KHAFAJI_FA5

Formative Assessment 5 - Store Sales Data Analysis

We'll analyze the store sales data, and create a Poisson distribution to examine how sales counts are influenced by the day of the week, promotions, holidays, and store size.

Loading and Initial Exploration of data

first let's load the data:

```
store_sales_data <- read_csv("store_sales_data.csv", show_col_types = FALSE) %>%
  mutate(across(c("promo", "holiday", "store_size", "day_of_week"), as.factor)) %>%
  mutate(across(where(~ is.factor(.) && nlevels(.) == 2),
               ~ factor(., levels = levels(.), labels = c("No", "Yes"))))

str(store_sales_data)
```

```
## tibble [5,000 x 5] (S3: tbl_df/tbl/data.frame)
##  $ day_of_week: Factor w/ 7 levels "0","1","2","3",..: 7 4 5 7 3 5 5 7 2 3 ...
##  $ promo      : Factor w/ 2 levels "No","Yes": 1 1 1 2 1 1 1 2 2 2 ...
##  $ holiday    : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 2 1 1 1 1 ...
##  $ store_size : Factor w/ 3 levels "large","medium",..: 2 2 1 3 2 2 2 1 2 3 ...
##  $ sales_count: num [1:5000] 18 13 24 16 11 13 12 34 19 8 ...
```

```
head(store_sales_data)
```

```
## # A tibble: 6 x 5
##   day_of_week promo holiday store_size sales_count
##   <fct>       <fct> <fct>   <fct>            <dbl>
## 1 6           No    No      medium              18
## 2 3           No    No      medium              13
## 3 4           No    No      large               24
## 4 6           Yes   No      small               16
## 5 2           No    No      medium              11
## 6 4           No    Yes     medium              13
```

```
skim(store_sales_data)
```

Table 1: Data summary

Name	store_sales_data
Number of rows	5000
Number of columns	5
Column type frequency:	
factor	4
numeric	1
Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
day_of_week	0	1	FALSE	7	3: 747, 4: 738, 0: 735, 5: 712
promo	0	1	FALSE	2	No: 3494, Yes: 1506
holiday	0	1	FALSE	2	No: 4522, Yes: 478
store_size	0	1	FALSE	3	med: 2512, sma: 1498, lar: 990

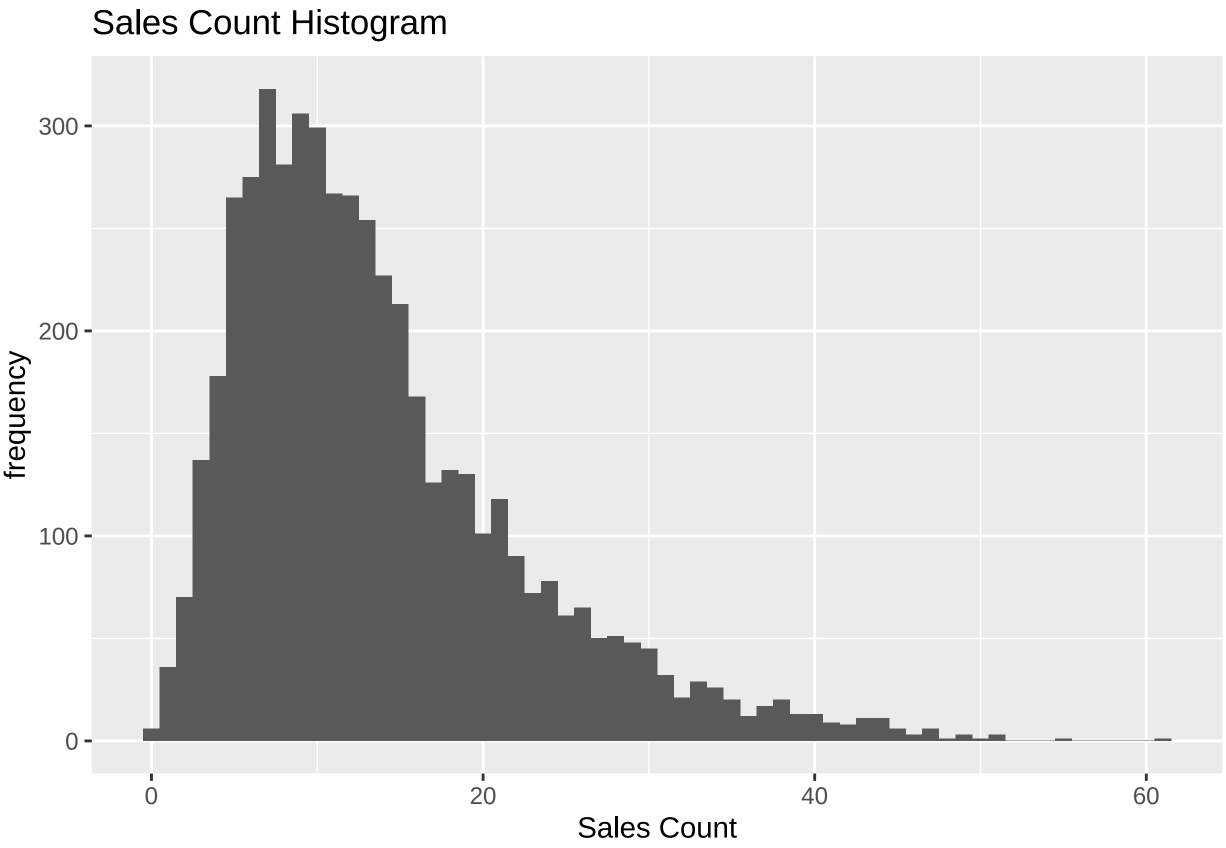
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
sales_count	0	1	13.73	8.68	0	7	12	18	61	

