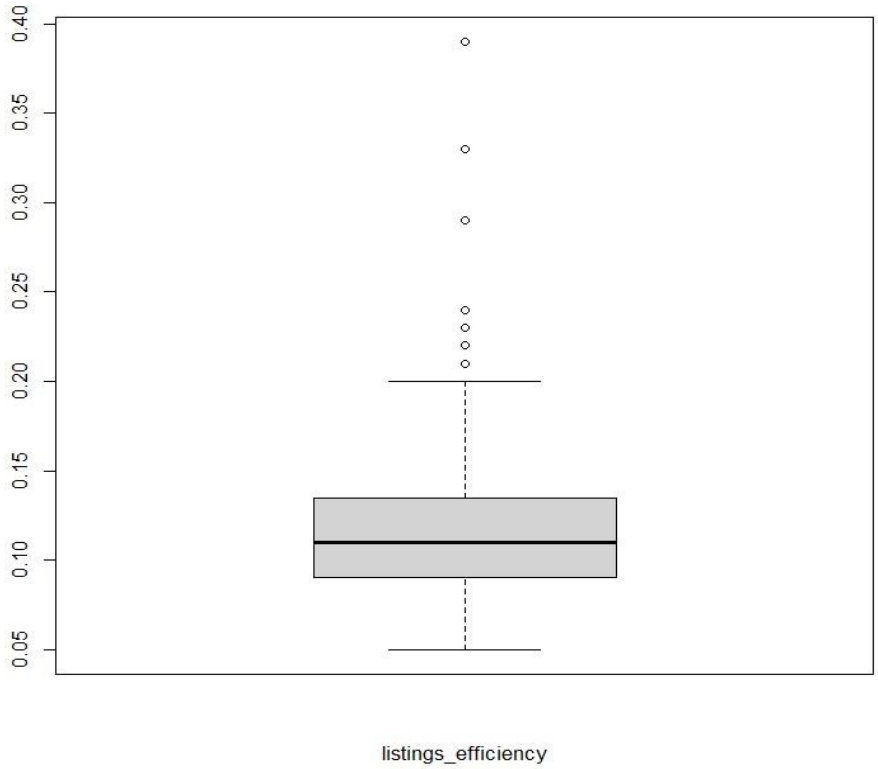
Sulla base delle metodiche operative espresse nel punto precedente del progetto ho provveduto a creare una nuova colonna vettore chiamata **"listings_efficiency"** data dal rapporto del numero di vendite (sales) rispetto al numero di annunci (listings) per le stesse osservazioni. Matematicamente **listings_efficiency= sales/listings.** Ho creato un nuovo dataframe chiamata df_efficiency, dove ho accorpato le colonne dei vettori city, year, sales, listings, listings_efficiency ed iniziato ad indagare le proprietà di questo nuovo dataframe. Lanciando la funzione range(listings_efficiency) è possibile osservare min e massimo della distribuzione appena creata



Osserviamo che :

- Annunci meno efficienti ❼ listings_efficiency=0.05 ❼ ci vogliono 20 annunci/vendita
- Annunci più efficienti ❼ listings_efficiency=0.39 ❼ ci vogliono 2.56 annunci/vendita

Visivamente la distribuzione dei valori dii listings_efficiency si distribuiscono secondo quanto illustrato nel relativo **boxplot(listings_efficiency):**

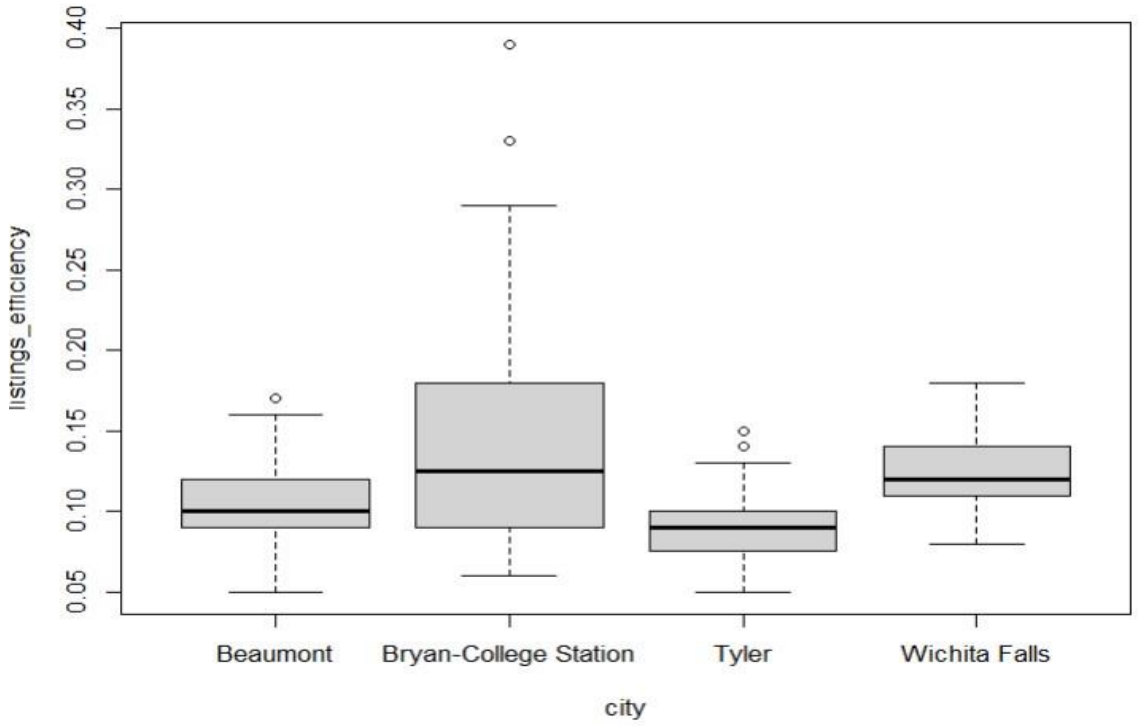


Si osservano outliers nella distribuzione, i quali si distinguono per migliore efficienza rispetto alla distribuzione. Possiamo osservare meglio i risultati ordinando il dataframe df_efficiency per listings_efficiency decrescente, e vedere dove la città dove si devono fare meno inserzioni pubblicitarie per realizzare una vendita corrisponde alla citta di Bryan-College Station:

