Quantiful Assessment

Yongqi Liang

21/08/2021

Task 1

Load the data into an environment of your choice. Have a look at the structure of the data. What information are you provided?

```
# read the datasets for Exploratory Analysis and Modelling.
dataset_train <- read_csv("train.csv")
dataset_stores <- read_csv("stores.csv")

# have a look what information does the data conveys.
head(dataset_train)
```

```
## # A tibble: 6 x 5
##   Store  Dept Date       Weekly_Sales IsHoliday
##   <dbl> <dbl> <date>            <dbl> <lgl>
## 1     1     1 2010-02-05       24924. FALSE
## 2     1     1 2010-02-12       46039. TRUE
## 3     1     1 2010-02-19       41596. FALSE
## 4     1     1 2010-02-26       19404. FALSE
## 5     1     1 2010-03-05       21828. FALSE
## 6     1     1 2010-03-12       21043. FALSE
```

```
head(dataset_stores)
```

```
## # A tibble: 6 x 3
##   Store Type    Size
##   <dbl> <chr>  <dbl>
## 1     1 A     151315
## 2     2 A     202307
## 3     3 B      37392
## 4     4 A     205863
## 5     5 B      34875
## 6     6 A     202505
```

```
# check the dimensions of the loaded datasets.
dim(dataset_train)
```

```
## [1] 421570      5
```

```
dim(dataset_stores)
```

```
## [1] 45  3
```

The `dataset_train` has 421570 rows and 5 columns. Each row shows the weekly sales in different departments, the store they are from, and whether or not this week is on holiday. It also contains the date of the week they recorded.

The `dataset_stores` has 45 rows and 3 columns. Each row gives us information about the stores, the type and size of these stores.

```
# join these two datasets by "Store".
dataset_sales <- dataset_train %>% left_join(dataset_stores, by = c("Store"))

dataset_sales$item_name <- NULL
head(dataset_sales)
```

```
## # A tibble: 6 x 7
##   Store  Dept Date       Weekly_Sales IsHoliday Type    Size
##   <dbl> <dbl> <date>            <dbl> <lgl>     <chr>  <dbl>
## 1     1     1 2010-02-05       24924. FALSE     A     151315
## 2     1     1 2010-02-12       46039. TRUE      A     151315
## 3     1     1 2010-02-19       41596. FALSE     A     151315
## 4     1     1 2010-02-26       19404. FALSE     A     151315
## 5     1     1 2010-03-05       21828. FALSE     A     151315
## 6     1     1 2010-03-12       21043. FALSE     A     151315
```

```
# remove unused datasets.
rm(dataset_train)
rm(dataset_stores)
dataset_sales <- as.data.frame(dataset_sales)
```

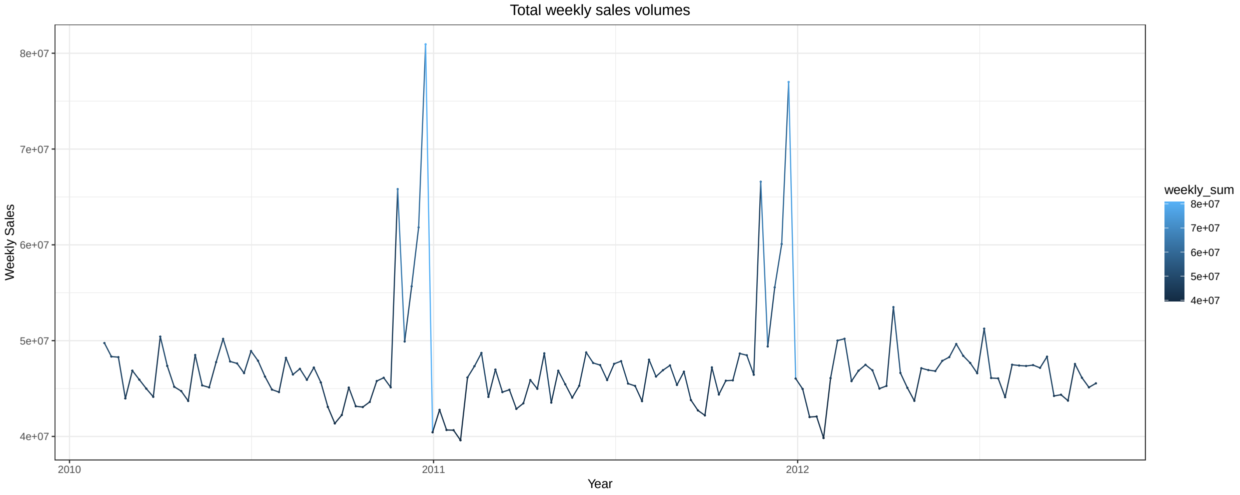
```
# check the structure of the new dataset
str(dataset_sales)
```

```
## 'data.frame':    421570 obs. of  7 variables:
##  $ Store       : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Dept        : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Date        : Date, format: "2010-02-05" "2010-02-12" ...
##  $ Weekly_Sales: num  24925 46039 41596 19404 21828 ...
##  $ IsHoliday   : logi  FALSE TRUE FALSE FALSE FALSE FALSE ...
##  $ Type        : chr  "A" "A" "A" "A" ...
##  $ Size        : num  151315 151315 151315 151315 151315 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Store = col_double(),
##   ..   Dept = col_double(),
##   ..   Date = col_date(format = ""),
##   ..   Weekly_Sales = col_double(),
##   ..   IsHoliday = col_logical()
##   .. )
```

Task 2. Prepare and comment on the following exploratory plots:

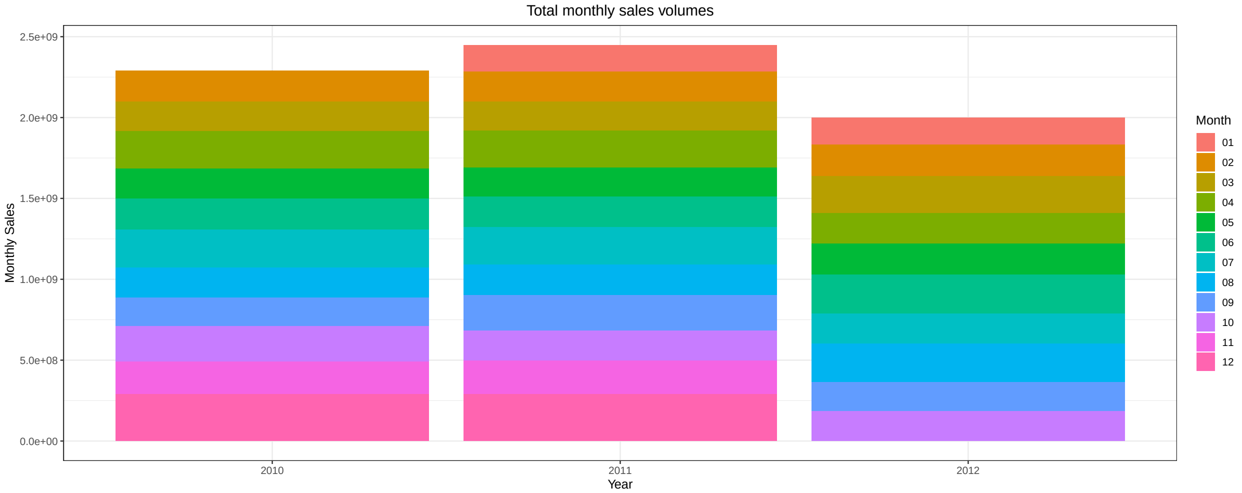
a) Total weekly and monthly sales volumes.

```r
# weekly
dataset_sales$Year <- format(dataset_sales$Date, "%Y")
dataset_sales %>%
  group_by(Date) %>%
  summarise("weekly_sum" = sum(Weekly_Sales)) %>%
  ggplot(aes(x = Date, y = weekly_sum, color = weekly_sum)) +
  geom_line() +
  geom_point(size=0.25) +
  labs(title = "Total weekly sales volumes", x = "Year", y = "Weekly Sales") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5))
```



