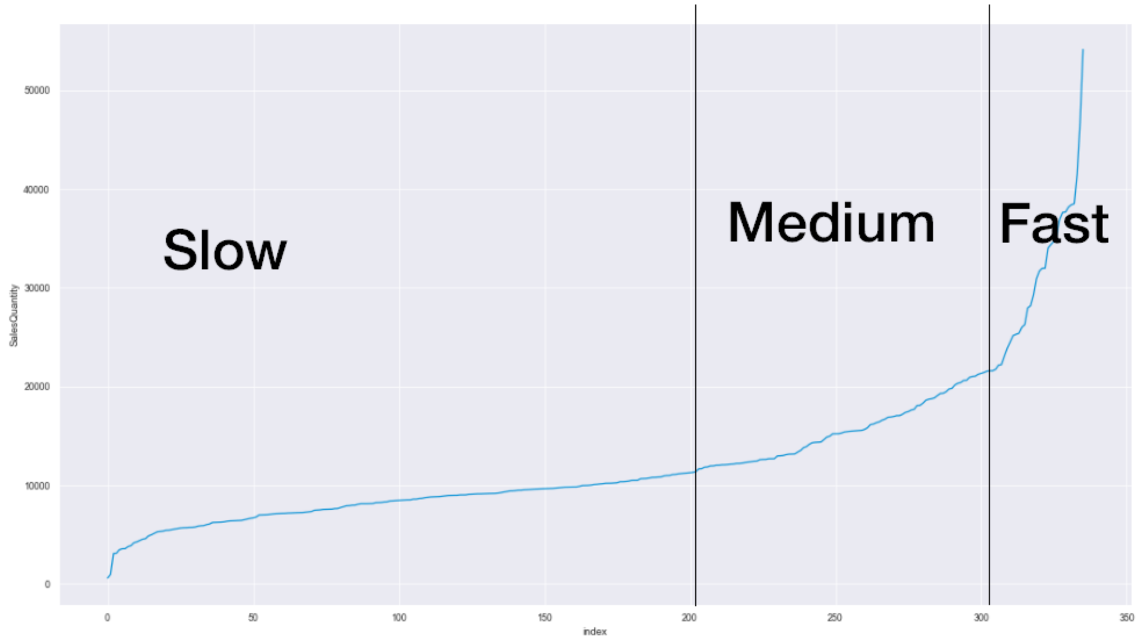
Fatih Kubilay Yavuz

Promotion Bump Assignment

My overall approach, in this case, was as follows:

- Analyze the data using Python and Excel
- Add more variables so that the models could explain more of the unexplained variance.
- Product groups are added in the same manner.
- Item types, store types, day, month, and weekday are added in order for models to explain the seasonality in the data.
- Promotion Dates are added as a binary variable so that models can explain the bump in the sales quantity during promotions. (Also, the promotion dates were wrongly arranged, so I fixed them.)
- In the end, a gradient boosted decision tree, Linear Regression, and Stochastic Gradient Descent regression are tried.
- Because the Linear Regression and SGD didn't accept the categorical variables as an argument unlike the Decision Tree algorithm, I had to convert to categorical variables to binary variables using OneHotEncoder.

What are your criteria for separating Fast, Medium and Slow items? Why?



My criteria to separate items was sales quantity. First I took a look at the chart. It can be seen that the difference between the sales quantity of items can be observed by looking at the derivative line of the point. So, I decided on a decision boundary, as seen above. Then to justify my claim I look at the raw data. The quantity increased immensely between the boundaries, as seen in the below frame.

	index	StoreCode	SalesQuantity
200	200	46	11271
201	201	86	11292
202	202	290	11363
203	203	158	11645
204	204	171	11658
205	205	161	11808
206	206	114	11832
207	207	338	11942

As seen, before passing the 203rd index sales quantity difference increases from 20~100 to 300~. Thus we call items less than 203rd index "Slow" items.