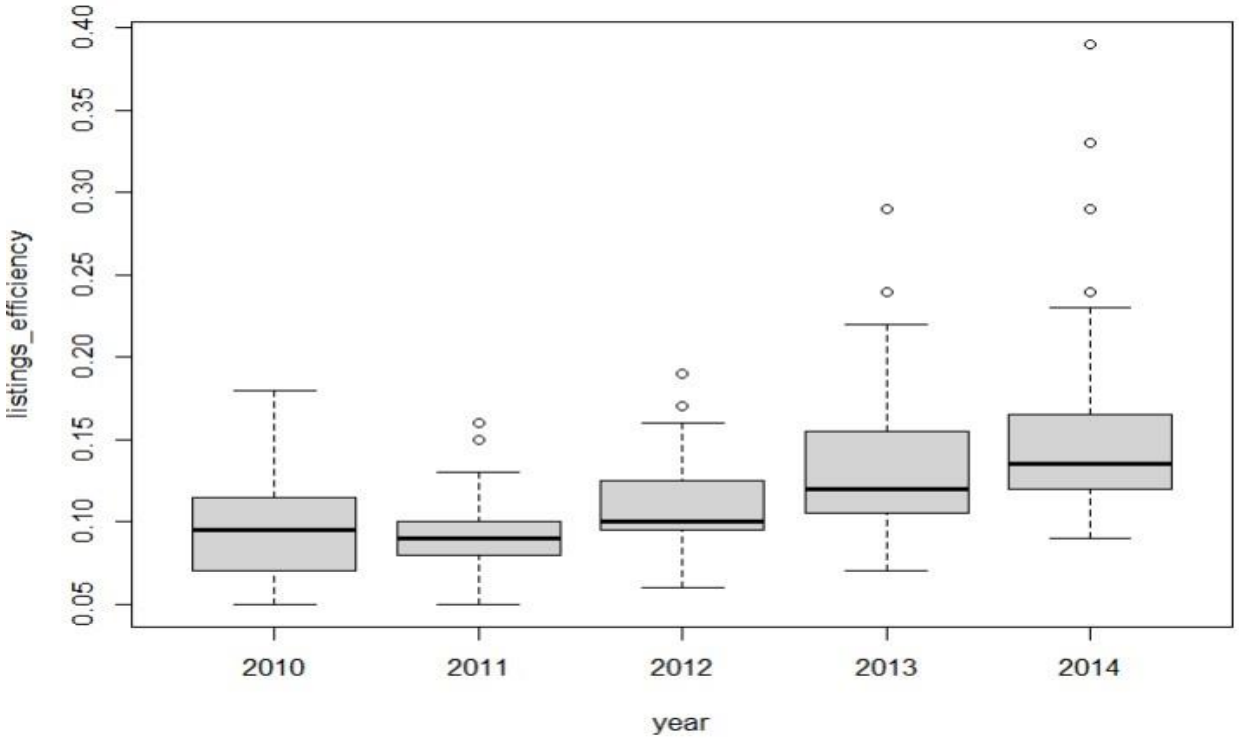
city	year	sales	listings	listings_efficiency
Bryan-College Station	2014	403	1041	0.39
Bryan-College Station	2014	377	1152	0.33
Bryan-College Station	2013	402	1385	0.29
Bryan-College Station	2014	353	1212	0.29
Bryan-College Station	2014	298	1016	0.29
Bryan-College Station	2013	357	1462	0.24
Bryan-College Station	2013	328	1385	0.24
Bryan-College Station	2014	303	1271	0.24
Bryan-College Station	2014	200	882	0.23
Bryan-College Station	2013	341	1581	0.22
Bryan-College Station	2014	275	1261	0.22
Bryan-College Station	2014	218	1031	0.21
Bryan-College Station	2014	204	1022	0.2

Altre informazioni interessanti è possibile ottenerle lanciando i boxplots della listings_efficiency condizionati alla città dove sono presenti le inserzioni pubblicitarie e si nota come prima che la città

più efficiente risulta essere Bryan-College Station, mentre quella meno performante da un punto di vista pubblicitario è quella di Tyler.



Continuando è possibile osservare come l'efficienza delle inserzioni pubblicitarie sia cresciuta mediamente negli anni dal 2010 al 2014, probabilmente grazio all'utilizzo di campagne mirate, strategie di marketing più efficienti e tecnologie più evolute:



9. PROVA A CREARE DEI SUMMARY(), O SEMPLICEMENTE MEDIA E DEVIAZIONE STANDARD, DI ALCUNE VARIABILI A TUA SCELTA, CONDIZIONATAMENTE ALLA CITTÀ, AGLI ANNI E AI MESI. PUOI UTILIZZARE IL LINGUAGGIO R DI BASE OPPURE ESSERE UN VERO PRO CON IL PACCHETTO DPLYR. TI LASCIO UN SUGGERIMENTO IN PSEUDOCODICE, OLTRE AL CHEATSHEET NEL MATERIALE

Il raggruppamento per città, anno e mese ci permette di osservare come cambiano nel tempo le vendite e i volumi in ciascuna città e mese specifico. Questo tipo di analisi è utile per identificare trend o variazioni stagionali. Nel particolare ho utilizzato la funzione pipe %>% della libreria dplyr(), unitamente a group_by() e summarise() per filtrare i dati in funzione delle city e degli anni di riferimento nei dataset. Sulla base di filtri andrò ad ottenere informazioni su come si sono spostate metriche di performance nel tempo, fra le quali:

- Prezzi mediani delle case ○ Media delle vendite ○ Dev. Standard delle vendite ○ Listings_efficiency (sales/listings)
- Months_inventory (quantità di tempo necessaria per vendere tutte le inserzioni correnti al ritmo attuale delle vendite, espresso in mesi.)