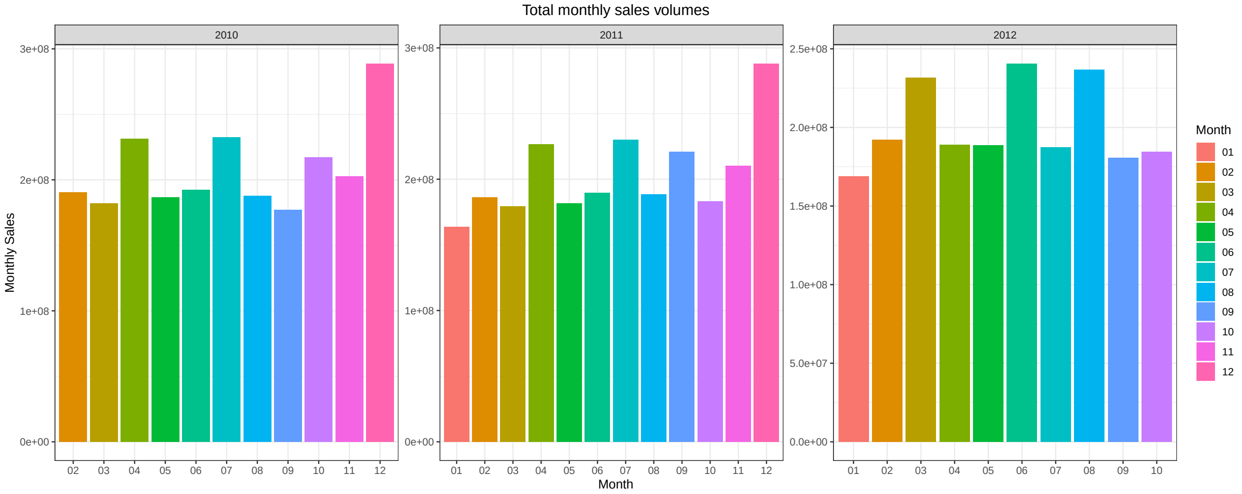
From the plot, we can see there is no obvious increasing or decreasing trend for the weekly sales from 2010 to 2011. But there are 4 weeks at the end of 2011 and 2012 respectively have weekly sales much larger than usual. One of the possible reasons for this is that these 3 weeks are exactly around Christmas, so people might be more willing and have more chance to consume.

```r
# monthly
dataset_sales$Month <- format(dataset_sales$Date, "%m")
dataset_sales %>%
  group_by(Year, Month) %>%
  summarise(Monthly_Sales = sum(Weekly_Sales)) %>%
  ungroup() %>%
  ggplot(aes(x = Year, y = Monthly_Sales, fill = as.factor(Month))) +
  geom_bar(stat = "identity") +
  labs(title = "Total monthly sales volumes", x = "Year", y = "Monthly Sales", fill = "Month") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5))
```



The difference might not be too obvious. So I would plot a bar chart.

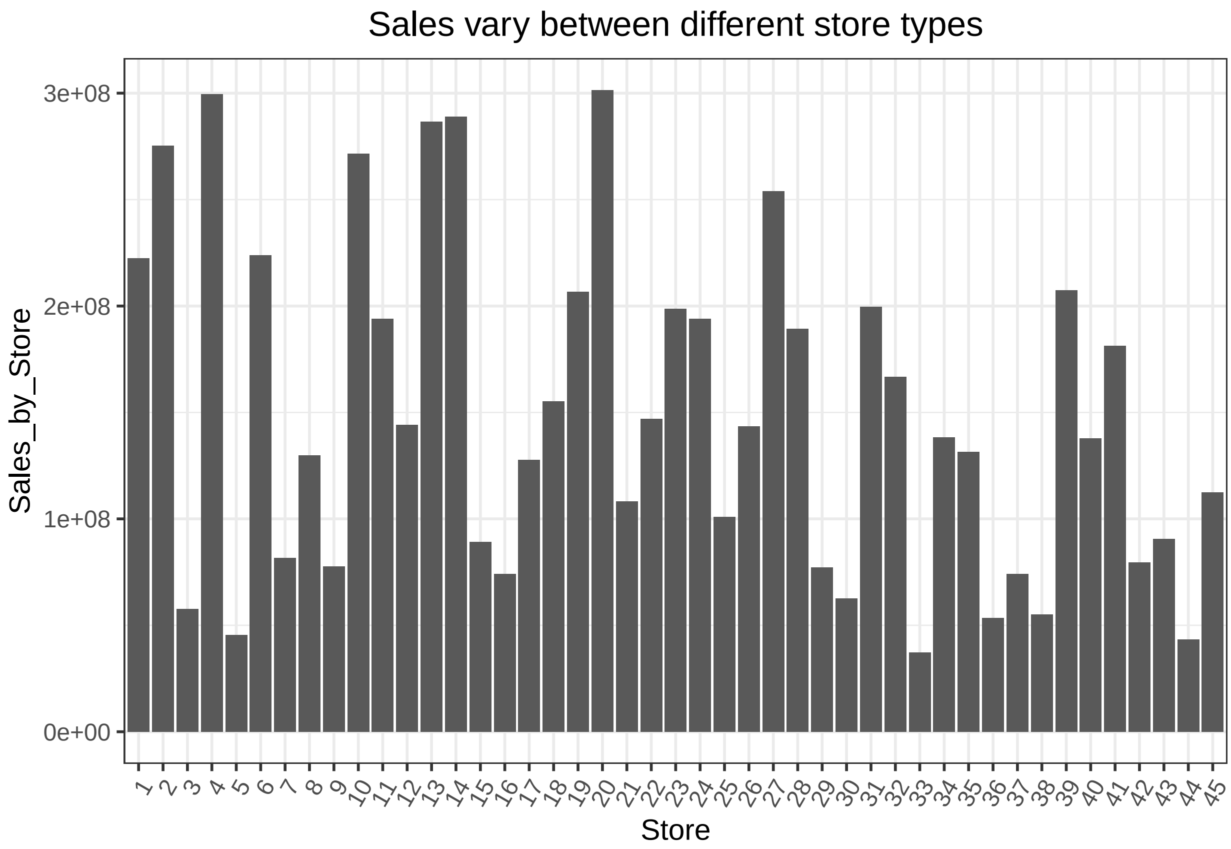
```
dataset_sales %>%
  group_by(Year, Month) %>%
  summarise(Monthly_Sales = sum(Weekly_Sales)) %>%
  ggplot(aes(x = Month, y = Monthly_Sales, fill = Month, group = Year)) +
  geom_bar(stat = "identity") +
  facet_wrap(~ Year, scales="free") +
  labs(title = "Total monthly sales volumes", x = "Month", y = "Monthly Sales") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5))
```



I separate the data into different years this time. The monthly sales plots for 2010 and 2011 show that in December, the monthly sales are markedly larger than other months. Furthermore, in April and July in 2010 and 2011, and September of 2011, the monthly sales for these months are slightly larger than those for other months in the same year. It can be explained by that the inter semester or mid-semester break usually happens during these months. Furthermore, the monthly sales in 2012 are slightly larger than in 2010 and 2011.

