Team Michael Jackson

Data Storm 3.0 Report

Kaggle Username: ds22-98
Display Name: Michael Jackson

Navin Thamindu Chandrasiri
Shamila Mihiraj Wijekoon
Kasun Kavishka Herath

Git Repo: https://github.com/ShamilaWije/DataStorm3

Lowest Total MAPE score: 25.053030781872195

1.0 The Approach

As the storming round started, our main problem was to have a clear insight about the dataset in a bird's eye overlook. In order to achieve this, we used *Tableau Public* to insert the given data sets and get a visualized graph of the dataset.

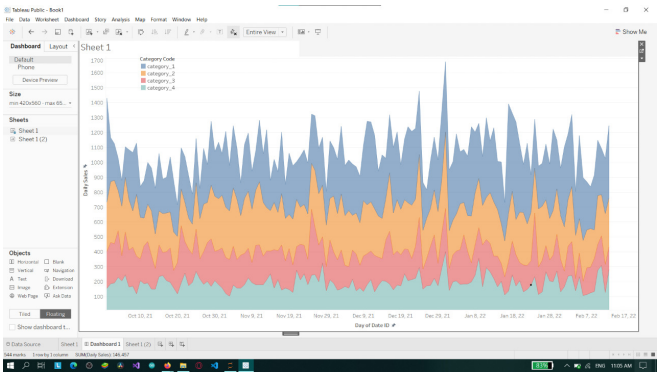


Figure 1 : Visualized datasets in Tableau Public

This view enabled us to understand that there are 4 subcategories of datasets are at play and the fact these data spans throughout months from October 2021 to February 2022. As we progressed through the datasets, we understood that the *validation data* are presented in a weekly manner, but the *train data* is given with dates.

	CategoryCode	ItemCode	Week	WeeklySales
2	category_2	1044502	w1	11
3	category_2	1105009	w1	11
4	category_2	913561	w4	5
5	category_1	1048975	w4	30

Figure 2: Validation Data

```
1  CategoryCode,ItemCode,DateID,DailySales
2  category_2,117610,11/6/2021,7
3  category_4,836584,11/18/2021,16
4  category_1,370195,1/24/2022,6
5  category_2,172582,10/30/2021,5
6  category_2,1006009,10/30/2021,5
7  category_2,903976,1/6/2022,1
```

Figure 3: Train Data

Therefore, we must find a way to make a custom data set categorized according to weeks by using the given Train data to create a model to predict the "Reservation Status" of each customer as we will be using *the linear regression* method to predict. To create this custom data set, we used *Colab*.

1.1 Cleaning the dataset

For the cleaning process we used python with Jupyter Notebook. To make the filtering processes much easier, we used pandas, NumPy, seaborn and matplotlib libraries. Initially, csv library was tested for the cleaning but later we found out it is not the best solution for the problem.

As we identified earlier, there are 4 catagories of products and it seems we should train 4 different models to increase accuracy. But the more we progressed analyzing the data by plotting correlation maps (with heatmaps), it occurred to us that 4 models would not be enough to increase accuracy. Therefore, we decided to train 194 different models because there are 194 different items in the dataset.

```
cor = df.corr()
sns.heatmap(cor, annot=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa8c33da410>
```

Figure 4: correlation maps

Now, our goal was to somehow categorize the data into weekly data instead of day specific rows –separated by item codes.

```
fdf = df.pivot_table(index='Week', values='Sales', columns ='ItemCode',fill_value=0, aggfunc=sum).reset_index()
fdf
```