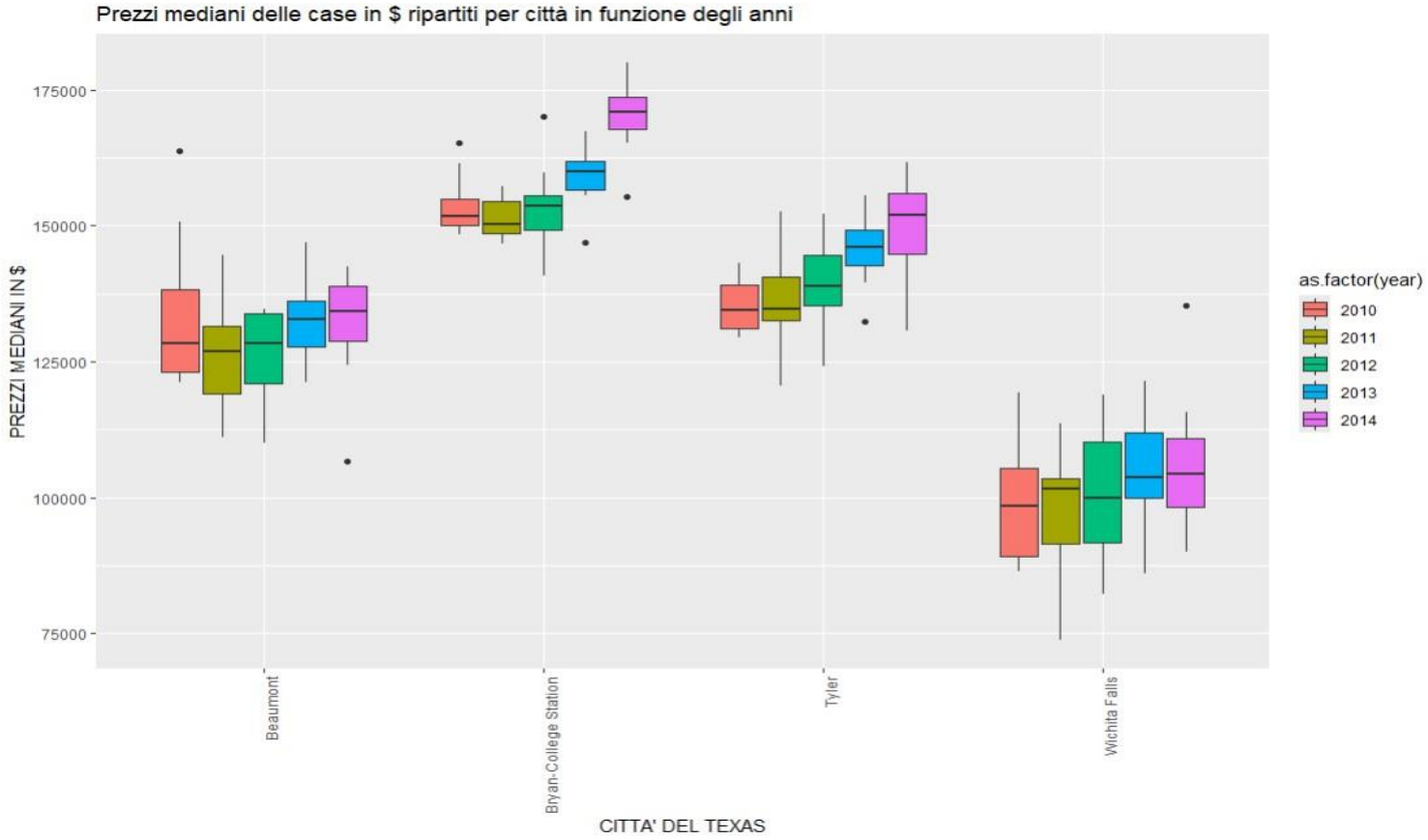
```
library(dplyr)
by_Year_City<-data_texas_real_estate %>%
  group_by(city,year)%>%
  reframe(media=mean(sales),
          median_price=mean(median_price),
            dev.st=sd(sales),
            listings_efficiency=mean(listings_efficiency),
            months_inventory=mean(months_inventory))

by_Year_City
```

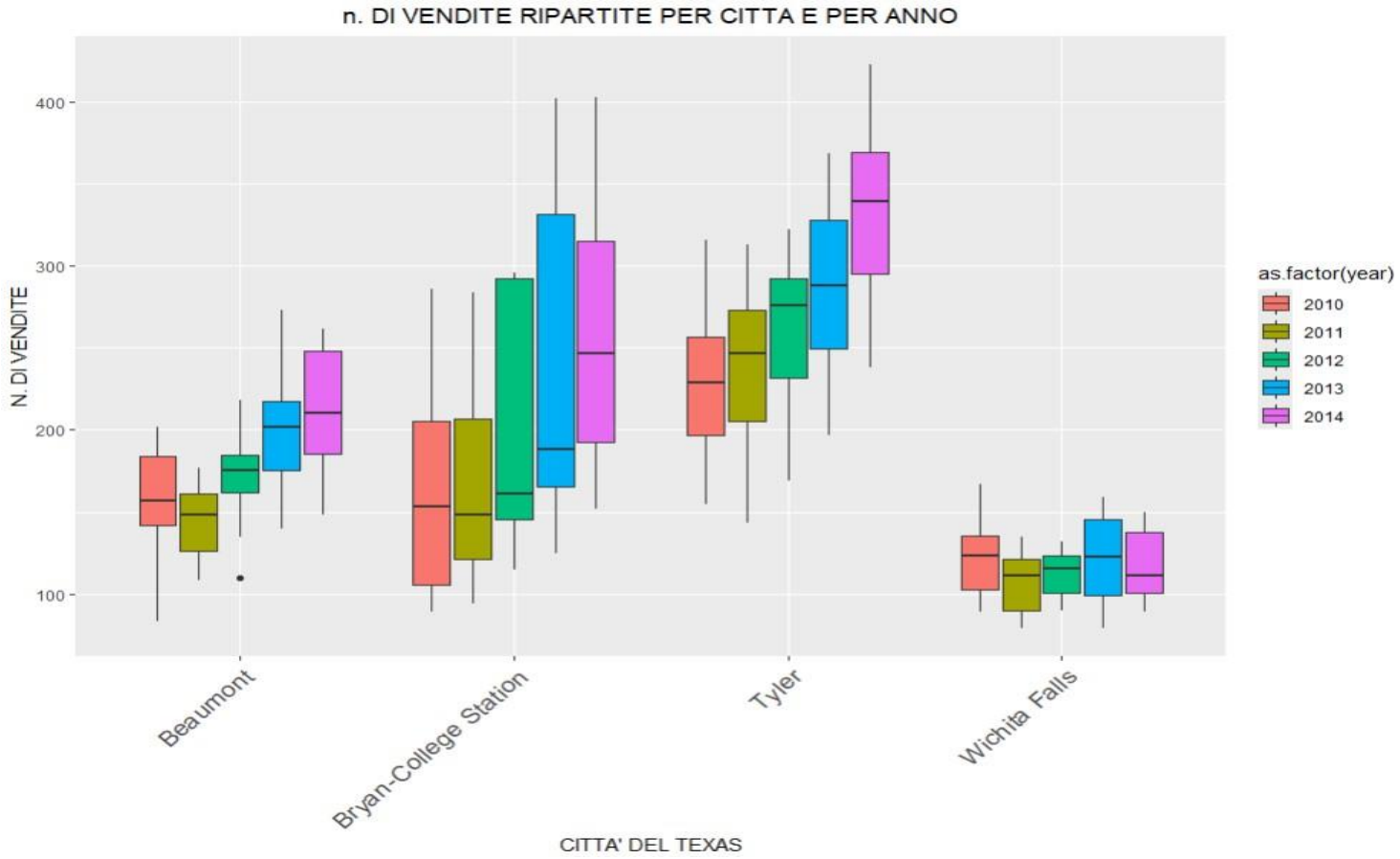
	city	year	median_price	media	dev.st	listings_efficiency	months_inventory
	<chr>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	Beaumont	2010	133117.	156.	36.9	0.09	10.9
2	Beaumont	2011	125642.	144	22.7	0.0808	11.7
3	Beaumont	2012	126533.	172.	28.4	0.102	10.8
4	Beaumont	2013	132400	201.	37.7	0.123	8.78
5	Beaumont	2014	132250	214.	36.5	0.134	7.64
6	Bryan-College Station	2010	153533.	168.	70.8	0.105	8.67
7	Bryan-College Station	2011	151417.	167.	62.2	0.102	9.8
8	Bryan-College Station	2012	153567.	197.	74.3	0.121	8.94
9	Bryan-College Station	2013	159392.	238.	95.8	0.171	6.5
10	Bryan-College Station	2014	169533.	260.	86.7	0.237	4.38
11	Tyler	2010	135175	228.	49.0	0.0742	12.6
12	Tyler	2011	136217.	239.	49.6	0.0767	13.5
13	Tyler	2012	139250	264.	46.4	0.09	11.6
14	Tyler	2013	146100	287.	53.0	0.102	10.2
15	Tyler	2014	150467.	332.	56.9	0.123	8.76
16	Wichita Falls	2010	98942.	123.	26.6	0.128	7.68
17	Wichita Falls	2011	98142.	106.	19.8	0.107	8.62
18	Wichita Falls	2012	100958.	112.	14.2	0.125	8.21
19	Wichita Falls	2013	105000	121.	26.0	0.143	7.13
20	Wichita Falls	2014	105675	117	21.1	0.133	7.44