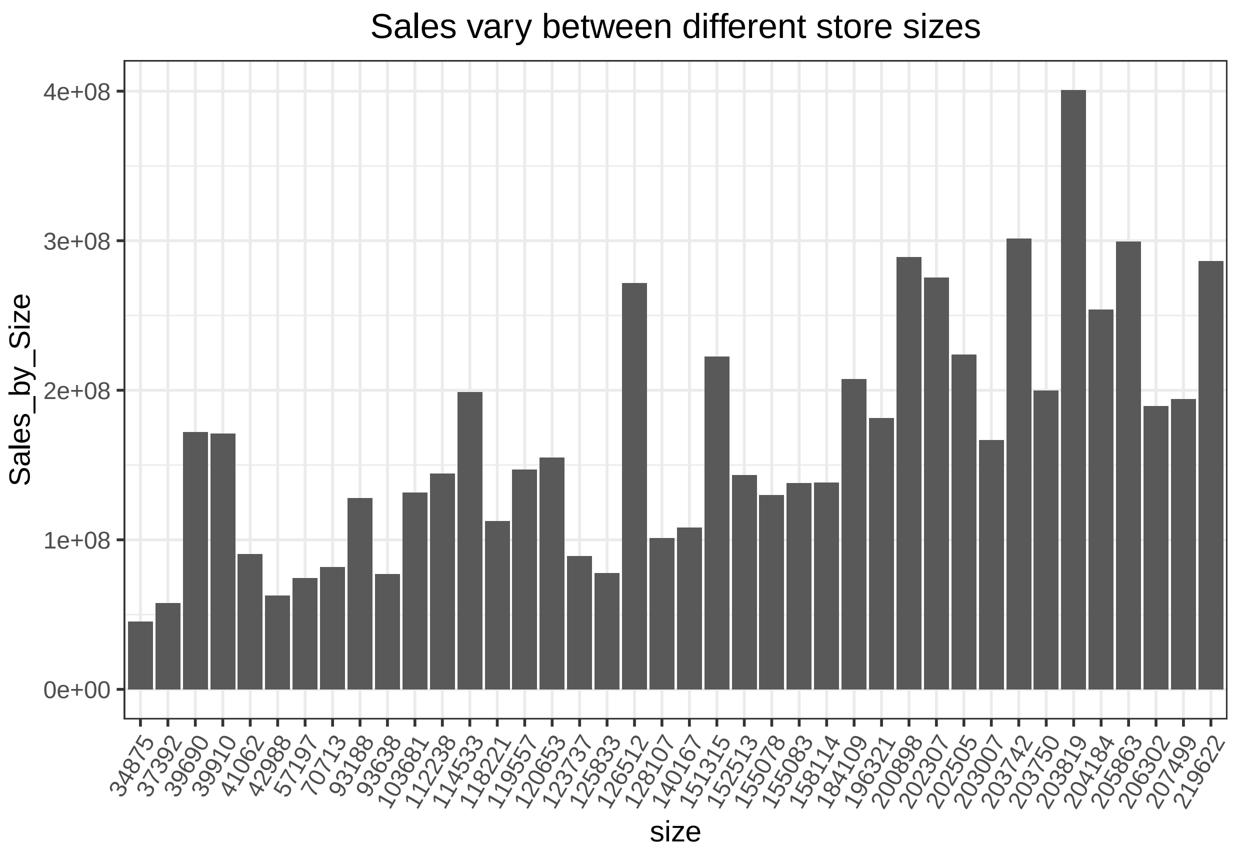
b) Sales volume by store - explore how sales vary between different store types and sizes.

```r
# by store types
dataset_sales$Store <- as.factor(dataset_sales$Store)
dataset_sales %>%
  group_by(Store) %>%
  summarise("Sales_by_Store" = sum(Weekly_Sales)) %>%
  ggplot(aes(x = Store, y = Sales_by_Store)) +
  geom_bar(stat = "identity") +
  labs(title = "Sales vary between different store types", x = "Store") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 60, hjust = 1))
```



We can see that the variation of the sales in different stores is large.

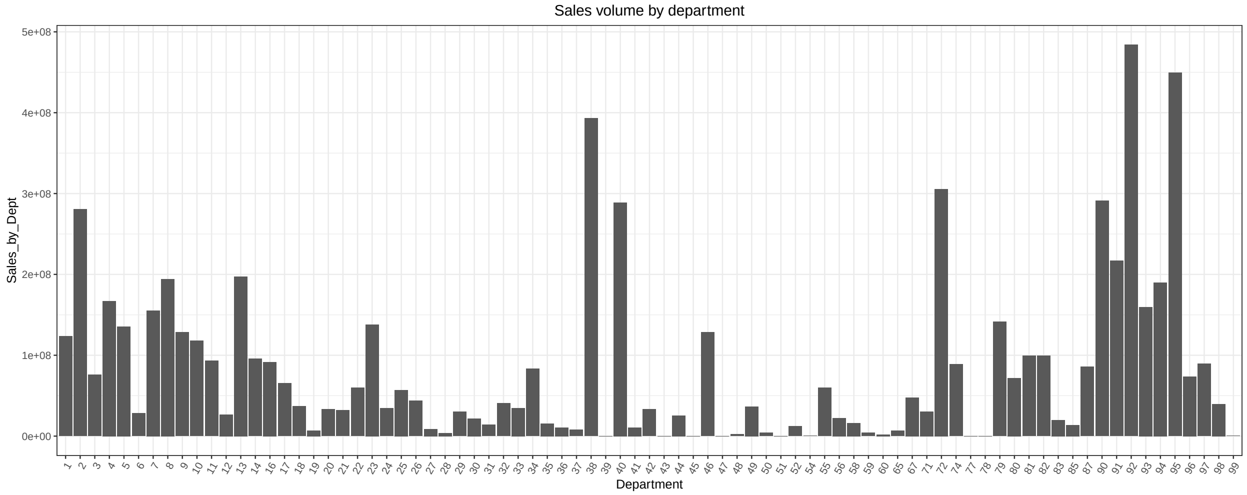
```r
# by sizes
dataset_sales$Size <- as.factor(dataset_sales$Size)
dataset_sales %>%
  group_by(Size) %>%
  summarise("Sales_by_Size" = sum(Weekly_Sales)) %>%
  ggplot(aes(x = Size, y = Sales_by_Size)) +
  geom_bar(stat = "identity") +
  labs(title = "Sales vary between different store sizes", x = "size") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 60, hjust = 1))
```



The plot shows a slightly increasing trend for the sales volume as the size of the store increases. It is not surprising as it makes sense that the store with larger size usually has more rooms to store their goods and have more places to sell them. Furthermore, a bigger store would have a better customer flow as it can contain more customers which increases the chance of making sales.

c) Sales volume by department.

```
# by department
dataset_sales$Dept <- as.factor(dataset_sales$Dept)
dataset_sales %>%
  group_by(Dept) %>%
  summarise("Sales_by_Dept" = sum(Weekly_Sales)) %>%
  ggplot(aes(x = Dept, y = Sales_by_Dept)) +
  geom_bar(stat = "identity") +
  labs(title = "Sales volume by department", x = "Department") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 60, hjust = 1))
```



From the plot, we can see the sales volume varies among different departments. The variation is extremely large between some of the departments. For example, the sale difference between department 77 and department 92 is approximately 4.8e+08. But we do not have information about what these departments are, as they are all represented by numbers.