ItemCode	Week	3418	3427	7666	9925	16936	17287	17296	20824	23200	...	1098502	1101553	1101562	1101571	1101661	1101769	1103056	1105009	1105018	1105027
0	1	42	40	119	9	25	16	274	144	7	...	8	11	9	7	22	3	0	0	0	0
1	2	41	20	196	11	40	6	438	123	33	...	7	10	8	18	26	0	8	0	0	0
2	3	41	48	185	13	31	10	276	112	7	...	4	5	10	7	14	0	12	5	0	0
3	4	44	54	94	5	21	21	485	103	64	...	9	3	19	6	41	18	28	5	3	3
4	5	46	28	15	11	43	21	578	163	80	...	3	5	4	7	23	20	27	10	6	11
5	6	43	36	10	11	7	12	643	284	61	...	9	9	13	3	14	22	20	17	11	10
6	7	49	52	30	13	34	20	540	140	30	...	3	5	9	8	45	17	17	17	4	7

Figure 5:Data categorized by item codes and weeks

This data table looks outstanding -so that we plotted the data into a graph to check whether the data can be used for linear regression models. As you can see the graph does not follow a direct line which means that this table is not usable for the model. We must go another round in fixing the data set.

```
plt.plot(fdf[3418])
```

```
[<matplotlib.lines.Line2D at 0x7feed615bc10>]
```

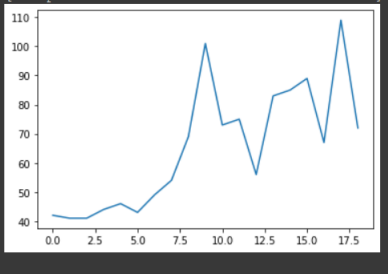


Figure 6: Plotted graph of the restructured table of an item

We wrote a new algorithm to organize the data in a Cumulative manner. This would ensure that this table is suitable for linear regression.

```
ffdf = pd.concat([fdf['Week'],fdf[com].cumsum()], axis=1)
ffdf
```

	Week	3418	3427	7666	9925	16936	17287	17296	20824	23200	...	1098502	1101553	1101562	1101571	1101661	1101769	1103056	1105009	1105018	1105027
0	1	42	40	119	9	25	16	274	144	7	...	8	11	9	7	22	3	0	0	0	0
1	2	83	60	315	20	65	22	712	267	40	...	15	21	17	25	48	3	8	0	0	0
2	3	124	108	500	33	96	32	988	379	47	...	19	26	27	32	62	3	20	5	0	0
3	4	168	162	594	38	117	53	1473	482	111	...	28	29	46	38	103	21	48	10	3	3
4	5	214	190	609	49	160	74	2051	645	191	...	31	34	50	45	126	41	75	20	9	14
5	6	257	226	619	60	167	86	2694	929	252	...	40	43	63	48	140	63	95	37	20	24

Figure 7: Cumulated Table

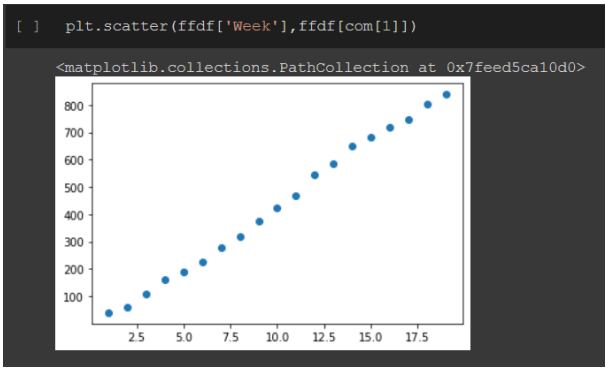


Figure 8: Graph of the cumulated data of an item

1.2 Making models

To get the predictions, we must initialize all the 194 models. We tried using an algorithm to automate the model creation process, but we came across an index error, so that we had to drop this method and think of a fast solution due to the limited time frame.