a. **Insight su median_price** : E' possibile notare che il prezzo mediano delle case nel corso degli anni sia stato più alto nella città Bryan-college Station, seguito da Tyler, Beaumont e Wichita Falls, probabilmente indice del fatto che Bryan-College Station sia una regione con maggiore presenza di persone benestanti, mentre al contrario Wichita Falls sia il perfetto opposto o una zona in via di sviluppo.

Interessante il caso della citta di Tyler e Bryan-College Station , in cui il prezzo mediano della case dal 2010 al 2014 è salito di quasi 16.000$. Nei casi delle città di Beaumont e Whichita Falls, invece, la mediana dei prezzi anno per anno è rimasta suppergiu costante.



b. **Insight su sales**: per le città di Bryan-college Station, Tyler, Beaumont le vendita sono state in crescita dal 2010 al 2014, a differenza di Wichita Falls dove le vendite addirtura nel 2014 (117) sono diminuite rispetto al 2010 (123). La città che realizza più vendite mediamente negli anni è la città di Tyler, mentre ancora una volta al "fanalino di coda" troviamo la città di Wichita Falls che ha realizzato meno vendite nei periodi presi in analisi.



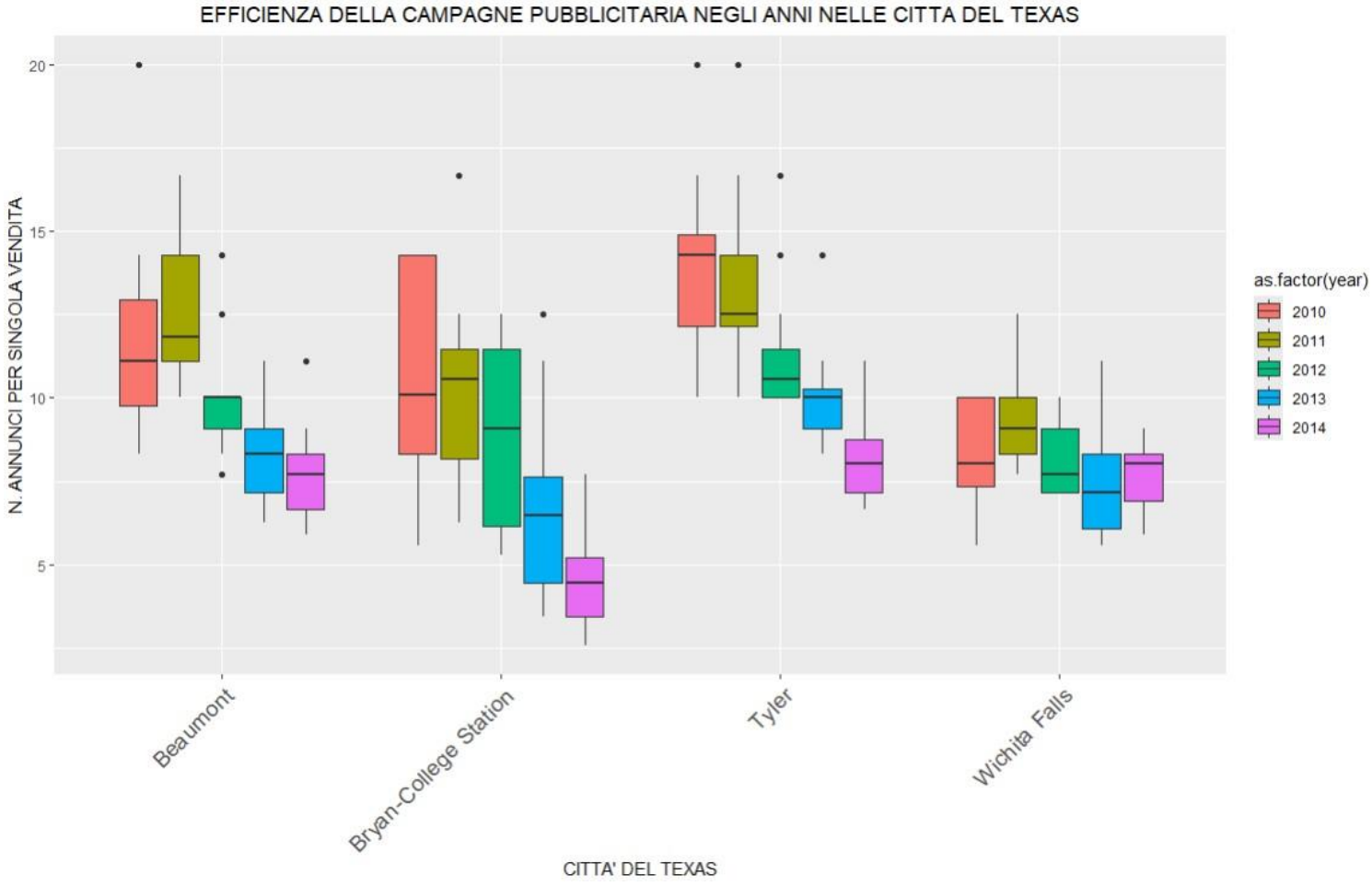
c. **Insight su listings_efficiency e month_inventory:**