Next, we can look at the distributions and covariations of the features in the data.

Let's first check the distribution of sales count:

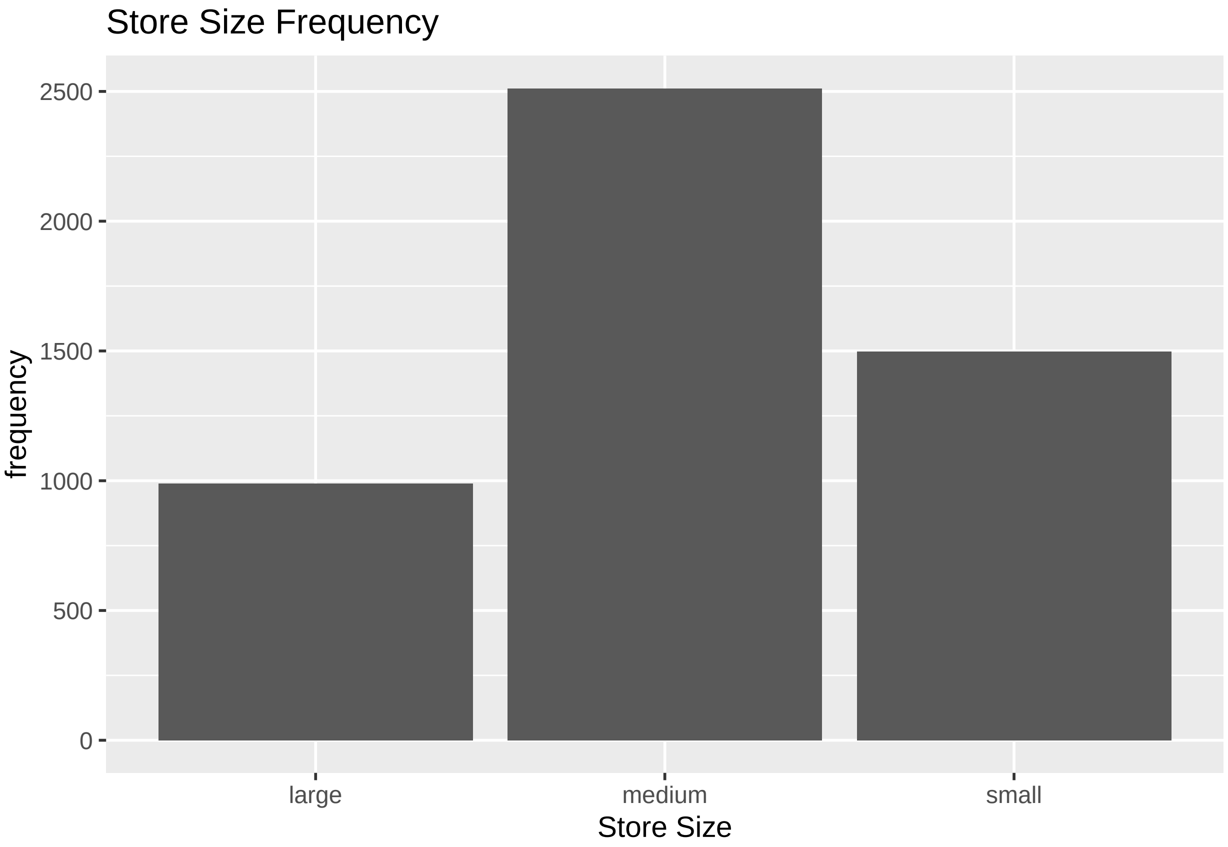
```r
store_sales_data %>% ggplot(aes(x=sales_count)) +
  geom_histogram(binwidth = 1) +
  xlab("Sales Count") +
  ylab("frequency") +
  ggtitle("Sales Count Histogram")
```



