**Rossman Store Dataset EDA report**

General description of the dataset:

The Rossman store dataset is a historical sales data for 1,115 Rossmann stores. The task is to forecast the "Sales" column for the test set. Note that some stores in the dataset were temporarily closed for refurbishment.

**Files**

* **train.csv** - historical data including Sales
* **test.csv** - historical data excluding Sales
* **sample\_submission.csv** - a sample submission file in the correct format
* **store.csv** - supplemental information about the stores

**Data fields**  
Most of the fields are self-explanatory. The following are descriptions for those that aren't.

* **Id** - an Id that represents a (Store, Date) duple within the test set
* **Store** - a unique Id for each store
* **Sales** - the turnover for any given day (this is what you are predicting)
* **Customers** - the number of customers on a given day
* **Open** - an indicator for whether the store was open: 0 = closed, 1 = open
* **StateHoliday** - indicates a state holiday. Normally all stores, with few exceptions, are closed on state holidays. Note that all schools are closed on public holidays and weekends. a = public holiday, b = Easter holiday, c = Christmas, 0 = None
* **SchoolHoliday** - indicates if the (Store, Date) was affected by the closure of public schools
* **StoreType** - differentiates between 4 different store models: a, b, c, d
* **Assortment** - describes an assortment level: a = basic, b = extra, c = extended
* **CompetitionDistance** - distance in meters to the nearest competitor store
* **CompetitionOpenSince[Month/Year]** - gives the approximate year and month of the time the nearest competitor was opened
* **Promo** - indicates whether a store is running a promo on that day
* **Promo2** - Promo2 is a continuing and consecutive promotion for some stores: 0 = store is not participating, 1 = store is participating
* **Promo2Since[Year/Week]** - describes the year and calendar week when the store started participating in Promo2
* **PromoInterval** - describes the consecutive intervals Promo2 is started, naming the months the promotion is started anew.

Insights on dataset preparation:

1. **Loading the data**

The data was saved into three pandas data frames store, train and test respectively.

1. **Formatting the data**

* The date column was separated into day, month and year to make visualizations more effective and help with model training.
* Added a sales per customer column by dividing the sales column by customer to see on average how much each customer spends in each store.
* Some sales values and / or customer values are zero thus the null values in the sales per customer column were filled out with 0.

1. **Outlier detection:**

* The outliers in the sales column is made up of about 2.62% of the total, thus seen as insignificant and might be important for model training.

1. **Merging the store data frame with the training data frame:**

* Created a new data frame that has both the train and store datasets joined by the Store column.
* There were many null values in the store dataset specifically the ‘CompetitionDistance’, ‘CompetitionOpenSinceMonth’, ‘CompetitionOpenSinceYear’, Promo2SinceWeek’, ‘Promo2SinceYear’ and ‘PromoInterval’, each column was cleaned with respect to their significance in the data.
* The PromoInterval column’s null values were changed to zeros to signify that the store was not running a promo in a specific interval.
* The null values in the CompetitionDistance were replaced by a very high value (i.e. 200000). To signify that any competitive stores were too far.
* Stores with no competition and no promo their 'CompetitionOpenSinceMonth', 'CompetitionOpenSinceYear', 'Promo2SinceWeek', 'Promo2SinceYear' were placed with zeros.
* The remaining null values in the 'CompetitionOpenSinceMonth', 'CompetitionOpenSinceYear','Promo2SinceWeek', 'Promo2SinceYear' were imputed using the ‘IterativeImputer’ function.
* The PromoInterval column was replaced with a one hot encoding of each month with a 1 if they were running a promo and 0 if they were not, the purpose of this is to change to column into a numerical one and to help with model training by increasing the number of features.
* Fixing the StateHoliday column as it had some numerical values as strings and it will be converted using the one hot encoding method after the general visualization steps.

**General plots and Visualizations**

**Sales distribution by year:**

A pie chart with different colors

AI-generated content may be incorrect.

sales being highest in 2013 and lowest in 2015, showing a downward trend for yearly sales.

Sales in 2013 were equal to 39.2% of total sales with sales totaling over 2.3 billion.

Sales in 2014 were equal to 37.1% of total sales with sales totaling over 2.1 billion.

Sales in 2015 were equal to 23.7% of total sales with sales totaling over 1.3 billion.

**Monthly trend sales by year:**

A graph of a line graph

AI-generated content may be incorrect.

Sales fluctuate throughout the year, showing noticeable seasonal variations.

2013 (light line) and 2014 (purple line) follow a somewhat similar pattern, with a significant rise in December, possibly due to holiday shopping.

2015 (dark line) shows slightly higher sales in most months, suggesting potential overall growth compared to previous years.

There’s a dip around February in all years, indicating a consistent post-holiday slowdown in sales.

2015 had the least yearly sales yet that seems to be due to the lack of data points for the last 5 months rather than a dip in general sales as the overall sales for 2015 seems better than the other years in the same period.

**Total sales by month:**

A graph with blue lines

AI-generated content may be incorrect.

