**Machine Learning 2 Project Report:**

**Rossmann Store Sales Forecasting**

PREPARED BY:

Adil Kumar, Ivy Zhou,

Jake Arendsen, Zihan Wang

**TABLE OF CONTENTS**

[**1. Business Understanding**](#_31ju5891cx1) **3**

[**2. Data Understanding**](#_x84iead8nzqj) **4**

[**3. Data Preparation**](#_tjvystcndj2c) **9**

[**4. Modeling**](#_pypjq0cg6584) **11**

[**5. Evaluation**](#_wnyizh62r2jq) **14**

[**6. Deployment**](#_r0pmmw3r5o4q) **16**

[**7. Source**](#_r0pmmw3r5o4q) **List 18**

**8. Appendix 18**

# 1.Business Understanding

Rossmann is a German drug store chain with over 3,790 stores spread throughout six European countries. Due to this incredible foundation Rossmann has developed, it’s simply impossible for the company to micro-manage all of their stores, so they leave many of the day-to-day or otherwise regular responsibilities to store managers. Currently, one of these responsibilities includes projecting daily sales for up to six weeks in advance. However, given analytics may not be a strength for many store managers, Rossmann receives all kinds of wildly inaccurate predictions based upon numerous different factors. Consequently, Rossmann has decided to begin handling these predictions on their own to increase both accuracy of the results and consistency of the errors.

Rossmann would like to take advantage of all their business knowledge to best build models to predict daily sales at the individual store level. As a drug store chain, they face both unique problems and advantages in projecting their sales. European drug stores see much more casual shopping than their American counterparts, resulting in better overall sales but also increased variability. Additionally, seasonal illness which increases medicinal purchases can still have large effects on sales. Due to more casual shopping, these drug stores also see increased competition when compared to their American counterparts. In particular, within Germany, Rossmann faces great competition from dm-drogerie markt, the leading drug store chain in Germany with almost 2,000 stores in the country. This means that pricing, promotions and other strategic decisions will likely have an increased effect on sales, as they could contribute to taking sales directly from the competition.

In this case, even a good predictive model will likely be an improvement over Rossmann’s current forecasting system. More accurate sales forecasting provides Rossmann with numerous advantages. Given strategic decisions are made with forecasts as a key piece of information, having more accurate forecasts is an obvious advantage. Using unreliable forecasts to make decisions on promotions, opening or closing stores, inventory management, or personnel decisions, all very possible use cases for sales forecasting, could result in suboptimal decisions, which could eventually lead to a demise for the company. Meanwhile, proper decisions could lead to stronger and stronger profits, leading to expansion and strengthening position in the market.

# 2. Data Understanding

We have two data sets from Rossmann to help build our predictions: sales data and store data. Each row of sales data is a record for the sales of a particular store on a particular day. The dataset includes the following features:

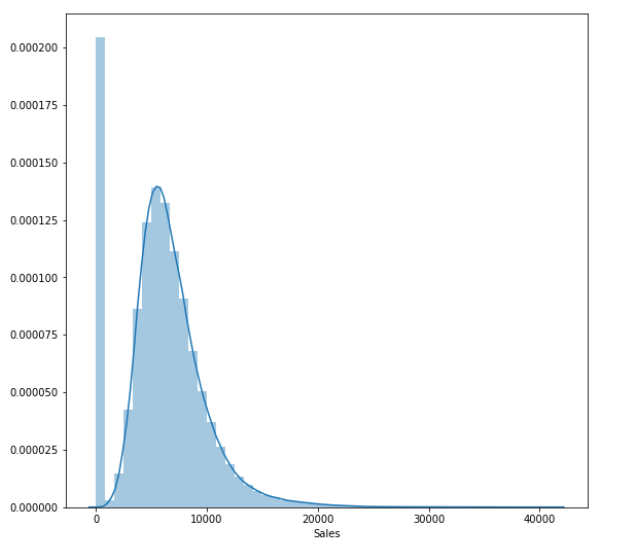
* Store: The ID number of the store
* DayOfWeek: The day of the week (Monday, Tuesday, etc.)
* Date: The exact data
* Open: A dummy variable indicating whether or not the store was open that particular day.
* Promo: A dummy indicating whether or not the store was running a promotion on the particular day.
* StateHoliday: A dummy indicating if the day was a state holiday. In particular, b indicates Easter, c indicates Christmas, and a indicates any other state holiday.
* SchoolHoliday: A dummy indicating whether Schools are closed for any reason except the weekend (e.g. breaks, holidays, natural disasters, etc.). The value remains 0 on the weekends.