- è interessante notare come quasi tutte le città tranne Tyler abbiano aumentato la loro efficiency in termini di vendite per singola inserzione; infatti si nota come dal 2010 al 2014 alcune citta abbiamo aumentato in media il valore di listings_efficiency, con ottimi risultati da parte di Bryan-College Station che nel 2014 ha registrato una media di listings_efficiency dell'0.234 (1 vendita ogni 4.27 inserzioni in media), mentre il month_inventory si sia ridotto solo a 4.34 mesi. Questi dati crescenti in termini di efficienza nel caso di Bryan-College Station dal 2010 al 2014 è probabile che siano legati ad un maggiore affinamento delle strategie di marketing utilizzate dalla real estate in questa località, unitamente al fatto che probabilmente la località si sia rivalutata per alcuni servizi (scuole, università prestigiose, centri di eccellenza, qualità della vita) e/o sia stata presa come meta di riferimento dai clienti come posto su cui investire dal punto di vista immobiliare.

  La città di Tyler , invece, pur realizzando le vendite assolute più alte in termini di numero negli anni, la real estate in questa città ha dovuto tenere alti gli investimenti in termini di inserzioni

pubblicitari. Infatti nel 2014 la listings_efficiency in Tyler era dello 0.123 (1 vendita ogni 8.13 inserzioni in media). Paragonando per efficienza delle vendite di Bryan-College Station con la citta di Tyler, la real estate performa quasi il doppio in termini di vendite a parità di inserzioni pubblicitari in Bryan-College Station, piuttosto che in Tyler deve investire molto di più in pubblicità.

Il seguente grafico prova a mostrare l'efficienza delle campagne pubblicitarie nello stato del texas, nei vari anni e per singola città in cui ha operato la real estate. Nell'asse delle ordinate troviamo l'inverso della listings efficiency (1/listings_efficiency), e per ogni distribuzione considerata, vuole rappresentare il numero di annunci necessari per realizzare un singola vendita. Più basso è il suo valore, e maggiore è l'efficienza delle campagne pubblicitarie della real estate per quell'anno di riferimento e per quella città:



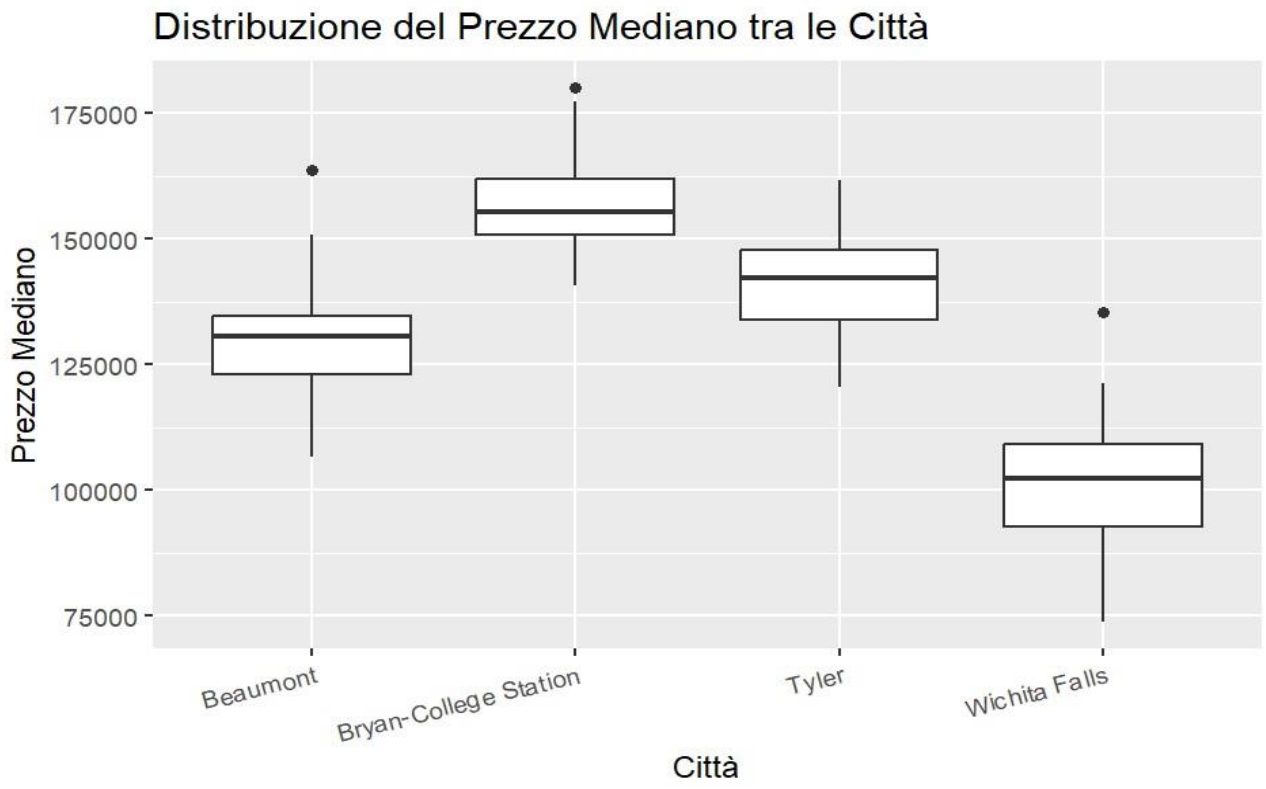
d. **Ulteriori insight** sono stati ottenuti prendendo in considerazione le media delle metriche di performance (median_price, sales,etc..) totalizzate nel totale del periodo considerato (2010-2014),in tutte le località **e filtrate su base mensile da gennaio a dicembre (da mese 1 a mese 12).**

	month	median_price	vendite	listings_efficiency	months_inventory
	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1	1	124250	127.	0.082	8.84
2	2	130075	141.	0.0875	9.06
3	3	127415	189.	0.116	9.40
4	4	131490	212.	0.126	9.72
5	5	134485	239.	0.141	9.68
6	6	137620	244.	0.143	9.70
7	7	134750	236.	0.143	9.62
8	8	136675	231.	0.140	9.39
9	9	134040	182.	0.112	9.18
10	10	133480	180.	0.112	8.94
11	11	134305	157.	0.103	8.66
12	12	133400	169.	0.116	8.12