We can see that the distribution of sales count is skewed to the left, and is leptokurtic. Most of the days, the sales count ranges from 6 to 15.

Let's now look at frequency of each store size.

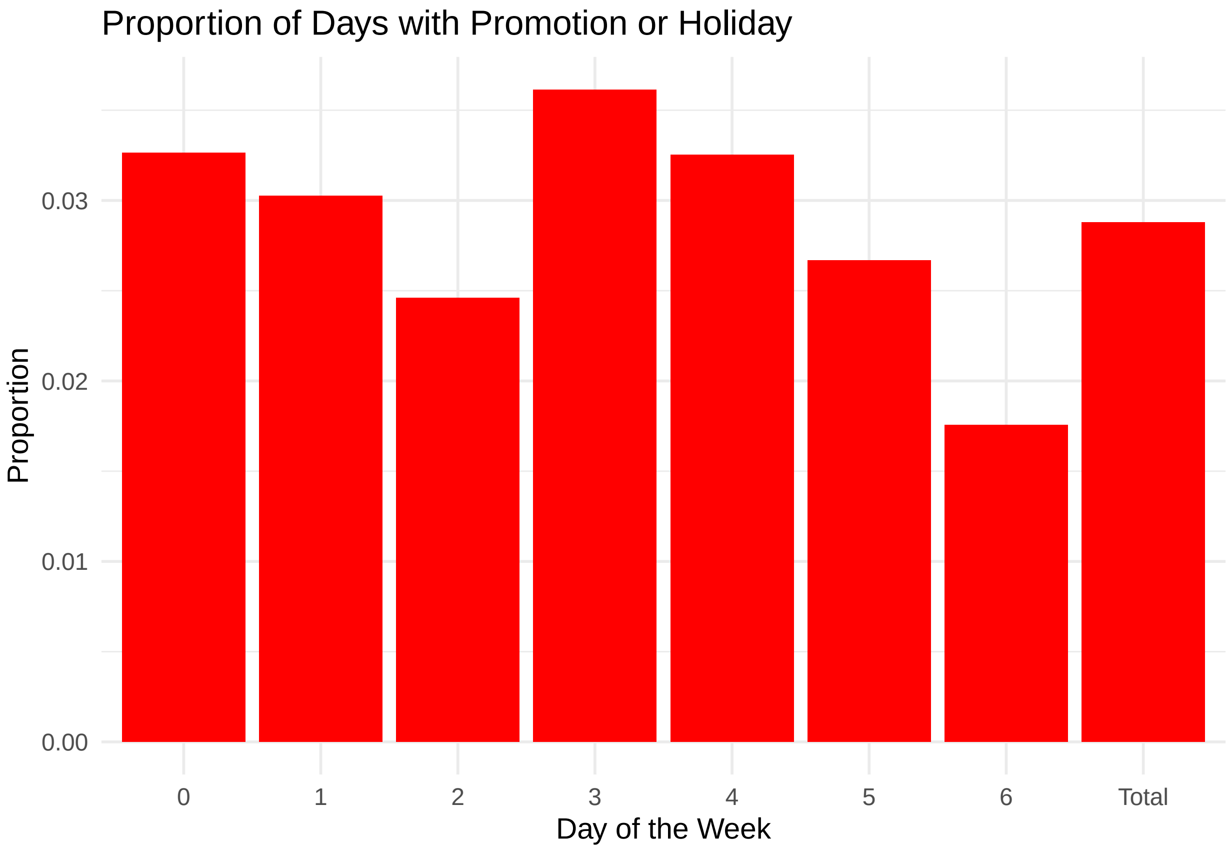
```
store_sales_data %>% ggplot(aes(x=store_size)) +
  geom_bar() + xlab("Store Size") +
  ylab("frequency") +
  ggtitle("Store Size Frequency")
```