**Pie chart**

A colorful circle with numbers

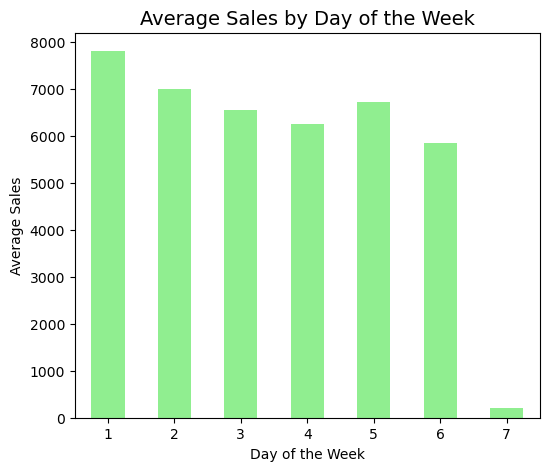
AI-generated content may be incorrect.

Sales peak around March and July, indicating periods of strong performance, possibly due to promotions, holidays, or seasonal demand.

Sales drop sharply from August to October, marking a low-performing period in the year.

December shows a partial recovery, which could correspond to end-of-year or holiday-related shopping activity.

**Average sales per day of the week:**

****

Monday (Day 1) has the highest average sales, suggesting strong customer activity at the start of the week.

Sales gradually decline toward the weekend, with Saturday (Day 6) showing a noticeable drop.

Sunday (Day 7) shows an extremely low average — this likely indicates that most stores are closed on Sundays, which is consistent with the Rossmann dataset’s Open column pattern.

**Daily sales trend:**

A graph showing a line

AI-generated content may be incorrect.

The peak sales appear to be around Day 2 and Day 3, reaching approximately $230 million and $225 million respectively, and again near Day 29/30 around $220 million.

the drops on Day 7, Day 14, Day 21, and the sharp drop on Day 31 could correspond to Sundays when stores might be closed or have reduced hours/traffic.

Day 31 has the biggest drop since not all months have 31 days

**Effect of Promotions on Average Sales:**

A graph with a bar and a green bar

AI-generated content may be incorrect.

Average sales increase noticeably when there is a promo being almost double that of the no promo bar.

**Total Sales by Store Type:**

**A graph of a sales report

AI-generated content may be incorrect.**

We see that store type a has the greatest number of sales and customers while store type b has the least

Store type b has the least number of datapoints in general.

**The impact of the state holidays on average sales:**

A graph of sales by state holiday type

AI-generated content may be incorrect.

- `0` → No holiday (a regular business day).

- `a` → Public holiday (e.g., national or state-level holidays).

- `b`→ Easter holiday (a special holiday period around Easter).

- `c`→ Christmas holiday (the holiday period around Christmas).

most of the sales come in normal non holidays which is logical as most shops are closed but it shows that on public holidays it sells the most followed by Easter and lastly Christmas.

**The number of open and closed days for each store type:**

**A graph with blue and orange bars

AI-generated content may be incorrect.**

We can see that store ‘b’ had the least number of open days and is far less represented in the dataset compared to the other stores followed closely by store ’c’.

Store a has the most number of open and closed days with the number of closed days for stores ‘a’ surpassing the number of open days for store ‘b’.

**Correlation Heatmap:**

A screenshot of a graph

AI-generated content may be incorrect.

We find that the number of customers correlates heavily with the sales.

More correlation will be made when the data is fully processed for modelling.

**Number of stores open by year:**

**A colorful circle with a few different colored parts

AI-generated content may be incorrect.**

stores have been opening less each year with the highest number of stores opened in 2013 and lowest being in 2015.

Although this is likely due to the smaller number of data points in the year 2015 as only the first 7 months are recorded for it.

**Number of stores open each month:**

A graph of blue and orange bars

AI-generated content may be incorrect.

Stores open on average consistently each month except for the last 5 months where there is a notable drop in general datapoints.

**Sales timeline for each store type:**

A screenshot of a graph

AI-generated content may be incorrect.

We find that each store type is affected by trends like the end of the year because each store type has a spike in sales.

Store type D isn’t represented from the period of July 2024 to January 2025.

**Testing for stationarity for each store type using the Dickey-Fuller test:**

Store type A:

A graph showing a graph of a graph

AI-generated content may be incorrect.

Results of Dickey-Fuller test:

ADF Statistics: -6.218237

P-value: 0.000000

Critical Values:

1% -3.4374778690219956

5% -2.864686684217556

10% -2.5684454926748583

Store type B

A graph of a graph of a graph

AI-generated content may be incorrect.

Results of Dickey-Fuller test:

ADF Statistics: -5.660918

P-value: 0.000001

Critical Values:

1% -3.437485646962348

5% -2.8646901138095378

10% -2.568447319459459

Store type C:

A graph showing the growth of the stock market

AI-generated content may be incorrect.

Results of Dickey-Fuller test:

ADF Statistics: -4.374784

P-value: 0.000329

Critical Values:

1% -3.4374778690219956

5% -2.864686684217556

10% -2.5684454926748583

store type D:

A graph with red and blue lines

AI-generated content may be incorrect.

Results of Dickey-Fuller test:

ADF Statistics: -6.237461

P-value: 0.000000

Critical Values:

1% -3.4392539652094154

5% -2.86546960465041

10% -2.5688625527782327

We find that the dataset is stationary and does not change significantly thus there is no reason to transform the dataset for models like SARIMA.

**Trends and seasonality analysis:**

Store type A:

A graph of green and blue lines

AI-generated content may be incorrect.

Store type B:

A graph of a graph of a graph

AI-generated content may be incorrect.

Store type C:

A graph of green and blue lines

AI-generated content may be incorrect.

Store type D:

A close-up of a graph

AI-generated content may be incorrect.

We find that the data does have obvious trend and seasonality, So, we'll use forecasting models that take both factors into consideration.