- Dalla tabella riassuntiva delle metriche si può osservare che, **i mesi più proficui** per la realestate sono i mesi che vanno da **aprile (mese 4) ad agosto (mese 8)** nelle quali si ha un massimo di vendite di 244 a giugno (mese 6) con un efficienza dello 0.143 (1 vendita ogni 7 annunci circa). I mesi dove la real estate in **media riceve meno incassi corrispondono al mese di gennaio** dove la mediana dei prezzi di vendita è la più bassa a livello mensile (124250$ in media), le vendite sono al minimo (127 in media) e l'efficienza del listings di 0.082 (1 vendita ogni 12 annunci circa). Nel mese di Gennaio quindi la real esate otterrà profitti netti più bassi in quanto dovrà investire molto di più in inserzioni pubblicitarie per poter vendere, e abbassare il costo mediano delle case per essere "catching" con i clienti e realizzare vendite.

9.1 UTILIZZA I BOXPLOT PER CONFRONTARE LA DISTRIBUZIONE DEL PREZZO MEDIANO DELLE CASE TRA LE VARIE CITTÀ. COMMENTA IL RISULTATO.

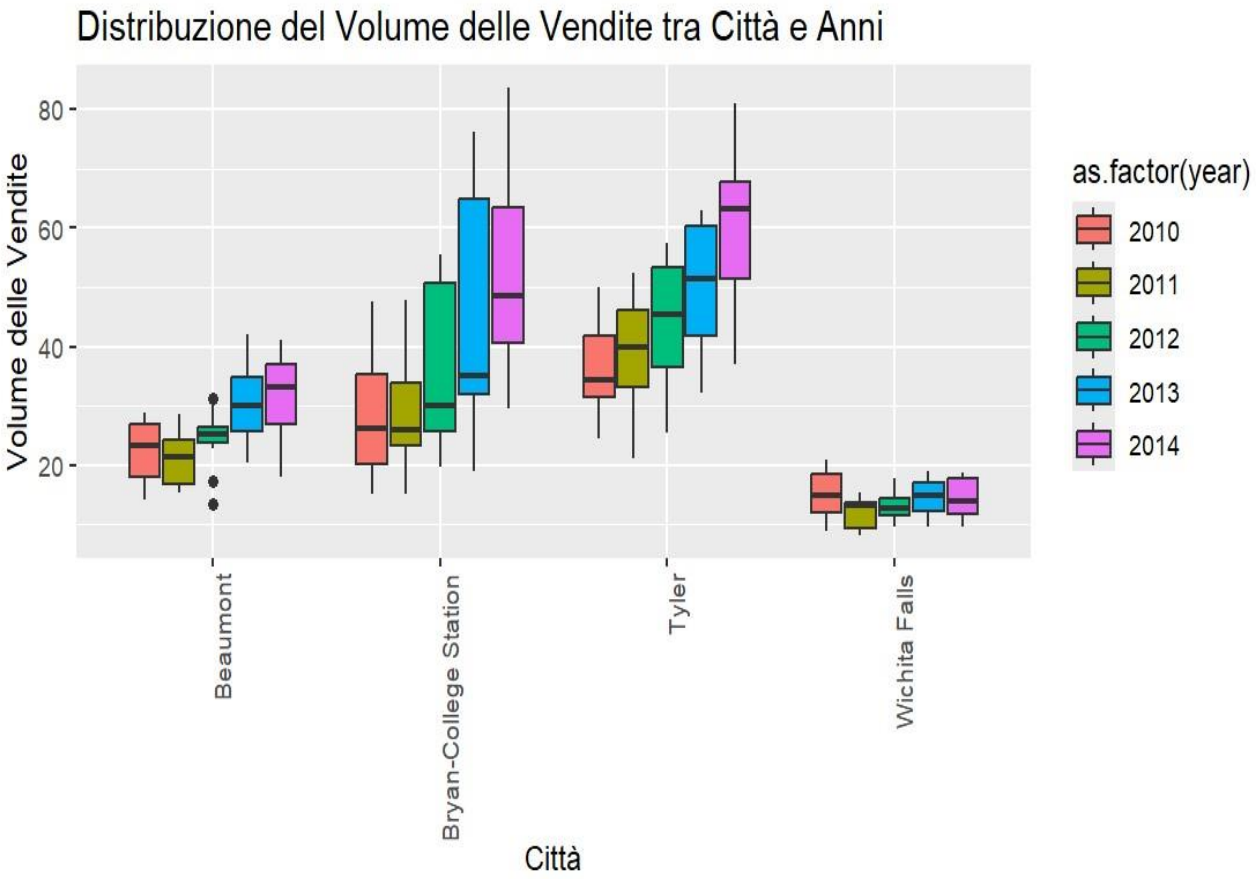
```r
#boxplot per confrontare la distribuzione del prezzo mediano delle case tra le varie città.
ggplot(data_texas_real_estate, aes(x = city, y = median_price)) +
  geom_boxplot() +
  theme(axis.text.x = element_text(angle = 15, hjust = 1)) +
  labs(title = "Distribuzione del Prezzo Mediano tra le Città", x = "Città", y = "Prezzo Mediano")
```



Commento sul grafico: E' possibile notare che le distribuzione dei prezzi mediani delle città variano notevolmente fra le città, evidenziando come città come Bryan-College Station risultino essere più care dal punto di vista immobiliare, mentre Wichita Falls sembra esssere più accessibile stando ai prezzi mediani applicati dalla real estate nelle città del Texas. Nella "via di mezzo" troviamo le città di Tyler e Beaumont.

9.2 UTILIZZA I BOXPLOT O QUALCHE VARIANTE PER CONFRONTARE LA DISTRIBUZIONE DEL VALORE TOTALE DELLE VENDITE TRA LE VARIE CITTÀ MA ANCHE TRA I VARI ANNI. QUALCHE CONSIDERAZIONE DA FARE?