We also have access to store data, which contains records involving basic information for every store. The features in this dataset are:

* Store: The ID number of the store, matching the same variable from the sales dataset.
* StoreType: A categorical variable indicating which of the four store types (a, b, c, or d) a store is. We don’t know what differentiates the four types of stores.
* Assortment: A categorical variable indicating the assortment strategy of the store. Here a stands for basic, b stands for extra, and c stands for extended. We assume this implies c carries more products than b, which carries more products than a, but otherwise we don’t exactly know what differentiates here or what particular products are carried by one and not the other.
* Competition Distance: Distance in meters to the closest competing store as defined by Rossmann
* Competition Open Since Month: Month in which the competing store opened
* Competition Open Since Year: Year in which the competing store opened
* Promo2: A dummy indicating whether or not the store participates in Promo2
* Promo2 SinceWeek: Month in which the store started participating in Promo2, if applicable. Otherwise, the value is null.
* Promo2 SinceYear: Year in which the store started participating in Promo2, if applicable. Otherwise, the value is null.
* Promo Interval: The months in which Promo2 is started anew at the store, if applicable. Otherwise, the value is null.

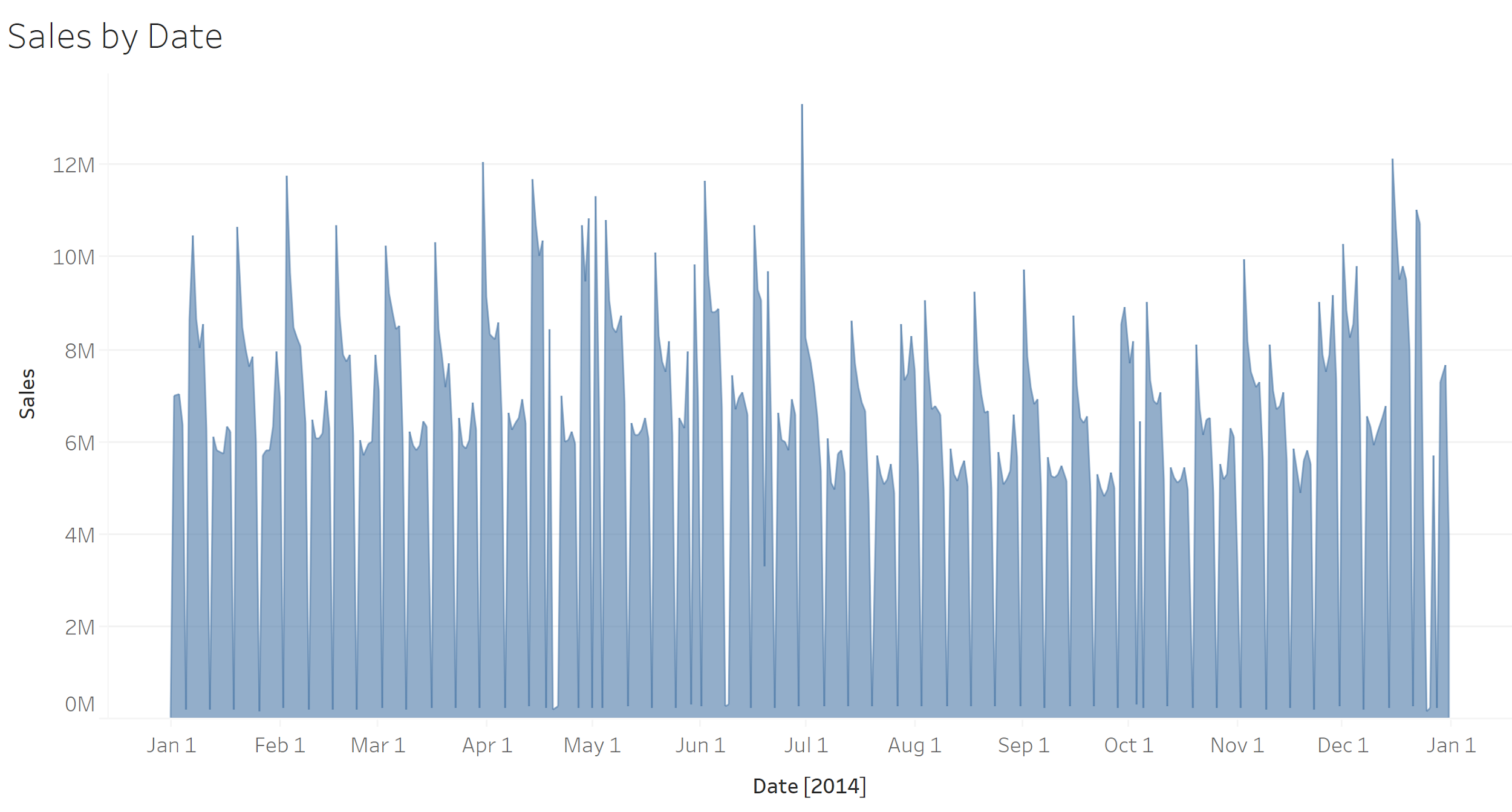
Given these data sets, we took a thorough look at the distribution of numeric variables including Sales, Customers, CompetitionDistance, Promo2 days since, and Competition Days Since. One common thing among these variable distributions is that all of them are heavily right skewed with a long right tail. And such skewness resulted from few extremely large outliers. We can take a good example from the target variable Sales in which, even though with an average below 10,000, few records amounted to over 40,000, dragging the average far away from the median. And the same situation applies to the distributions of other numeric variables as well.