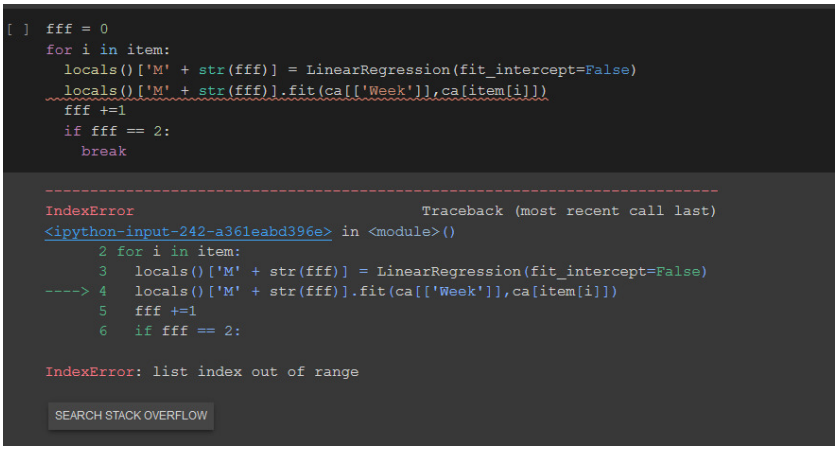


Figure 9: Index error

Our immediate idea was to hard code every model to save up time. We used another script to generate codes for the model creation. As programmers, we are not proud and happy about this approach, but at this brink of *final destination and desperation* we had to take a leap of faith and move along with this solution.

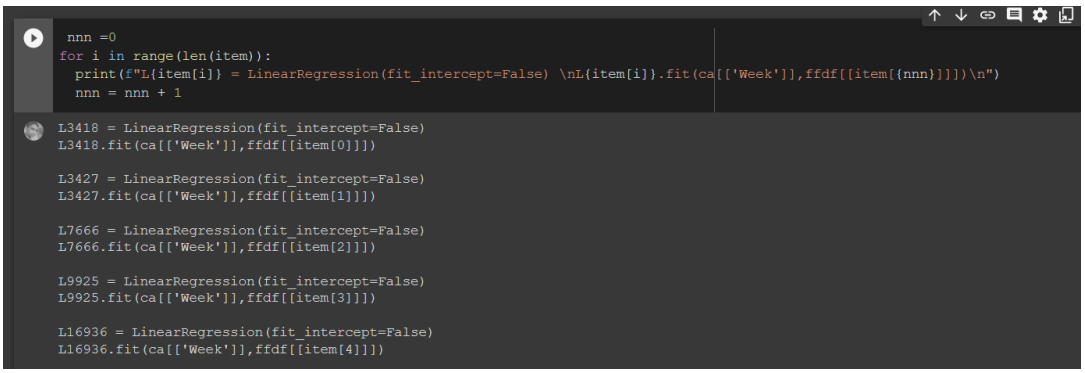


Figure 10: Generating hardcodes for model creation

2.0 Feature Engineering ideas

- Tableau Public used to visualize data beforehand to have a bird's eye view, -so that, it is easier to find ways to restructure data.
- Organize the data in a cumulative manner, -so that a certain table is suitable for linear regression.
- Automating the process of creating models, -so that a large number of models can be initiated with no hassle at all (failed)

3.0 Final Model

It is obvious by now that we use 194 different models. We are using the *sklearn* python library to train the models via *colab*. To be specific we are using sklearn's Linear Regression tool for this task.

According to analysis, we found that there is data worth 19 weeks. Before proceeding into the predictions, we must change W1, W2, W3 and W4 to 20, 21, 22, and 23 as we must continue with the flow of the custom made datasets of the models.

```
[ ]  vd.Week.replace("w1",20,inplace=True)
     vd.Week.replace("w2",21,inplace=True)
     vd.Week.replace("w3",22,inplace=True)
     vd.Week.replace("w4",23,inplace=True)
```