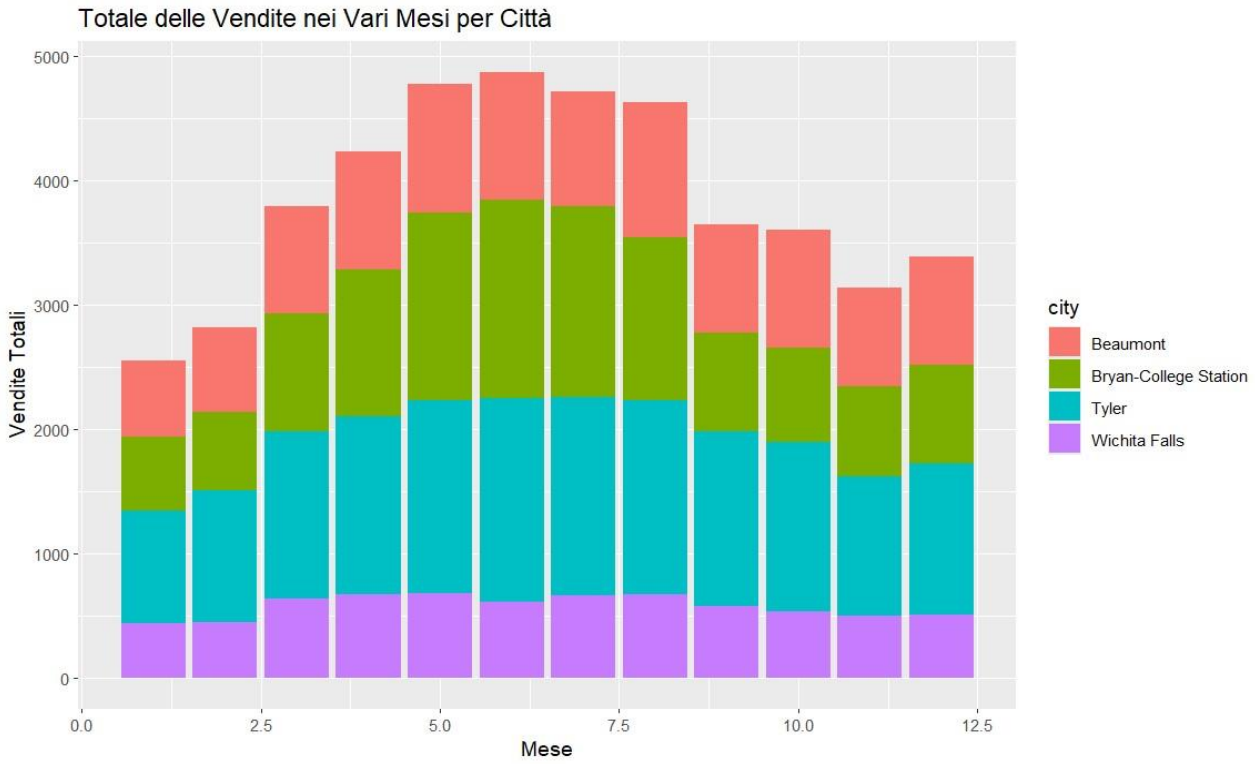
```
# Boxplot per il volume delle vendite per città e anno
ggplot(data_texas_real_estate, aes(x = city, y = volume, fill = as.factor(year))) +
  geom_boxplot() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  labs(title = "Distribuzione del Volume delle Vendite tra Città e Anni", x = "Città", y = "Volume delle Vendite")
```



Commento sul grafico: E' possibile osservare come l'anno considerato per le città sia una discriminante importante, fatta eccezione per la città di Wichita Falls dove il parametro anno non ha influenza sulla vendita. Si può notare che corso degli anni (dal 2010 al 2014) la real estate abbiamo realizzato vendite maggiori in tutte le città restati, ottenendo ottime performance in Bryan-College-Station e Tyler.

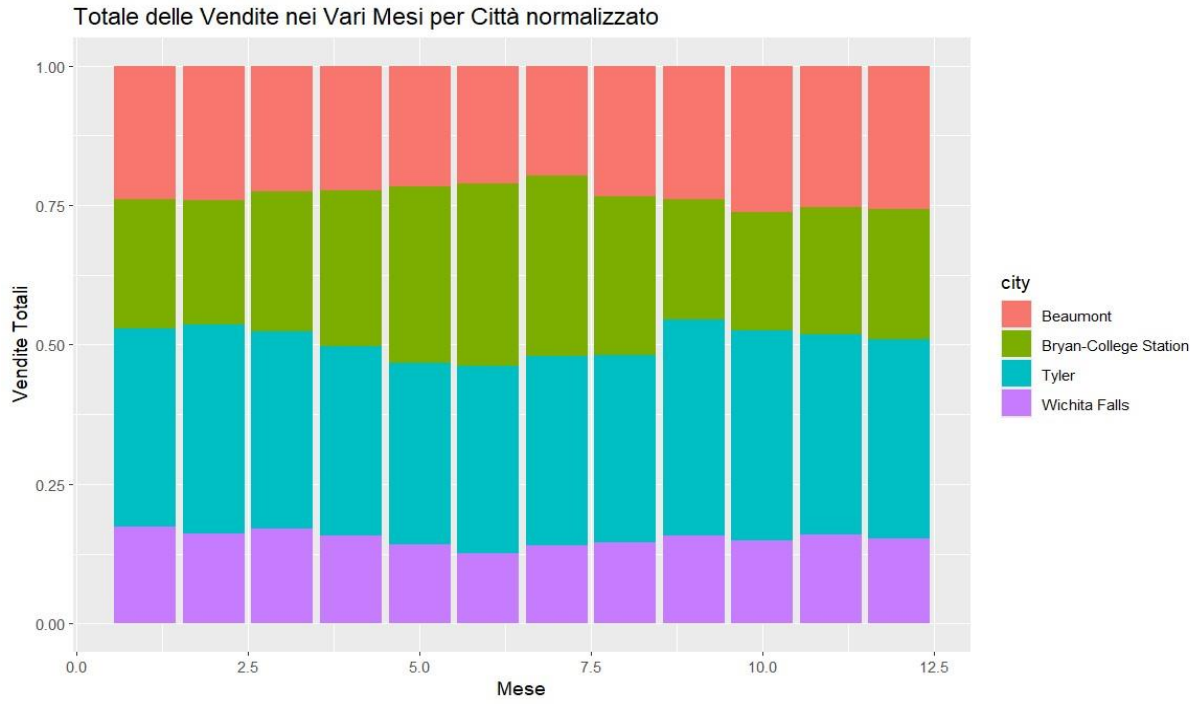
9.3 Usa un grafico a barre sovrapposte per confrontare il totale delle vendite nei vari mesi, sempre considerando le città. Prova a commentare ciò che viene fuori. Già che ci sei prova anche il grafico a barre normalizzato. Consiglio: Stai attento alla differenza tra geom_bar() e geom_col(). PRO LEVEL: cerca un modo intelligente per inserire ANCHE la variabile Year allo stesso blocco di codice, senza però creare accrocchi nel grafico.

```
# Grafico a barre sovrapposte per le vendite nei mesi
ggplot(data_texas_real_estate, aes(x = month, y = sales, fill = city)) +
  geom_bar(stat = "identity", position = "stack") +
  labs(title = "Totale delle Vendite nei Vari Mesi per Città", x = "Mese", y = "Vendite Totali")
```