As we can see from the graph, most of the stores are categorized as medium sized, with the next largest category being small stores, and the large stores having the least number of stores.

Now, let's try to chart the proportion of days with promo and holiday:

```
store_sales_data %>% mutate(holiday_and_promo = ifelse(promo=="Yes" & holiday == "Yes", 1, 0)) %>%
  group_by(day_of_week) %>% summarise(proportion = mean(holiday_and_promo)) %>%
  ungroup() %>%
  adorn_totals(name = "Total", where = "row") %>%
  mutate(proportion = ifelse(
    day_of_week == "Total",
    mean(store_sales_data$promo == "Yes" & store_sales_data$holiday == "Yes"),
    proportion
  ))%>%
  ggplot(aes(x = day_of_week, y=proportion)) +
  geom_bar(stat = "identity", fill = "red") +
  labs(
    title = "Proportion of Days with Promotion or Holiday",
    x = "Day of the Week",
    y = "Proportion"
  ) +
  theme_minimal()
```



```
proportion = mean(store_sales_data$promo == "Yes" & store_sales_data$holiday == "Yes")

cat("The proportion of days of both holidays and promo is:", proportion)
```

```
## The proportion of days of both holidays and promo is: 0.0288
```

The proportion of days with both holidays and promo, around the total days, is 2.88%. When sectioning it day by day, the proportion varies.

the proportion for day 0 is at around 3.25%, while it is slightly above 3% for day 1. for day 2, it is at slightly below 2.5%, while it is around 3.8% for day 3. it is around 3.25% for day 4, 2.75% for day 5, and 1.8% for day 6.

Fit a Poisson Regression Model

Now, let's try fitting a model to predict the sales count of a store, with the day of the week, promo, holiday, and store size as predictors.

```
# train_index <- createDataPartition(store_sales_data$sales_count, p = 0.8, list = FALSE)
# train_data <- store_sales_data[train_index, ]
# test_data <- store_sales_data[-train_index, ]

model_poisson <- glm(formula = sales_count ~ day_of_week + promo + holiday + store_size, family="poisso
summary(model_poisson)
```

```
##
## Call:
## glm(formula = sales_count ~ day_of_week + promo + holiday + store_size,
##     family = "poisson", data = store_sales_data)
```

```
##
## Coefficients:
##                   Estimate Std. Error  z value Pr(>|z|)    
## (Intercept)       2.986855   0.012142  245.993  < 2e-16 ***
## day_of_week1      0.060625   0.015432    3.929 8.55e-05 ***
## day_of_week2      0.126883   0.014958    8.482  < 2e-16 ***
## day_of_week3      0.165142   0.014575   11.330  < 2e-16 ***
## day_of_week4      0.191823   0.014597   13.141  < 2e-16 ***
## day_of_week5      0.270251   0.014227   18.996  < 2e-16 ***
## day_of_week6      0.315462   0.014331   22.012  < 2e-16 ***
## promoYes          0.410422   0.007820   52.485  < 2e-16 ***
## holidayYes       -0.330349   0.014945  -22.105  < 2e-16 ***
## store_sizemedium -0.697036   0.008311  -83.868  < 2e-16 ***
## store_sizesmall  -1.395325   0.011884 -117.416  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 25307.2  on 4999  degrees of freedom
## Residual deviance:  5134.7  on 4989  degrees of freedom
## AIC: 26509
##
## Number of Fisher Scoring iterations: 4
```

```r
coeffs <- coefficients(model_poisson)

cat(
  "The following model follows the formula:\n",
  coeffs["day_of_week1"], "*1st day of week + ",
  coeffs["day_of_week2"], "*2nd day of week + ",
  coeffs["day_of_week3"], "*3rd day of week + ",
  coeffs["day_of_week4"], "*4th day of week + ",
  coeffs["day_of_week5"], "*5th day of week + ",
  coeffs["day_of_week6"], "*6th day of week + ",
  coeffs["promoYes"], "*promo + ",
  coeffs["holidayYes"], "*holiday",
  coeffs["store_sizemedium"], "*medium store size + ",
  coeffs["store_sizesmall"], "*small store size +",
  coeffs["(Intercept)"]
)
```

```
## The following model follows the formula:
##  0.06062452 *1st day of week +  0.1268828 *2nd day of week +  0.1651417 *3rd day of week +  0.191823
```