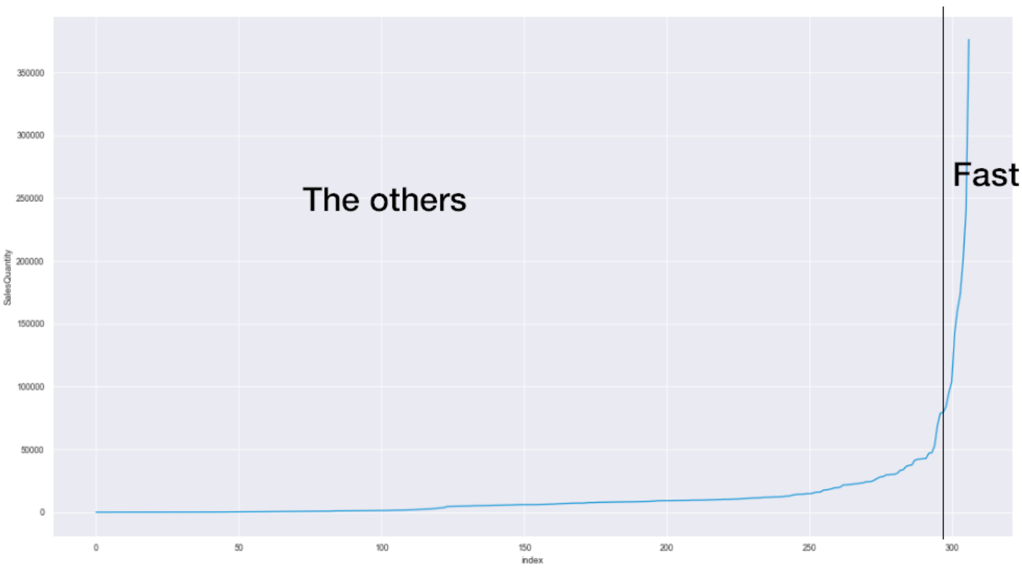
	index	StoreCode	SalesQuantity
303	303	193	21576
304	304	267	21581
305	305	83	21734
306	306	250	22155
307	307	274	22203
308	308	22	23059
309	309	57	23869

We can use the same claim for separating the fast and medium items. Thus, from the 203rd index to the 307th, we call the items "Medium" items.

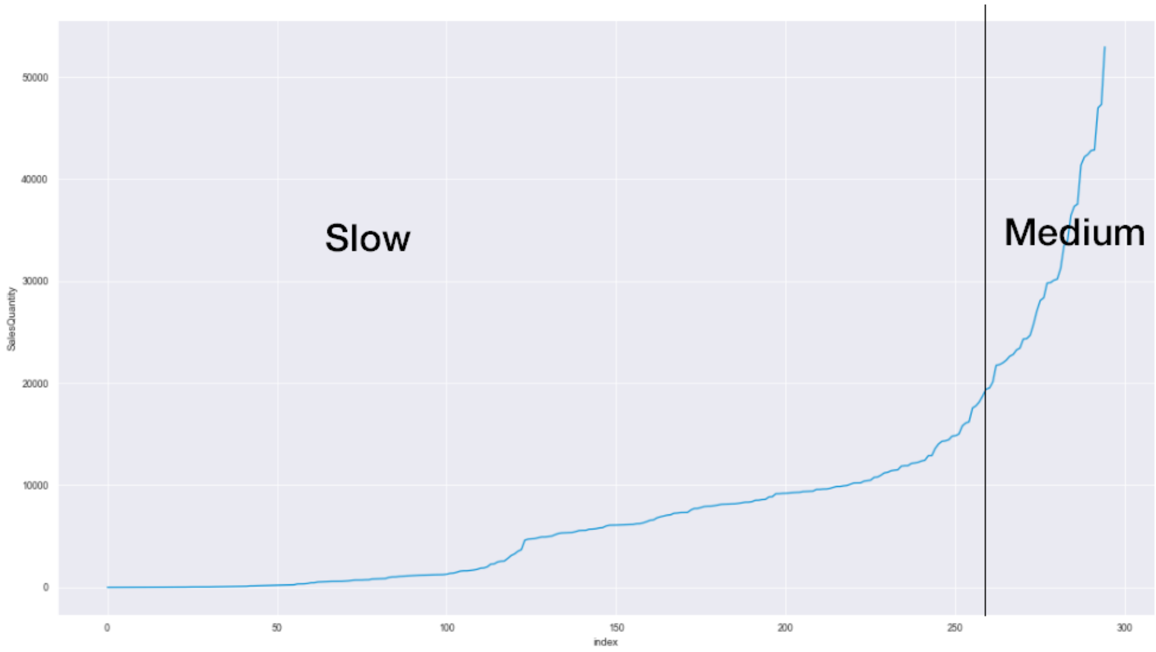
In the data, the time interval is the same so by comparing the sales quantity. The reason why I used such an approach is that the item's type depends on how fast it is sold. We can find this term by using the approach above.

What are your criteria for separating Fast, Medium and Slow Stores? Why?

The same approach, as above stays the same for the stores.