<u>Commento sul grafico</u>: Dalle barre del grafico si nota che i mesi favorevoli dove si realizzano il numero di vendite maggiori sono nei mesi di maggio (mese n.5) ed agosto (mese n.8)

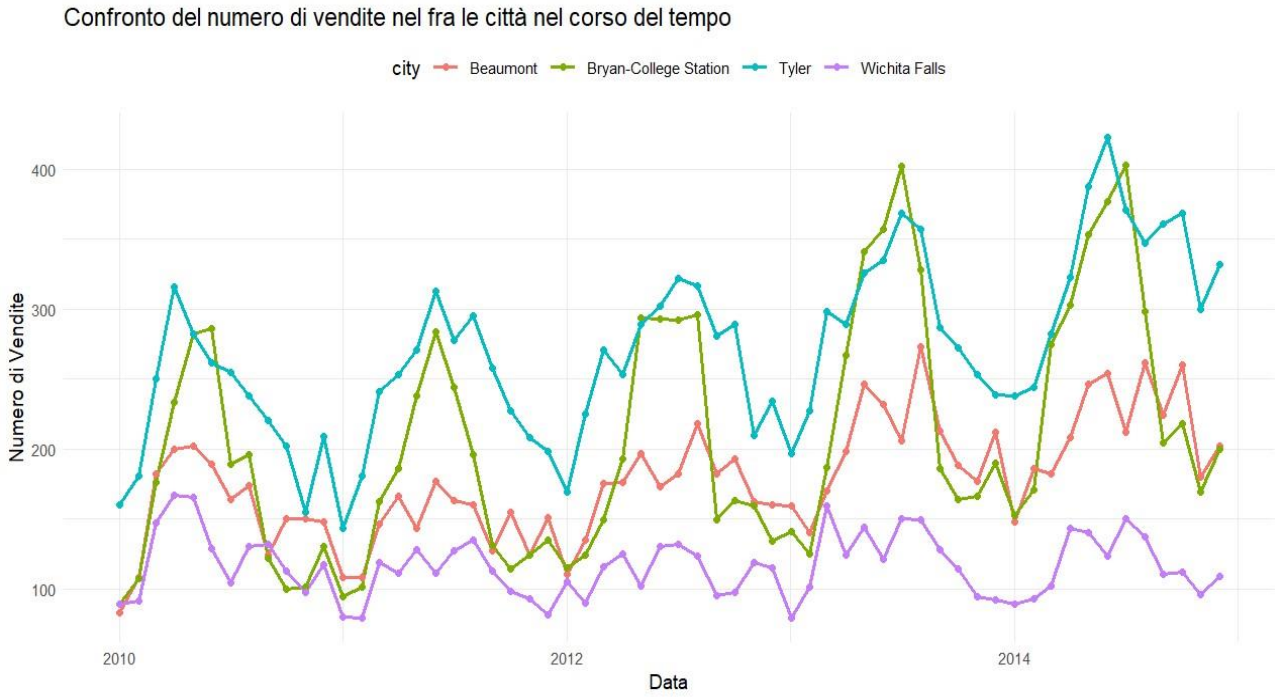
```r
# Grafico a barre sovrapposte per le vendite nei mesi normalizzato
ggplot(data_texas_real_estate, aes(x = month, y = sales, fill = city)) +
  geom_bar(stat = "identity", position = "fill") +
  labs(title = "Totale delle Vendite nei Vari Mesi per Città normalizzato", x = "Mese", y = "Vendite Totali")
```



Commento sul grafico: Dalla normalizzazione del grafico a barre sovrapposte notiamo che, per alcune città il mese considerato può inficiare leggermente nel volume di vendite. Infatti di nota che in Bryan-College-Station si vendono maggiori appartamenti nei mesi più caldi, mentre nelle città come Beaumont e Tyler si vendono maggiormente appartamenti nei mesi invernali/autunnali. Whichita Falls non è molto influenzata dal fenomeno stagionale delle vendite.

10. PROVA A CREARE UN LINE CHART DI UNA VARIABILE A TUA SCELTA PER FARE CONFRONTI COMMENTATI FRA CITTÀ E PERIODI STORICI. TI AVVISO CHE PROBABILMENTE ALL'INIZIO TI VERRANNO FUORI LINEE STORTE E POCO CHIARE, MA NON DEMORDERE.

```r
#SERIE STORICHE
# Caricamento del dataset
# Unione di anno e mese per creare una colonna "date"
data_texas_real_estate$date <- as.Date(with(data_texas_real_estate, paste(year, month, "01", sep="-")), "%Y-%m-%d")
# Filtrare per le città di interesse
city_filtered_data <- data %>%
  filter(city %in% c("Beaumont", "Bryan-College Station", "Tyler", "Wichita Falls"))
# Raggruppamento per data e città, e somma delle vendite
library(dplyr)
time_series_cities <- city_filtered_data %>%
  group_by(date, city) %>%
  summarise(total_sales = sum(sales))
time_series_cities
# Creazione del grafico a linee per il confronto tra le città
library(ggplot2)
ggplot(time_series_cities, aes(x = date, y = total_sales, color = city)) +
  geom_line(size = 1) +
  geom_point() +
  labs(title = "Confronto del numero di vendite nel fra le città nel corso del tempo",
       x = "Data",
       y = "Numero di Vendite") +
  theme_minimal() +
  theme(legend.position = "top")  # Sposta la legenda in alto
```



<u>Commento sul grafico</u>: Dalla visualizzazione della serie storica si nota che i trends di vendita a partire dal 2010 sono state crescenti per città Tyler, mentre per Bryan college station ci sono stati periodi di rialzo e ribasso durante la serie storica con picchi che negli anni fra 2012-2013 superavano addirittura le vendite nella località di tyler. Le vendite in Beumont mostrano un trend leggermente crescente e senza picchi eccessivi, mostrando un mercato abbastanza "stabile", mentre per la citta di Wichita Fall si è posizionata già dal 2010 negli utlimi posti fra il numero di vendite intercittà, con un trend quasi decrescente nel periodo totale considerato.