Task 3. Investigate the effect of holidays on sales volume. Are all holidays accounted for with the IsHoliday flag in the data set? Describe your findings using appropriate visuals and commentary.

In order to find whether or not all the `IsHoliday` values are labeled correctly, Firstly, I have created a vector called "hlist" which contain some common holiday.

```
hlist <- c("ChristmasDay","GoodFriday","USLaborDay", "NewYearsDay","USThanksgivingDay", "BoxingDay")

myholidays  <- dates(as.character(holiday(2010:2012,hlist)),format="Y-M-D")
```

Then I want to extract all the distinct date and their corresponding `IsHoliday` values from the whole data set as it is always easier to handle a smaller data set and I do not want to count the observation which has the wrong `IsHoliday` label more than once.

```
unique_date <- dataset_sales %>%
  select(Date, IsHoliday) %>%
  group_by(Date) %>%
  distinct(Date, .keep_all = TRUE)

head(unique_date)
```

```
## # A tibble: 6 x 2
## # Groups:   Date [6]
##   Date       IsHoliday
##   <date>     <lgl>
## 1 2010-02-05 FALSE
## 2 2010-02-12 TRUE
## 3 2010-02-19 FALSE
## 4 2010-02-26 FALSE
## 5 2010-03-05 FALSE
## 6 2010-03-12 FALSE
```

Once I have this sub-data, I can use the function is.holiday() in the chron package to check whether or not the individual is label correctly. I did it by passing my list of dates in this function. Since is.holiday() will return a logical value, then I compare these logical values to my `IsHoliday` column to get a series of logical values. If the logical value is `FALSE`, it means the label in the `IsHoliday` column is wrong. Then I use the sum() to get the total number of observations which is not labeled correctly because I want to know how many of them have the wrong label.

```
sum(is.holiday(unique_date$Date, myholidays) != unique_date$IsHoliday)
```

```
## [1] 13
```

Then I use the which() function to get the index of the date that has problem and extract them to check whether or not they are really labeled wrong or is just due to my list of holidays is not fit for New Zealand.

```
incorrect <- unique_date[which(is.holiday(unique_date$Date, myholidays) != unique_date$IsHoliday),]
incorrect
```

```
## # A tibble: 13 x 2
## # Groups:   Date [13]
##    Date       IsHoliday
##    <date>     <lgl>
##  1 2010-02-12 TRUE
##  2 2010-04-02 FALSE
##  3 2010-09-10 TRUE
##  4 2010-11-26 TRUE
##  5 2010-12-31 TRUE
##  6 2011-02-11 TRUE
##  7 2011-04-22 FALSE
##  8 2011-09-09 TRUE
##  9 2011-11-25 TRUE
## 10 2011-12-30 TRUE
## 11 2012-02-10 TRUE
## 12 2012-04-06 FALSE
## 13 2012-09-07 TRUE
```

Good Friday is a holiday in New Zealand and it happened on April 2nd in 2010, April 22nd in 2011 and April 6th in 2012. However, the labels of the `IsHoliday` column corresponding to these days are all marked as `FALSE`. Hence, we can conclude that not all holidays are accounted for with the `IsHoliday` flag in the data set.

Furthermore, 2010-02-12, 2011-02-11 and 2012-02-10 are close to Waitangi Day, 2010-12-31 and 2011-12-30 are close to Christmas Day. Thus, I would not consider their labels are incorrect.

Hence, I will modify the `IsHoliday` values for the days labeled incorrect to make a more accurate result of investigating the effect of holidays on sales volume.

```
temp <- which(is.holiday(unique_date$Date, myholidays) != unique_date$IsHoliday)

incorrect_day_F <- unique_date[temp,][c(3, 4, 8, 9, 13),]

incorrect_day_T <- unique_date[temp,][c(2, 7, 12),]

dataset_sales$IsHoliday[which(dataset_sales$Date %in% incorrect_day_F$Date)] = FALSE

dataset_sales$IsHoliday[which(dataset_sales$Date %in% incorrect_day_T$Date)] = TRUE
```

Double-check to make sure I have modified the labels correctly by repeating the same steps using the modified version of `dataset_sales`. The "incorrect" labels should only be the days close to Waitangi Day and Christmas Day.

```
unique_date2 <- dataset_sales %>%
  select(Date, IsHoliday) %>%
  group_by(Date) %>%
  distinct(Date, .keep_all = TRUE)

incorrect2 <- unique_date2[which(is.holiday(unique_date2$Date, myholidays) != unique_date2$IsHoliday),]
incorrect2
```

```
## # A tibble: 5 x 2
## # Groups:   Date [5]
##   Date       IsHoliday
##   <date>     <lgl>
## 1 2010-02-12 TRUE
## 2 2010-12-31 TRUE
## 3 2011-02-11 TRUE
## 4 2011-12-30 TRUE
## 5 2012-02-10 TRUE
```

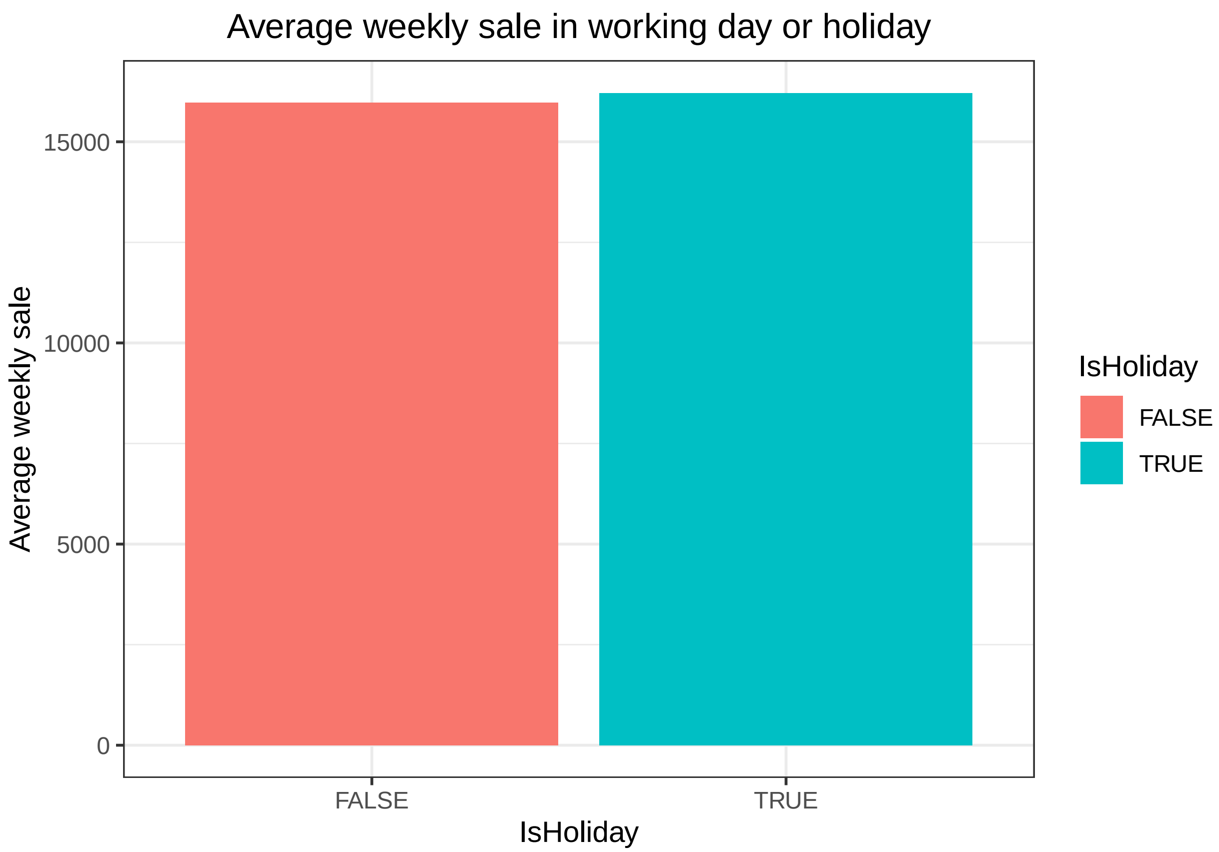
```
head(dataset_sales$Date[which(dataset_sales$Date %in% incorrect2$Date)])
```

```
## [1] "2010-02-12" "2010-12-31" "2011-02-11" "2011-12-30" "2012-02-10"
## [6] "2010-02-12"
```