As seen, above the 295th index, the stores have an immense number of sales. This caused the chart not to show the other type of stores, so I pruned the chart a bit:



As seen, the difference between the 260th and the 261st store is more compared to the stores that are in the lower index than the 260th.

Again to prove my point, here is the boundaries' place from the raw data:

	index	ProductCode	SalesQuantity
255	255	232	17568
256	256	246	17776
257	257	249	18156
258	258	199	18729
259	259	156	19417
260	260	106	19525
261	261	162	20110
262	262	148	21749
263	263	126	21824
264	264	241	22005

Decision boundary between Slow and Medium

	index	ProductCode	SalesQuantity
290	290	190	42808
291	291	238	42859
292	292	210	46981
293	293	207	47305
294	294	216	52948
295	295	222	68946
296	296	170	78753
297	297	220	79720
298	298	219	84115
299	299	209	95116

Decision boundary between Medium and Fast

Which items experienced the biggest sale increase during promotions?

As expected, fast items experienced the biggest sale increase.

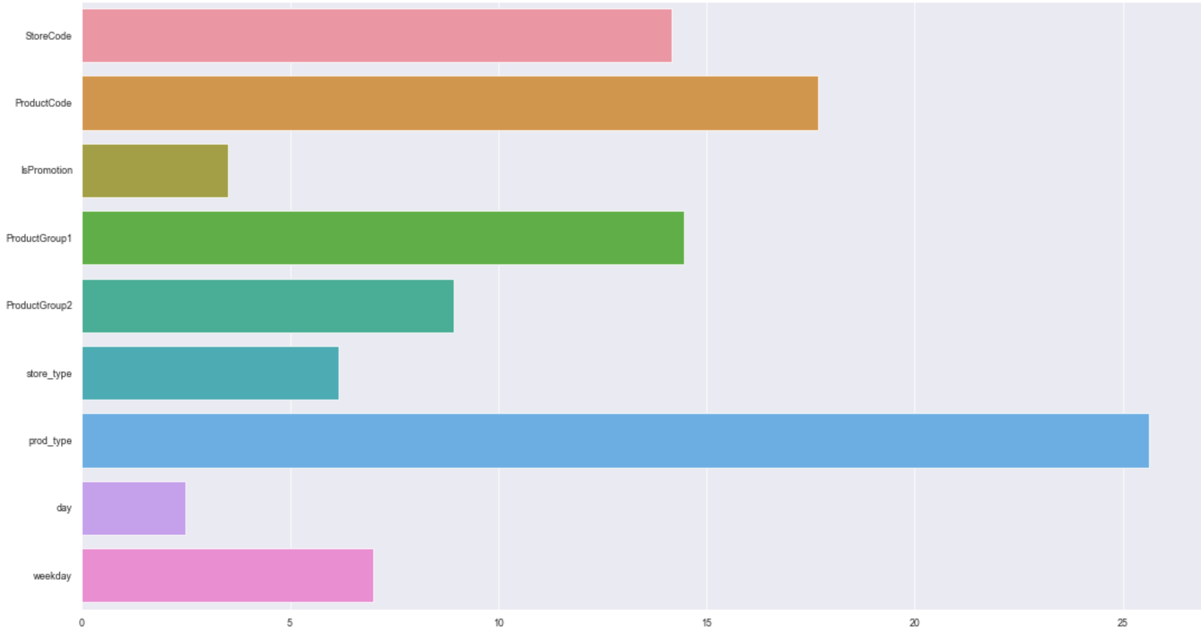
	before_promotion	in_promotion	increase	ProductCode	prod_type
212	1632.508380	2548.121212	915.612832	218	Fast
215	874.759777	1329.030303	454.270526	221	Fast
199	448.446927	705.939394	257.492467	205	Fast
203	410.452514	655.909091	245.456577	209	Fast
213	365.089385	568.606061	203.516675	219	Fast
214	345.737430	540.393939	194.656509	220	Fast
216	302.301676	449.515152	147.213476	222	Fast
210	232.497207	343.363636	110.866430	216	Medium
204	205.106145	311.121212	106.015067	210	Medium
201	206.882682	311.303030	104.420349	207	Medium
207	184.016760	287.000000	102.983240	213	Medium
163	803.659218	898.121212	94.461994	167	Fast
178	181.720670	267.757576	86.036905	184	Medium
164	1120.553073	1204.090909	83.537836	168	Fast
179	186.184358	268.000000	81.815642	185	Medium
183	164.407821	246.181818	81.773997	189	Medium
180	160.234637	234.515152	74.280515	186	Medium

Are there stores that have higher promotion reaction?