So, for example, if a promotion happens, store sales are expected to rise by 0.4094046.

Another example is that the model takes large store sizes as the baseline. So if the store size is medium, the expected sales count decreases by 0.06989193, while a small store size decreases the expected value by 1.396902.

Assess model fit

To check the fit of the model, let's check for over-dispersion, which is when the deviance, divided by the degrees of freedom, is greater than 1.5.

```
deviance <- summary(model_poisson)$deviance
cat("The deviance of the model is:", deviance)
```

```
## The deviance of the model is: 5134.651
```

```
cat("\nWith deviance/df = ", deviance/3990, "< 1.5, over-dispersion is not present")
```

```
##
## With deviance/df =  1.28688 < 1.5, over-dispersion is not present
```

```
cat("\nThe p-value of the goodness of fit is",
pchisq(deviance(model_poisson), df = model_poisson$df.residual, lower.tail = FALSE)
)
```

```
##
## The p-value of the goodness of fit is 0.07341504
```

Given that the dispersion is less than 1.5, and that the chi-square goodness of fit of the deviance is > 0.05, we can say that our model fits our data quite well

Still, other models might be better. let's check the negative binomial model's AIC and compare it with our Poisson model.

```
neg_binom_model <- glm.nb(formula = sales_count ~ day_of_week + promo + holiday + store_size, data = st
```

```
## Warning in glm.nb(formula = sales_count ~ day_of_week + promo + holiday + :
## alternation limit reached
```

```
AIC(model_poisson, neg_binom_model)
```

```
##                 df      AIC
## model_poisson   11 26508.82
## neg_binom_model 12 26510.15
```

We can see that the poisson model has a slightly lower AIC score, meaning that it is the better model, at least compared to the negative binomial regression model.

Make Predictions

Let's now make predictions. We want to predict the expected sales for the following:

a medium store on a Monday with a promotion and no holiday, and a large store on a Sunday with no promotion and a holiday.

```
prediction_data <- data.frame(
  day_of_week = c("1", "0"),
  promo = c("Yes", "No"),
  holiday = c("No", "Yes"),
  store_size = c("medium", "large")
)

predicted_counts <- predict(model_poisson, newdata = prediction_data, type = "response")

cat("For the medium store on a monday with a promotion and no holiday, the predicted sales count is", p
```

```
## For the medium store on a monday with a promotion and no holiday, the predicted sales count is 15.81
```

```
cat("\n\n")
cat("For the large store on a Sunday with no promotion and a holiday, the predicted sales count is", pre
```

```
## For the large store on a Sunday with no promotion and a holiday, the predicted sales count is 14.246
```

For the medium store on a monday with a promotion and no holiday, the predicted sales count is 15.81354

For the large store on a Sunday with no promotion and a holiday, the predicted sales count is 14.24643

Of course, to be certain, we can round it down to 15 and 14, respectively.

Reflection

With a dispersion well below 1.5, as well as the P-values for the different predictors being less than 0.05, meaning that they are significant, I would say that the model that we have is a good model for predicting the store count of a given store at a given day, provided that we can properly categorize its size, and identify if it has any promotions going on.

Given this, it was found that every predictor: the day of the week, the store size, if there is a holiday, and if there is a promotion, are all very significant in the prediction. However, if I had to pick, it would either be the store size or the identification of a promotion.

However, implementing this model in the real world may be problematic, given that the dispersion and the fit is not perfect, as shown by the p-value of the goodness of fit being only a few points above our alpha of 0.05.