It is exactly what I expect.

Have a look at the average weekly sales in these 2 groups:

```
dataset_sales %>%
  group_by(IsHoliday) %>%
  summarise("Average_weekly_sale" = mean(Weekly_Sales)) %>%
  ggplot(aes(y = Average_weekly_sale, x = IsHoliday, fill = IsHoliday)) + geom_bar(stat = "identity") +
  labs(title = "Average weekly sale in working day or holiday", y = "Average weekly sale") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5))
```



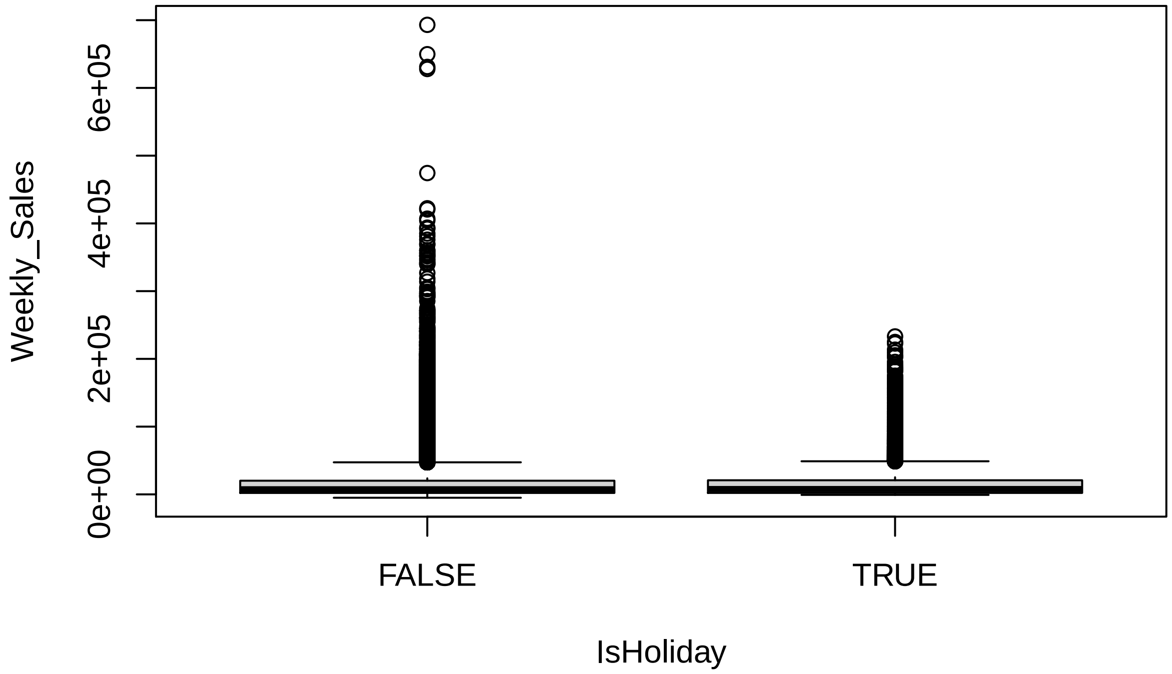
It can be seen that the sales volume in holiday is slightly larger. However, the difference is very tiny.

I will fit a simple model to support my finding.

```
# have a look at the data and how each weekly sale distribute in different groups
summaryStats(dataset_sales$Weekly_Sales, dataset_sales$IsHoliday)
```

```
##        Sample Size     Mean   Median  Std Dev Midspread
## FALSE      397842 15967.29 7594.485 22721.85  18087.83
## TRUE        23728 16215.45 7947.580 22530.76  18700.26
```

```
boxplot(Weekly_Sales~IsHoliday, data = dataset_sales)
```



```r
# Fit a linear model(assume all the assumption are valid)
fit = lm(Weekly_Sales~ IsHoliday, data = dataset_sales)
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: Weekly_Sales
##               Df     Sum Sq    Mean Sq F value Pr(>F)
## IsHoliday      1 1.3789e+09 1378949048  2.6734  0.102
## Residuals 421568 2.1744e+14  515795809
```

The p-value for `IsHoliday` is testing the null hypothesis $H_0 : \beta_1 = 0$. Here we can see that this p-value is not statistically significant as p $= 0.102 > 0.05$. We do not have evidence that the sales volume is related to holiday.

Task 4

The directors of the retailer want to know how many sales to expect in total for the months of November and December 2012, to inform decision making around their marketing

One straightforward solution is to calculate the average sales volume in 2010 and 2011 for November and December. Because from task 2, we can see the monthly sales volume is different in each month, but the sales volume in the same month is similar in 2010 and 2011. So it is reasonable to predict the monthly sales for 2012 base on both 2010 and 2011. For example, the monthly sales volume in November is 288760533 in 2010, the monthly sales volume in November 2011 is 288078102. So we predict the sales volume in November 2012 to be their mean: $(288760533 + 288078102)/2 = 288419317.5$.

However, predicting the future value in this way might not be accurate since it ignores the fact that the overall trend for the sales volume from 2010 to 2012 may not be steady. It can be increasing or decreasing. For example, if all the monthly sales volume in 2011 is greater than that in 2010, it is reasonable to expect that the monthly sales volume in 2012 should be greater than 2011. However, calculating the 2012 sales volume by average the amount in 2010 and 2011 will give us a value less than that in 2011. Hence, it might not be a suitable method.

Another method to predict the future value is using Holt-Winter model. Holt-Winters is one of the most popular forecasting techniques for time series. It uses exponential smoothing to encode lots of values from the past and use them to predict "typical" values for the present and future.

Comparing these two methods, I think using Holt-Winter model to make a prediction is the best approach. Because it is easy to implement and it can handle lots of complicated seasonal patterns by simply finding the central value, then adding in the effects of slope and seasonality.