Yes, as expected, mostly slow stores have an immense percent increase among other types. Actually, here we should emphasize on the last entry of the data below because the others has low sales quantity during non-promotion days.

	before_promotion	in_promotion	percent_increase	StoreCode	store_type
220	0.000000	0.030303	inf	226.0	Medium
160	0.005587	0.060606	9.848485	163.0	Slow
282	0.016760	0.090909	4.424242	291.0	Medium
12	0.033520	0.090909	1.712121	13.0	Slow
221	0.106145	0.212121	0.998405	229.0	Slow
64	1.368715	2.484848	0.815461	66.0	Slow
222	0.033520	0.060606	0.808081	230.0	Slow
120	0.592179	1.030303	0.739851	122.0	Fast
223	0.178771	0.303030	0.695076	231.0	Slow
273	0.670391	1.121212	0.672475	282.0	Slow
306	2.195531	3.666667	0.670059	317.0	Slow
33	0.111732	0.181818	0.627273	34.0	Medium
29	6.983240	11.181818	0.601236	30.0	Slow
203	410.452514	655.909091	0.598015	209.0	Slow

What is the biggest effect explaining sales change during promotions?



Depending on the best-performed model which is a gradient boosted decision tree, the product type affects the sales the most.

StoreType	ProductType	Sum of SalesQuantity	AvgDaily	Change
Fast	Fast	31,141	865	0%
Medium	Fast	49,150	1,365	10%
Slow	Fast	53,923	1,498	12%
Fast	Medium	20,797	578	24%
Medium	Medium	34,640	962	21%
Slow	Medium	35,170	977	21%
Fast	Slow	29,495	819	18%
Medium	Slow	53,581	1,488	17%
Slow	Slow	62,830	1,745	15%
Grand Total		370,727	10,298	15%

Is there any significant difference between promotion impacts of the Fast versus Slow items?

When we take a look at the above analysis I made on excel, we see that Medium type items tend to have more sales quantity change during promotions.

StoreType	ProductType	Sum of SalesQuantity	AvgDaily	Change
Fast	Fast	31,141	865	0%
Fast	Medium	20,797	578	24%
Fast	Slow	29,495	819	18%
Medium	Fast	49,150	1,365	10%
Medium	Medium	34,640	962	21%
Medium	Slow	53,581	1,488	17%
Slow	Fast	53,923	1,498	12%
Slow	Medium	35,170	977	21%
Slow	Slow	62,830	1,745	15%

Is there any significant difference between promotion impacts of the Fast versus Slow stores?

Here, we see that the overall change during promotions for stores is the same for slow and medium, but little lesser for fast stores.

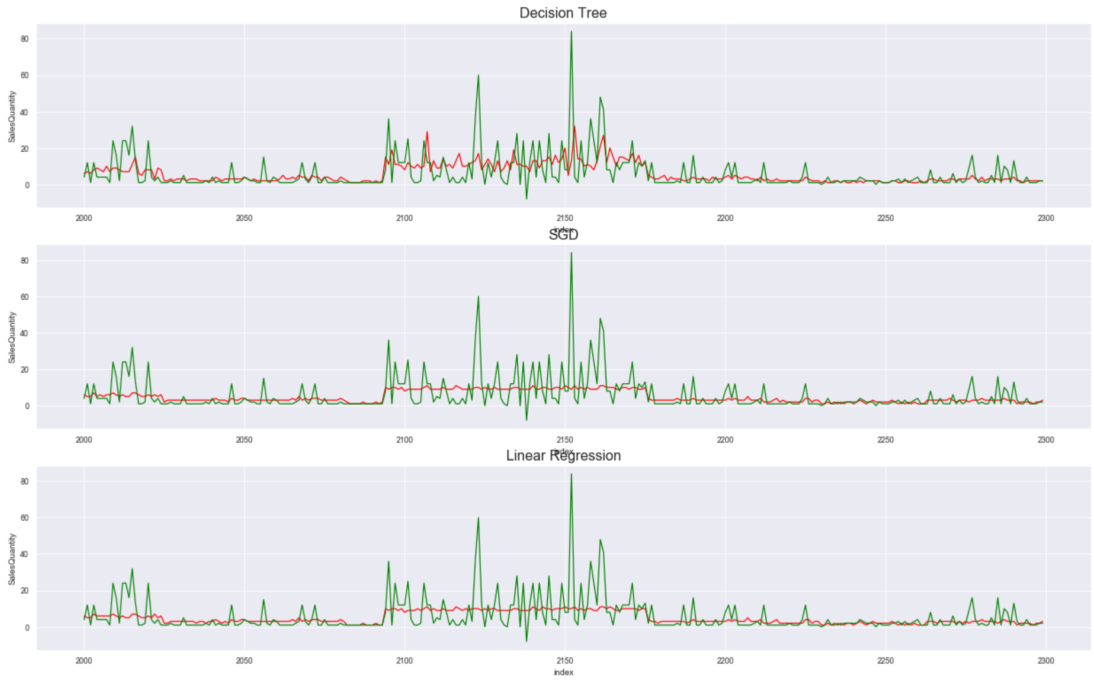
You are asked to measure how well your model has worked on this new data. Based on the model developed in part A forecast what would the effect of promotion 5 will be on sample store – item pairs. Compare the results of your forecast for promotion5 with the real observed sales during that period.