We first examine the records with sales values over 40,000 since we need to figure out whether they are outliers or legit values. We found that the records with outstanding sales also are consistent with records with a large number of customers, especially on promotion day. This explains the high sales value. So we could take these records as rare scenarios that the stores are holding promotion events or holiday events and bring a large number of customers and boost sales accordingly. So we did not exclude any large values as outliers in our dataset.



*Figure 1: Histogram of Sales per day per store*

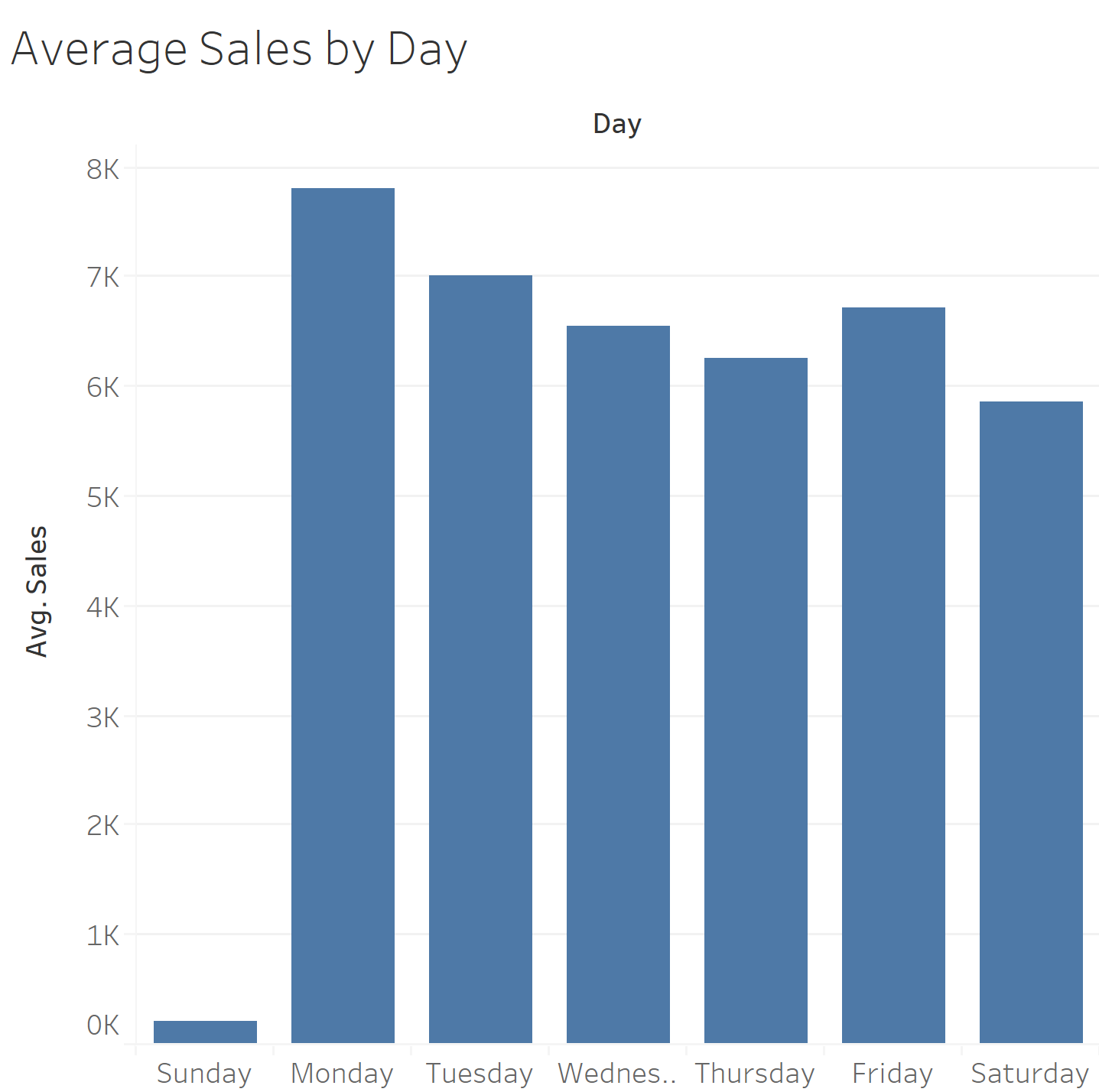
At this point, we wanted to build some visualizations to help visualize the volatility in sales and attempt to understand what could potentially cause it. The first thing we did was simply plot sales across all stores by date for the year of 2014.