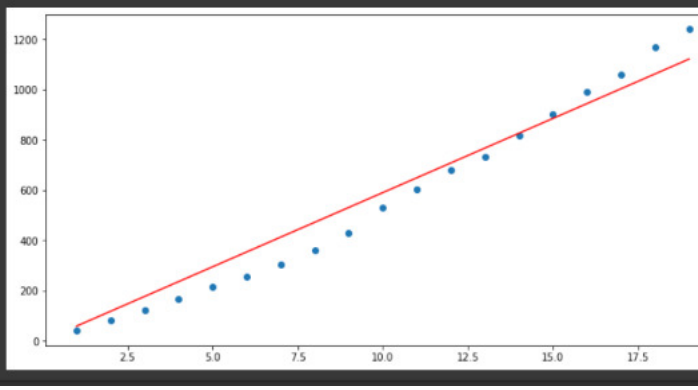
```
vd
```

	CategoryCode	ItemCode	Week	WeeklySales
0	category_2	1044502	20	11
1	category_2	1105009	20	11
2	category_2	913561	23	5
3	category_1	1048975	23	30
4	category_1	17287	21	60
...	...	...	...	...
365	category_2	124954	21	43
366	category_2	40759	20	48
367	category_1	1090303	20	19
368	category_2	1090276	22	6
369	category_1	3418	23	69

Figure 11: Changing W1, W2, W3 and W4 to 20, 21, 22, and 23

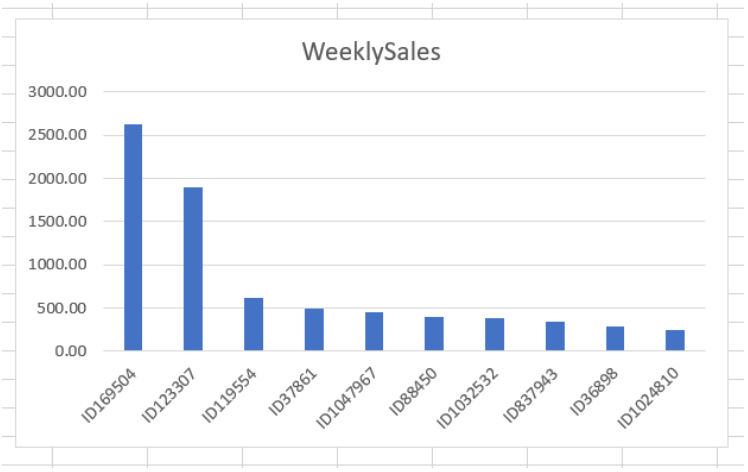
Currently, our model looks as follows. The blue dots of the graph signifies the given data sets and the red line signifies the prediction line.

```
plt.figure(figsize=(12,6))
plt.scatter(ffdf['Week'], ffdf[3418])
plt.plot(ffdf['Week'],  float(theta_1)*ffdf['Week'] + float(theta_0), 'r')
plt.show()
```



4.0 Business Insights

According to the data we gathered, the following 10 items will be the super star products in the upcoming month. Therefore, we suggest to run marketing campaigns to market these products even further or to encourage competitors to level up their game or to increase the logistics efficiency and pile up the stocks of those products in warehouses. We used Microsoft Excel to have these insights from the predicted data as we found it is much easier to work with it.



Furthermore, we are capable of analysis of the data to investigate the weeks which will have the most sales of products. This would not deliver any value over the items but it is valued over human resource management stand Point. Because, this would enable the HR department to allocate enough employees to the Sale Stations in Peak weeks.

Another insight that we can have from this prediction data, is that we can identify what categories of products will have their Peak of Sales in each week. This data may be valued for manufacturers and marketers, because it can be used to fortify their products in production, logistics, and marketing for the peaks of need of a month. Infact, it is also will be a great way to strategize selling and marketing mix to fight along with competitors.

5.0 Conclusion

Cleaning big data and cleaning those data to create predictions can be a hard task. As beginners we found it challenging and had a huge learning curve to come up with good predictions. It is also occurred to us how such data can be used in a business context. The overall experience of Data Storm 3.0 remains as an unforgettable yet very complex challenge.

-End-