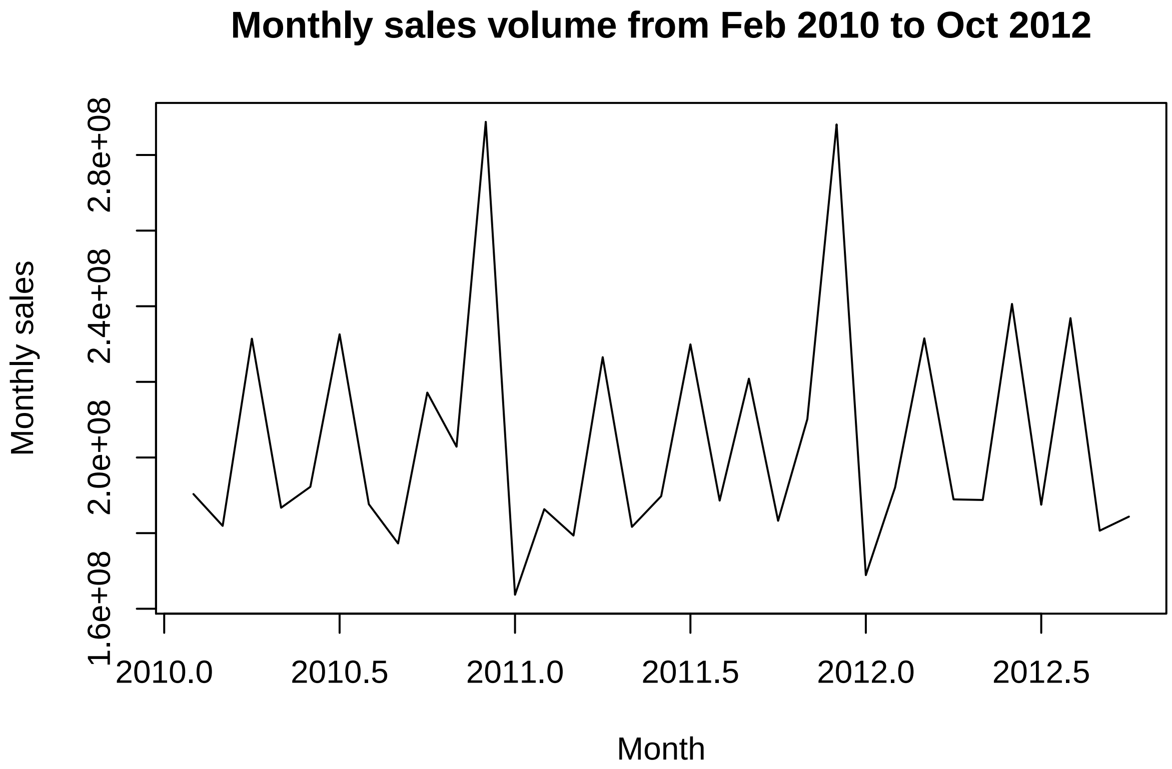
I will implement Holt-Winter model to predict the sales volume in November and December 2012:

```
#Reading Time Series Data
value <- dataset_sales %>% group_by(Year, Month) %>% summarise("Monthly_Sales" = sum(Weekly_Sales))

timeseries <- ts(value$Monthly_Sales, frequency=12, start=c(2010,2))
timeseries
```

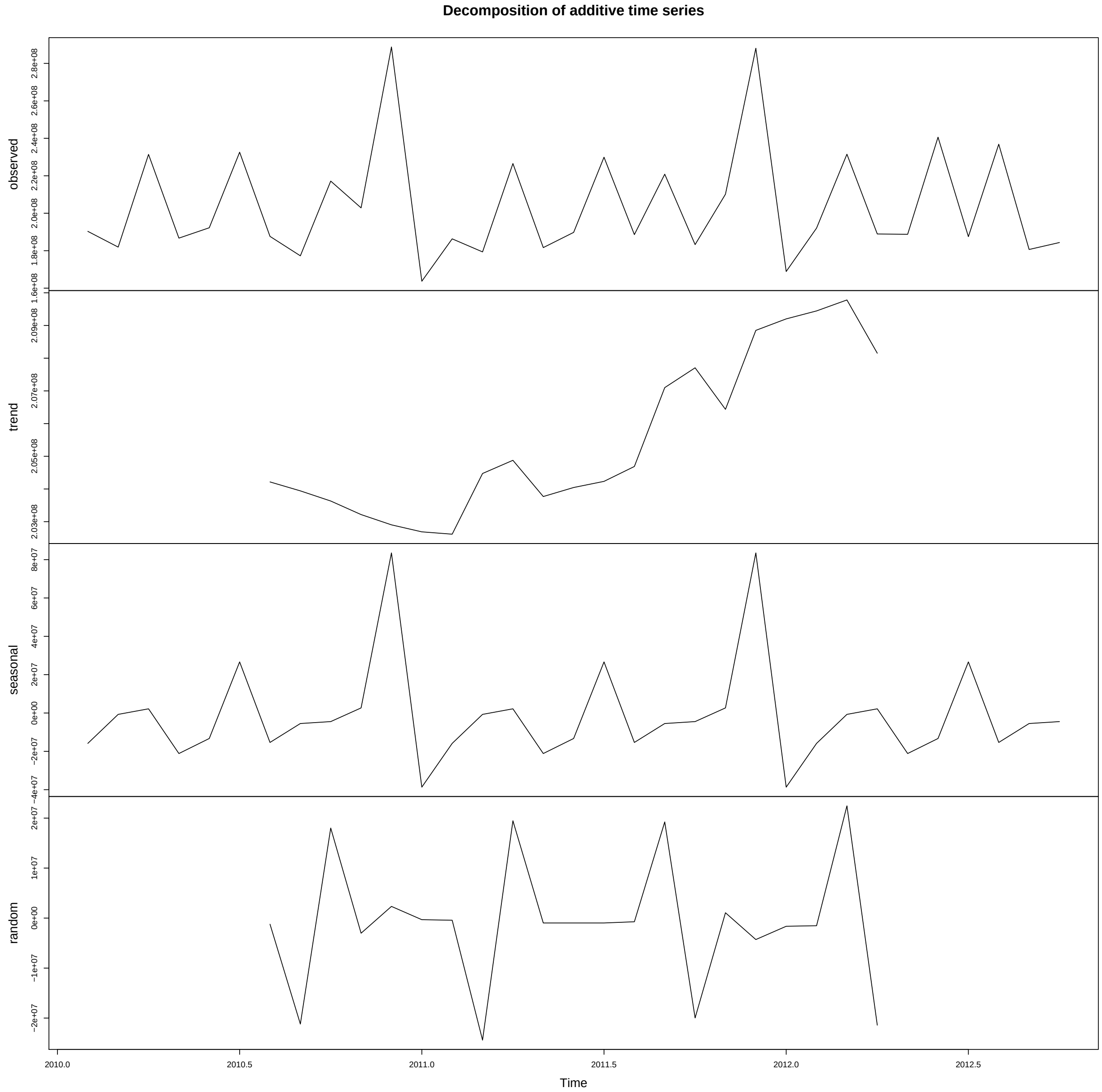
```
##            Jan       Feb       Mar       Apr       May       Jun       Jul
## 2010           190332983 181919803 231412368 186710934 192246172 232580126
## 2011 163703967 186331328 179356448 226526511 181648158 189773385 229911399
## 2012 168894472 192063580 231509650 188920906 188766479 240610329 187509452
##            Aug       Sep       Oct       Nov       Dec
## 2010 187640111 177267896 217161824 202853370 288760533
## 2011 188599332 220847738 183261283 210162355 288078102
## 2012 236850766 180645544 184361680
```

```
# Plotting Time Series
plot(timeseries, xlab="Month", ylab="Monthly sales", main="Monthly sales volume from Feb 2010 to Oct 20
```