*Figure 2: Sales by Date in 2014 for Rossmann*

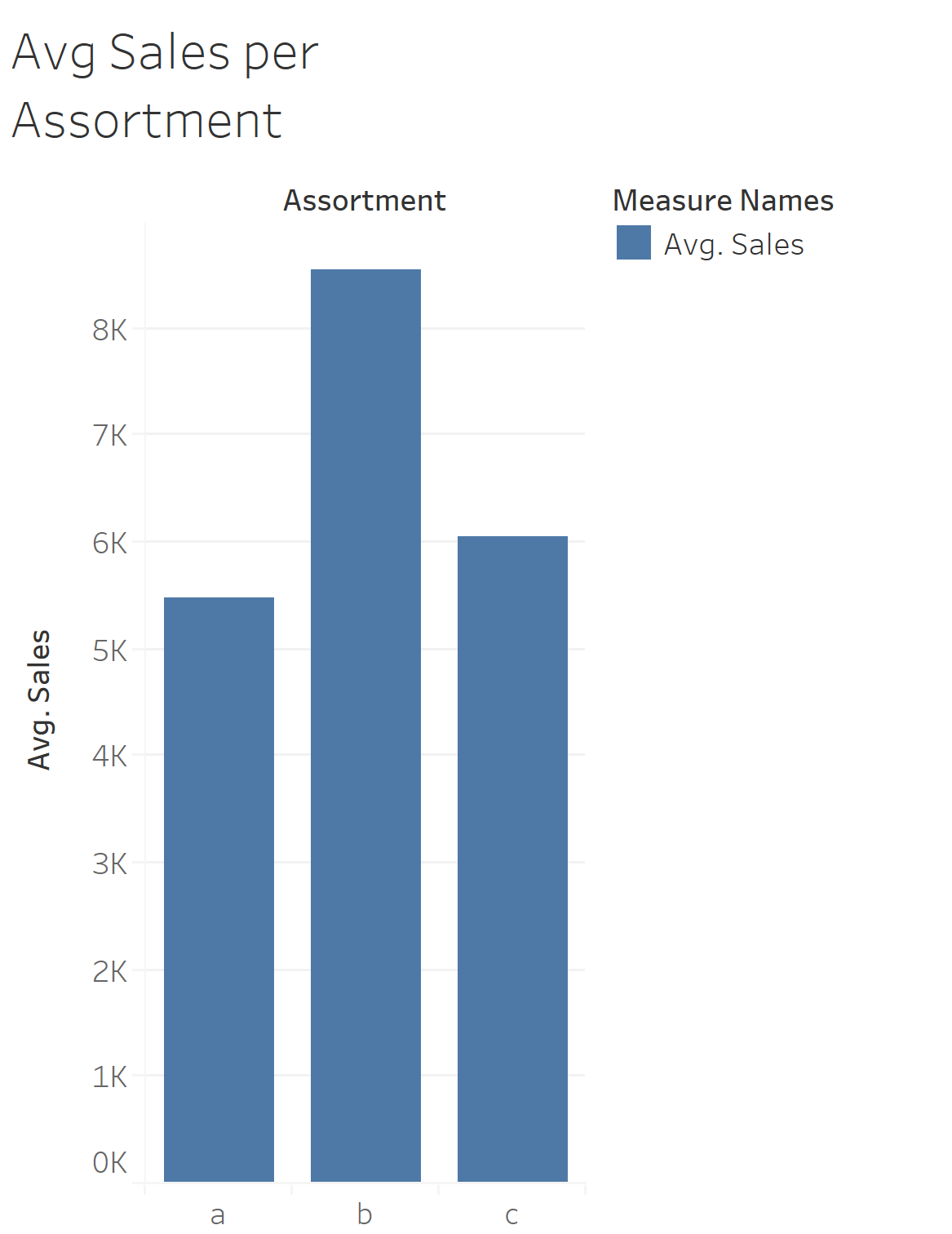
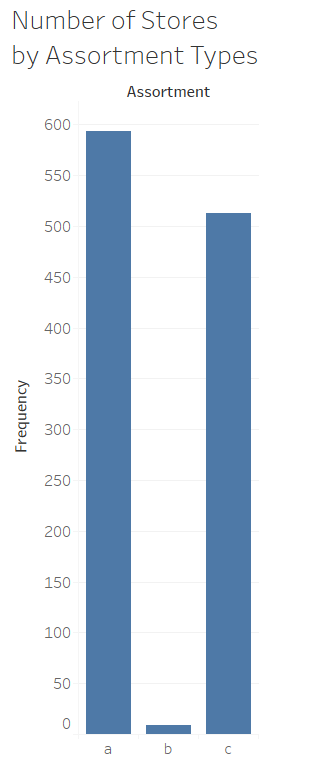
Here, we see the volatility that can make sales so challenging to predict. Fortunately, we can explain at least some of this volatility using our business understanding. We see a short, severe dip in sales repeating frequently in the dataset, which occurs because most Rossmann stores are closed on Sunday so Rossmann generates almost no sales on Sundays. We also see three longer dips, easily noticeable due to the white gaps they cause in the chart. These dips, which occur in late April, early June, and late December, are indicative of notable holidays (Easter, Whit Monday, and Christmas, respectively). This shows to us that holidays clearly have a large, significant impact on sales which we must account for within our model. However, the within week variation follows no obvious or consistent pattern, making it harder to try to understand.

To attempt to see if there is any regular variation within the week, we also plotted average sales by day, as shown below.



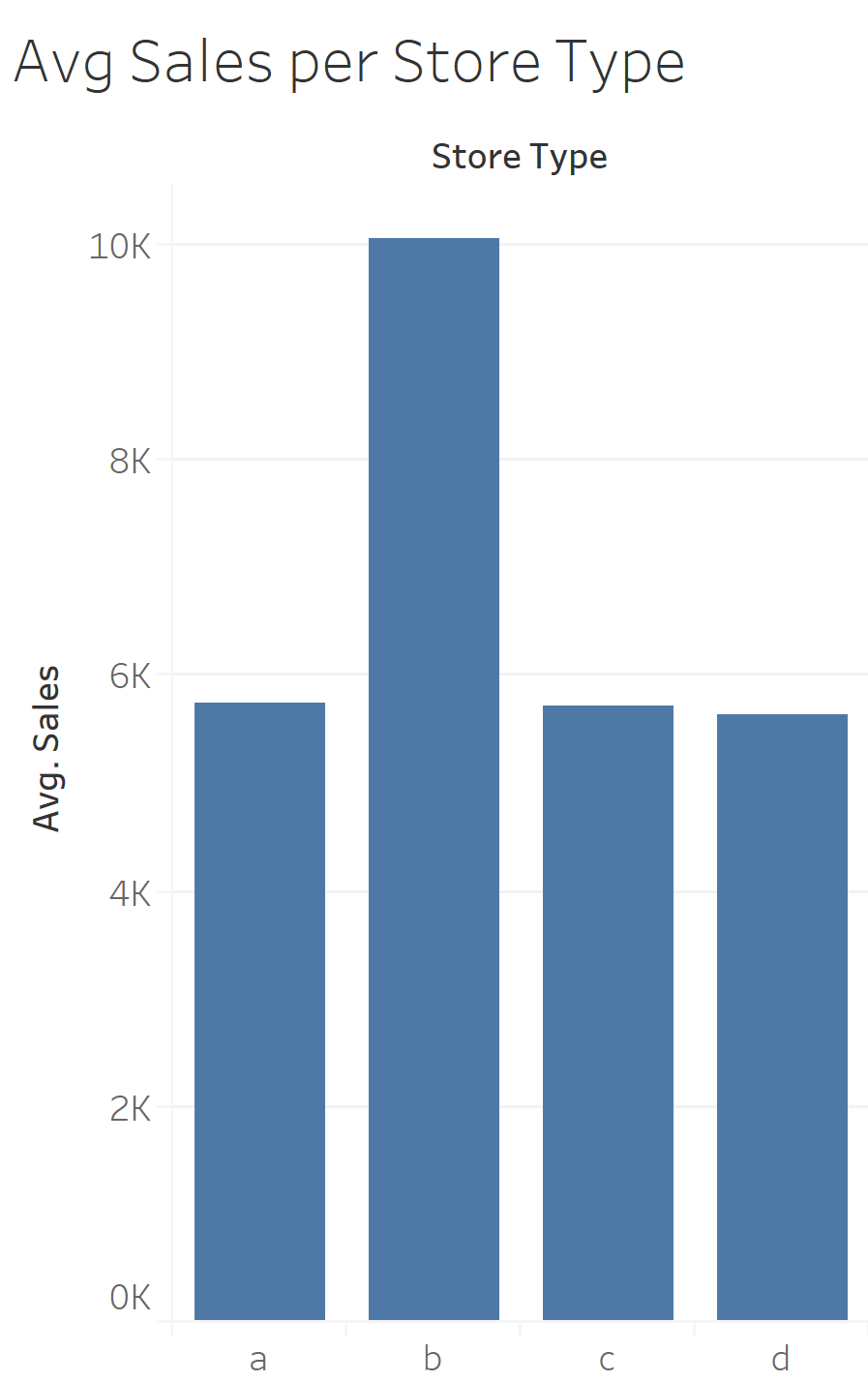
*Figure 3: Average Sales by Day of Week across all stores*

Here, we do see some kind of overall consistent pattern. Obviously, we have low sales on Sunday due to stores closing, which also likely has some effect on Monday’s increased sales. We then see sales slowly decrease throughout the week except for Friday, when people likely do some extra shopping to avoid having to shop on the weekend. However, our first chart does tell us that this pattern is no exact rule, as it’s broken with consistency when viewing daily sales. Finally, we wanted to see if different types of stores or store assortments resulted in different average sales. Both of those charts are shown below.