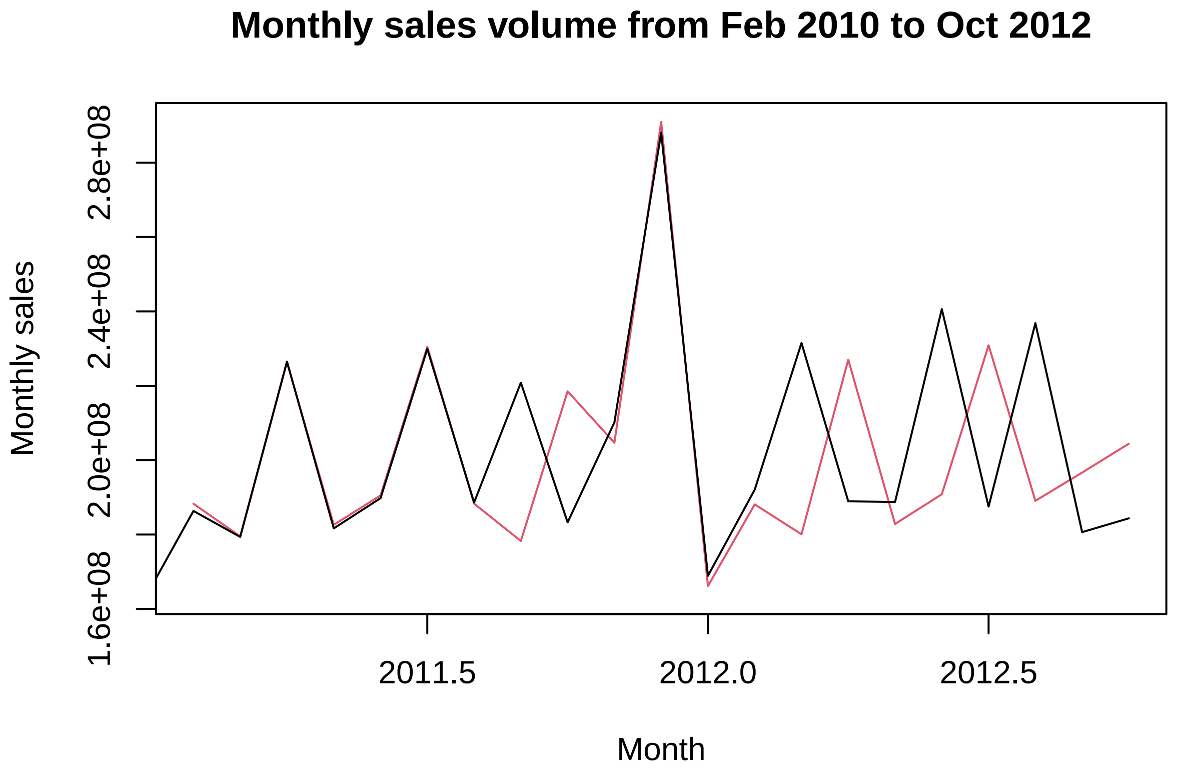
```
# estimate the trend, seasonal and irregular components of this time series
timeseriescomponents <- decompose(timeseries)
```

```
#plot the estimated trend, seasonal, and irregular components of the time series by using the "plot()"
plot(timeseriescomponents)
```

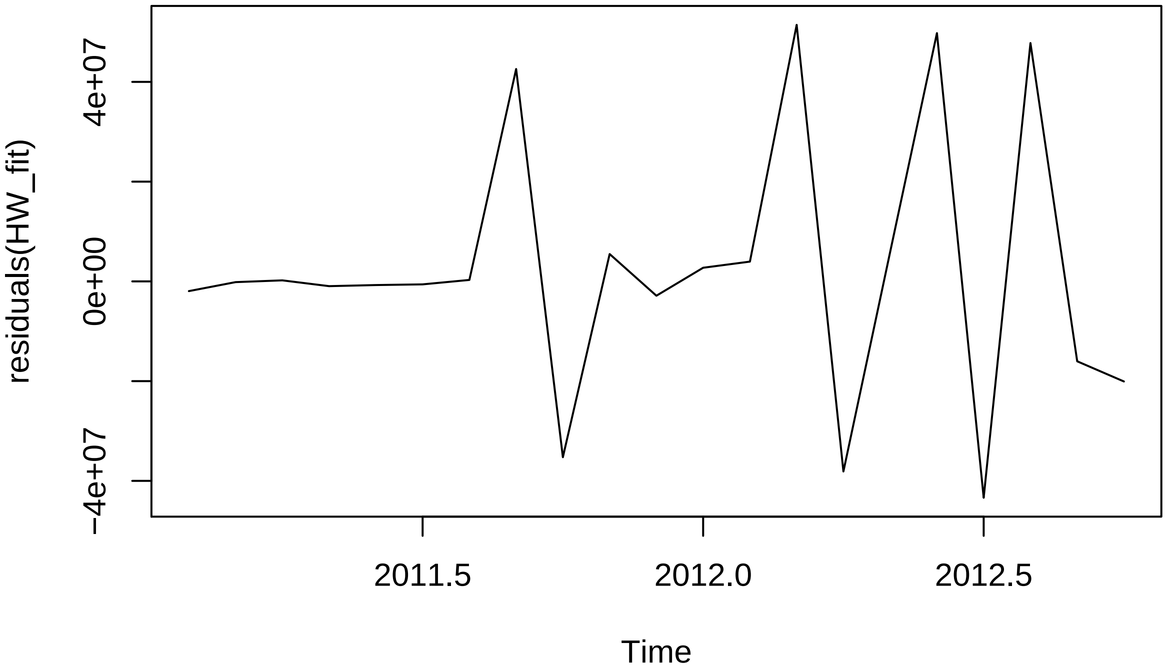


The plot above shows the original time series (top), the estimated trend component (second from top), the estimated seasonal component (third from top), and the estimated irregular component (bottom). We see that the estimated trend component shows a small decrease from about 2.045e+08 in June 2010 to about 2.025e+08 in February 2011, followed by a unstable increase from then on to about 2.075e+08 in March 2012. Since the increasing is not exponential and the random fluctuations in the time series seem to be roughly constant in size over time, so it is probably appropriate to describe the data using an additive Holt Winters Forecasting Model:

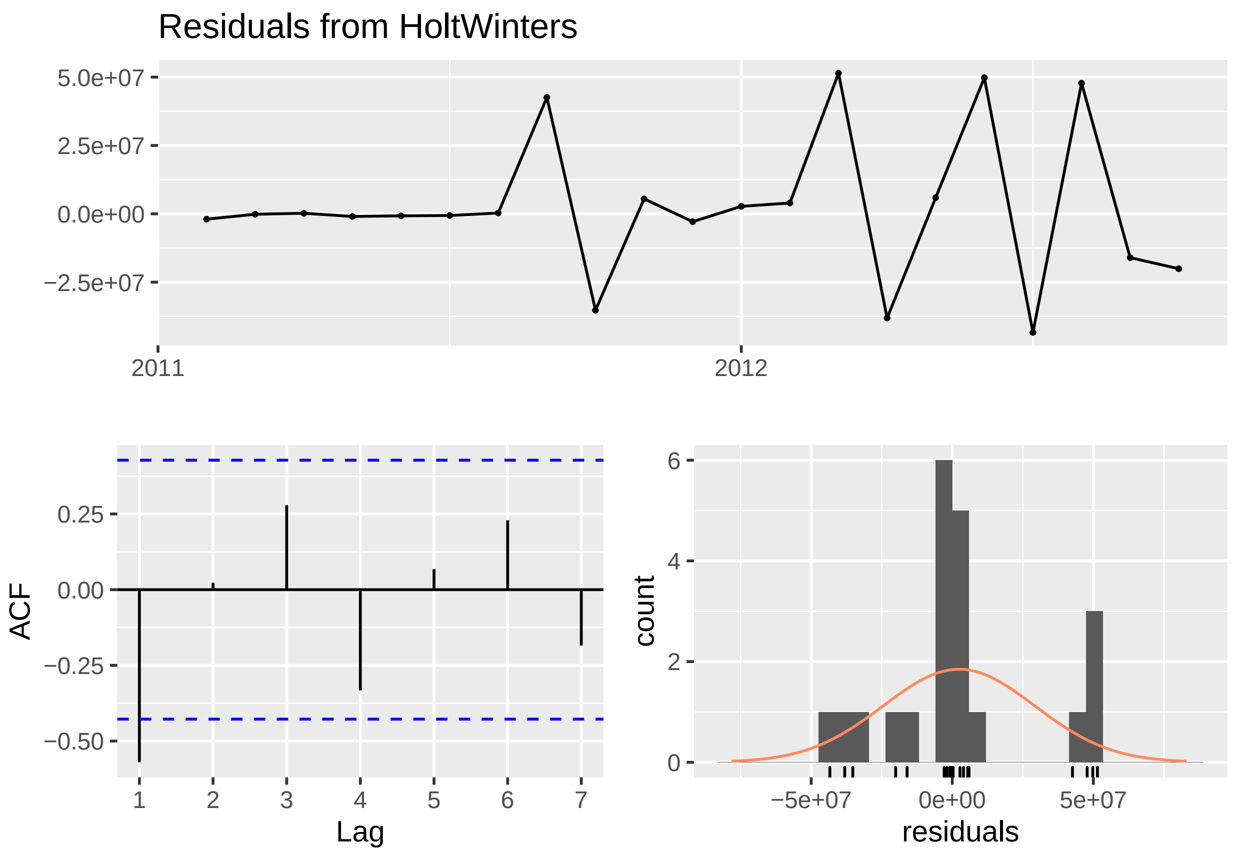
```
HW_fit = HoltWinters(timeseries, seasonal = "additive")
# model check:
plot(HW_fit, xlab="Month", ylab="Monthly sales", main="Monthly sales volume from Feb 2010 to Oct 2012")
```



```
plot.ts(residuals(HW_fit))
```

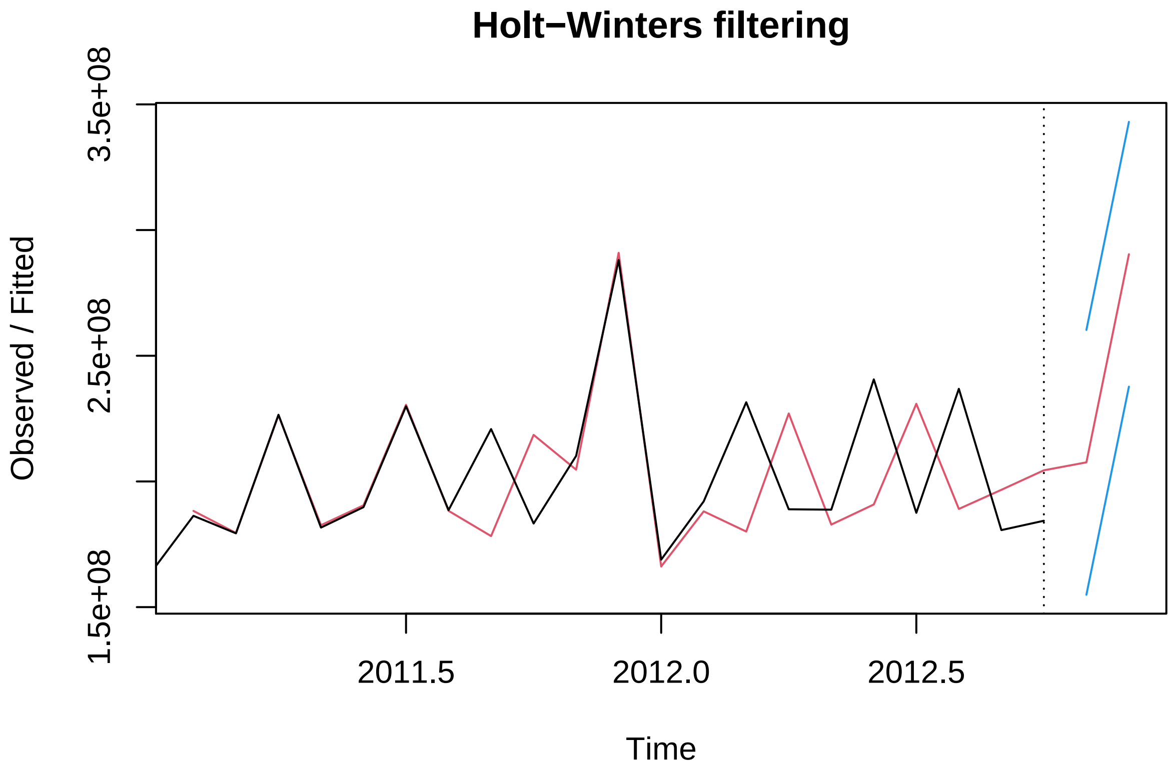


```
checkresiduals(HW_fit)
```



The residual series shows a constant mean of 0 and constant variance. The ACF of the residual series shows a slight negative autocorrelation at lag 1, but it is just outside the confidence bans, so it is not of concern. The Normal plots show a little departure from the Normal distribution as the rightmost column has a very high count, but the data set is not larger, so I would not be concerned about that. Overall, the forecasts follow the actual values. Except in 2012, but it still roughly captures the pattern.

```
# prediction for November and December 2012
HW_pred <- predict(HW_fit, n.ahead=2, prediction.interval=TRUE)
plot(HW_fit,HW_pred)
```



```r
# 95% prediction interval
HW_pred
```

```
##                fit       upr       lwr
## Nov 2012 207596777 260264255 154929300
## Dec 2012 290382044 343049522 237714567
```

From the table above, with a 95% prediction interval, we can predict that in November 2012, we expect the sale volume to be **207596777**, with lower bound **154929300** and upper bound **260264255**. In December , we expect the sale volume to be **290382044**, with lower bound **237714567** and upper bound **343049522**.

The forecast can be better if we have information in earlier years so that we can make a better adjustment for our model such as the parameter of the model.