**

*Figure 4: Average Sales and Counts for each store assortment*

We see more sales for the “extra” assortment, though it’s also worth noting it’s by far the least frequent assortment, accounting for less than 1% of stores.



*Figure 5: Sales by Store Type*

We see fairly consistent sales for each store type with the exception of type B, which sees almost $4,000 more in average sales, which is about 67% more than the other store types.

# 3. Data Preparation

The two datasets we have are the sales data, which records over 1 million pieces of transaction details in 1115 Rossmann drug stores in a 3-year time frame, and the store data, which provides information about competition and promotion for each of these stores. First, we joined the datasets by Store IDs into one combined table that presents transaction and store details at the same time. Our next step was to check if there are any missing values in both datasets. Though we have no missing values in the sales data, we do have 6 columns with missing values in store data, primarily in competition and promotion columns. After checking the size of the missing values, it turned out that nearly one third of the stores have no competition details and nearly half of the stores have no promotion details. We did have to come up with crafty solutions to solve both of these problems. For the promotion columns, we noticed that only the stores participating in Promo2 had values for the other promotional information. Given that all of this data would be treated as factors, it was safe to simply set these values to 0 so it could act as its own factor. However, the competition data was not as simple. We found that three stores lacked information on a closest competitor.We could either drop these columns or make an educated assumption about them; we chose to assume that this meant there was no competition for these stores in a reasonable range, and as such set their value for competitor distance to the maximum value for competitor distance. Additionally, we saw plenty of columns that lacked a time range the competitor has been in business. We again chose to make an assumption here, this time that the stores with a competition distance but no date have been open for a long time, to the point where records of the store opening are hard to find. Therefore, we will assign them to the oldest opening data in the data and add a dummy to indicate there was no original date for them.

Helped by the visualizations, we understood that time has a significant effect on sales as well. Therefore, we generated a few variables to capture this effect. The first improvement we made was to extract months and years instead of just using one sale date column. Another variable that we believe would be helpful to our analysis is the week number of month (1,2,3,4,5), capturing trends within a month more precisely. The last time-series element we would be eager to include is the period of competition and promotion since it is reasonable to assume that a longer competition period decreases the sales while a longer promotion period increases the sales. Both periods were calculated by taking the number of days between the “CompetitionSince” and “Promo2Since” dates and the base date “2015-9-1”, the date since the Kaggle competition for Rossman store Sale prediction was live.

One last step before modeling was to encode categorical variables into dummy variables for the sake of certain modeling methods. Apart from those original dummy variables such as “Promo2” or “SchoolHoliday” in the dataset, we also encoded “DaysOfWeek”, “StateHoliday”, “StoreType”, “Assortment”, “PromoInterval” and “SaleWeek”. Given a time frame of 3 years, we split it by taking the records from January 2013 to December 2014 into the training set while January to July 2015 were grouped into the testing set. We do this so we can test on a timeframe not already seen by the training, which allows us to see how well we capture seasonality with our model as well as to make sure the model is suitable for a real world deployment scenario.

# 4. Modeling

When it came to model building, we encountered one of our first major obstacles with a primary step: feature selection. When reviewing correlations, we noticed that the number of customers and sales were very highly correlated; therefore, we obviously want to account for customer volume in our model. The problem, however, is this is very much so a case of data leakage. When the actual time for prediction came, we would not know how many customers will come to the store, so we cannot use that to predict sales for the stores. There’s a few potential workarounds we debated as a team. One option is to simply drop the variable, but this is obviously less than ideal given all the importance of customer volume. Another option is to use a reactive measure of customer volume, maybe the prior day’s volume, or maybe the average over the last week or month, or the average for that day of the week are all options. However, this would be a highly reactionary measure, which might hinder our predictive model attempts to be proactive. Therefore, the best option to us seemed to be to attempt to predict customer volume. This would allow us to continue to utilize this variable in a proactive way, which we would later confirm in modeling is indeed the best way to go about this. We do encounter some risk in this process, however. Stacking models like this makes it harder to determine potential problems with your models and can certainly mask effects of variables by having them affect both customers and sales. However, given the improvement the variable makes for the model, we consider it to be a risk worth taking.

This actually means we needed to create two models for our predictions. We built a model to predict customers, which then was utilized as a variable to predict sales. For each model, we built numerous different models to attempt to best capture the variation we saw. To best do that, we first need to make adjustments with respect to the skew that we mentioned earlier. We experimented with log, square roots, and cube roots for all the skewed variables, using histograms, skewness, and qq plots to evaluate the effect of the transformations. Ultimately, we decided to use logs for competitor distance and square roots for the number of customers and promo days.

In terms of model building, we started with our comfortable basics of kNN, linear regression, and decision trees. These all proved to be good but not great. We then proceeded to experiment with more models as necessary, trying things like gradient boosting and random forests. As necessary, we experimented with forwards and backwards selection, standardization of variables for kNN, and usage of transformed variables in appropriate linear models. We optimized all of our models through the use of grid search, optimizing multiple parameters on each model using root mean squared error (RMSE) as the measure to be optimized. Additionally, all models were trained and validated using 5 fold-cross validation, as that gives more reliable results than simply splitting into training and testing sets. The results of both sets of models are shown below.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Modeling Approach | Std Data | Transform | Grid Search | Stepwise | MAE | RMSE |
| Linear Regression | - | - | - | - | 194.20 | 288.12 |
| Linear Regression | Y | - | - | - | 624.19 | 770.83 |
| Linear Regression | - | Y | - | - | 193.44 | 282.78 |
| Linear Regression | - | - | - | Y | 194.20 | 288.12 |
| Linear Regression | - | Y | - | Y | 192.65 | 280.92 |
| K Nearest Neighbors | Y | - | Y | - | 91.87 | 160.45 |
| Decision Trees | - | - | Y | - | 186.36 | 287.35 |
| Random Forest | - | - | Y | - | 147.39 | 159.39 |
| Gradient Boosting | - | - | Y | - | 61.37 | 101.12 |
| Gradient Boosting | **-** | **-** | **Y** | **Y** | **58.3** | **97.59** |
| Ridge/Lasso | - | - | Y | - | 194.18 | 288.10 |
| Elastic net | - | - | Y | - | 194.06 | 288.06 |

*Figure 6: Customer Predictions Model Result*

We’d expect Rossmann to find a model like this very useful. Having sales projected by stores was the goal and is what they’re somewhat used to from what the managers currently do. However, these projections should both be more accurate and more consistent than prior projections the company has utilized. That would make any decisions based on these projections more accurate than they’ve been previously and allow Rossmann more confidence in their data-based decisions.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Modeling Approach | Std Data | Transform | Grid Search | Stepwise | MAE | RMSE |
| Linear Regression | - | - | - | - | 867.73 | 1217.34 |
| Linear Regression | Y | - | - | - | 868.2 | 1223.92 |
| Linear Regression | - | Y | - | - | 849.12 | 1161.91 |
| Linear Regression | - | - | - | Y | 918.89 | 1308.57 |
| Linear Regression | - | Y | - | Y | 909.29 | 1207.93 |
| K Nearest Neighbors | Y | - | Y | - | 588.55 | 941.73 |
| Decision Trees | - | - | Y | - | 1035.94 | 1547.48 |
| Random Forest | - | - | Y | - | 802.57 | 1044.39 |
| Gradient Boosting | - | - | Y | - | 312.96 | 493.72 |
| Gradient Boosting | **-** | **-** | **Y** | **Y** | **312.64** | **494.43** |
| Ridge/Lasso | - | Y | Y | - | 848.91 | 1161.97 |
| Elastic net | - | Y | Y | - | 846.73 | 1162.67 |

*Figure 7: Sales Prediction Model Results*

# 5. Evaluation

We used RMSE to evaluate our models. For the purpose of regression models like ours, RMSE and mean absolute error (MAE) are the most common and popular means of evaluating models. We’re choosing to use RMSE because, when compared to MAE, RMSE disproportionately punishes more severe errors. In our case, we prefer this effect. Given the odds of projecting sales exactly are so minimal, we’d much rather be frequently $5-$10 more off than occasionally hundreds to thousands of dollars off. The more severe mistakes can be potentially disastrous to business decisions, so we’d like to avoid those at all costs. The RMSE is lowest for the gradient boosted model with grid search and stepwise feature selection. However, since stepwise drops important variables like store assortment and promotion we decide to not use this restricted variable set since the performance improvement is not big enough and these variables are important in capturing sales variations. Thus, Gradient boosting with gridsearch for parameter optimization is the model we recommend Rossmann utilize to forecast their sales.

Unfortunately, an exact ROI for sales forecasting is challenging to calculate. Industry thought is that forecasting can cut inventory costs by 10-30% by properly managing safety stock and distribution tactics (Hughes). There’s an additional efficiency gain by allowing managers to reutilize time to tackle other problems. However, assuming the managers are salaried employees, this doesn’t actually result in a decrease in wages, assuming managers don’t take up work that would otherwise be done by hourly employees. There’s an obvious gain in operational efficiency across the organization that can be hard to quantify without knowing the costs of the prior lack of optimization. Additionally, there’s an obvious sense of consistency now with the predictions that will be quite a time save for operational decision making. Sadly, none of this is easy to quantify, but a savings of 10-30% alone in inventory costs at a minimal investment cost should justify the expense itself. At that point, any of the other potential gains can be considered an added bonus.

Should Rossmann want to try and visualize improvements, they could do so by slowly implementing the model. Hypothetically, half the stores could use the model while the stores could continue having managers forecast upcoming sales. Then, these projections could be compared against actual sales and costs associated with inventory decisions made based upon projections could be calculated. This would allow Rossmann to see tangible value provided to them by better sales projections and better inventory management decisions made due to said projections.

# 6. Deployment

The value of sales prediction lies in several aspects. First, forecasting sales will help the client set the sales target quota at the store level. Different stores will have different sales target quota according to the predicted sales. Given sales prediction and trend of customer number, store managers could better shape sales and promotion strategy. Next, sales forecasting would help the client to understand regional difference and twick operating strategy ahead of time. Since different regions acted differently on different events and holidays, predicted sales would be a good chance to understand the customer group and cater their need to a higher level.

We also have an interesting thought about one way to deploy the model. It’s possible that the client could use the model result to plan events. That is, the client would be able to find the best stores for holding sales events by taking advantage of the model prediction. Sales condition varies from store to store, from region to region. A lot of times the sales is determined by the combinations of all factors including location, competitor, customer group,etc. For large drug store chains like Rossmann, they need to decide which stores are the best to hold sales events in order to boost sales. With the model, the client would be able to practice virtual sales events on certain stores to see how well the event acts by the model result. The actual cost of holding an event could be avoided by holding virtual ones. This way, the client could optimize revenue by selecting the stores which brings the most lift on the revenue using the model.

There are some potential problems and risks involved when deploying the model. In terms of risk, over forecasting and under forecasting will cause negative effects to the client. Over forecasting will lead to a potential failure of the sales team in meeting the targets and the client might not be able to meet seasonal goals for revenue. Under forecasting will make the sales targets very easy to achieve and cause loss for the company. To mitigate the risk of either over forecasting or under forecasting, we could deliver a range for sales prediction which is within a certain level of confidence (like 95%) instead of a single number. In addition, one problem that brought our attention is that the model could not predict sales for new stores. Since the model is built on historical sales data, it would not be able to predict sales for any newly opened stores due to the lack of data. To solve this problem, a practical way is to run a similarity check on the new store, find the most similar stores and take advantage of their data.

Thankfully, this is a task with fairly minimal ethical concern. There is no personal information used anywhere in the model, so the fear of using protected information is pretty minimal. Additionally, very little information about the competition is used, and all the information being used is very public knowledge. Finally, the model is not intended to manipulate or trick consumers in any way. We simply want to forecast sales to make wise business decisions. We would also potentially use it to see what effects promotions might have towards consumer behavior, but this lacks ethical concerns as well. As with any data mining project, ethical concerns should be continuously monitored and considered, but we feel confident saying this project lacks any notable ethical risks.

**7. Source List**

1. Hughes, J. (n.d.). Determining the ROI of Forecasting Systems. Retrieved April 12, 2020, from <https://www.silvon.com/blog/determining-roi-forecasting-systems/>
2. Prof Vilma Todri for Python Codes on Grid search and Regression Models
3. Prof Panos Adomopoulos for guidance on evaluating data cleaning and modeling approaches
4. XGboost code reference <https://www.kaggle.com/jayatou/xgbregressor-with-gridsearchcv>
5. RFE code reference <https://towardsdatascience.com/feature-selection-in-python-recursive-feature-elimination-19f1c39b8d15>
6. Week number calculation <https://stackoverflow.com/questions/47986121/week-number-of-the-month>

**8. Appendix**

As requested, we will list what each team member contributed to this individual project.

Adil: Almost all of the coding, Presenting the modeling slides. Inputs on technical details of PPT and report

Ivy: Almost the entire PPT deck, presenting data understanding and data preparation slides, some writing of report (mostly deployment). Inputs on Data cleaning and EDA.

Jake: Writing majority of Business Understanding, Data Understanding, Modeling, and Evaluation sections, presenting business understanding slides, editing of report, project proposal. Inputs on modeling approaches and model selection.

Zihan: Writing Data preparation section, some contributions to almost all areas of report, create visualizations, present visualization slides