As seen, a more complex model proved to be better within the randomly selected frame of time. It gives more bias in the event of promotion thus causing the model to behave better.

The scores are RMSE, and MAE in order:

```
Decision Tree : 4.191470890958348
SGD : 4.332254494817876
LR : 4.310100704230157
```

```
Decision Tree : 1.8962242771035704
SGD : 2.1925065240375403
LR : 2.05779572637851
```



Furthermore, if we choose only promotion 5, the MAE becomes:

```
Decision Tree : 2.401711874623267
SGD : 2.526003616636528
LR : 2.5171790235081373
```

What measure would you use for goodness of fit?

The measures I used for this task are RMSE and MAE. However, using MAE is more beneficial because the RMSE error penalizes the closeness more than the MAE error. Since in our case, it is not that important to predict the exact quantity, using MAE is beneficial.

How good is your model developed in step 1?

The scores are given above.

What are the main problem points causing bad fits?

The main problem is the uncertainty of the quantities during promotions. As you can see, our overall is 1.8, however, during promotion 5 the MAE increases to 2.4. We can comment that the reason for the uncertainty of the predictions is caused by promotions.

What would you change in step 1?

My aim was to use Multivariate time series forecasting (VAR model), by using that I could have used the seasonality effect and using the promotion variable, I believe I could generate better results.

Conclusion and Report

Because it is easy to explain the variables using regression, I will emphasize on the Linear Regression when making assumptions.

When we take a look at which stores has the greater coefficient, we see that:

```
cat_vars=["StoreCode","ProductCode","ProductGroup1","ProductGroup2","store_type","prod_type","weekday","IsPromotion","day"]
store_coefs=feature_coefs["x0"]
store_index=store_coefs.index(max(store_coefs))
enc.categories_[0][store_index],max(store_coefs)
```

```
(117, 1.2011309438554605)
```

which means that the store with code 117 has the greatest coefficient in order words, has the most expectation in terms of sale quantity.

```
cat_vars=["StoreCode","ProductCode","ProductGroup1","ProductGroup2","store_type","prod_type","weekday","IsPromotion","day"]
item_coefs=feature_coefs["x1"]
item_index=item_coefs.index(max(item_coefs))
enc.categories_[1][item_index],max(item_coefs)
```

```
(137, 9.256194321396528)
```

As seen here, if the product has code 137 the sales quantity is increased by 10~. Actually, we could determine whether the product or store is whether Fast, Medium or Slow by observing the expectations of the variables.

```python
cat_vars=["StoreCode","ProductCode","ProductGroup1","ProductGroup2","store_type","prod_type","weekday","IsPromotion","day"]
weekday_coefs=feature_coefs["x6"]
weekday_index=weekday_coefs.index(max(weekday_coefs))
enc.categories_[6][weekday_index],max(weekday_coefs)
```

```
(5, 0.7213441916633714)
```

Here we see that the index of 5, which equals Saturday, has the most effect on the response variable. So, when the day is Saturday the sales quantity increases by 0.72, which makes sense because Saturday is a holiday.

```python
model_lr.coef_[-2]
```

```
0.4529783280468405
```

Furthermore, the coefficient of Promotion's binary variable (Whether the transaction happens on a promotion day or not) is 0.45. This means when there is promotion, a product has an expectation of being sold 0.45 more in terms of quantity. However, this is not biased when it should be, because this number contains all the items, Fast, Slow or Medium, and when there is promotion the Slow and Medium items tend to sell more. Also, this concept applies to stores, as well.

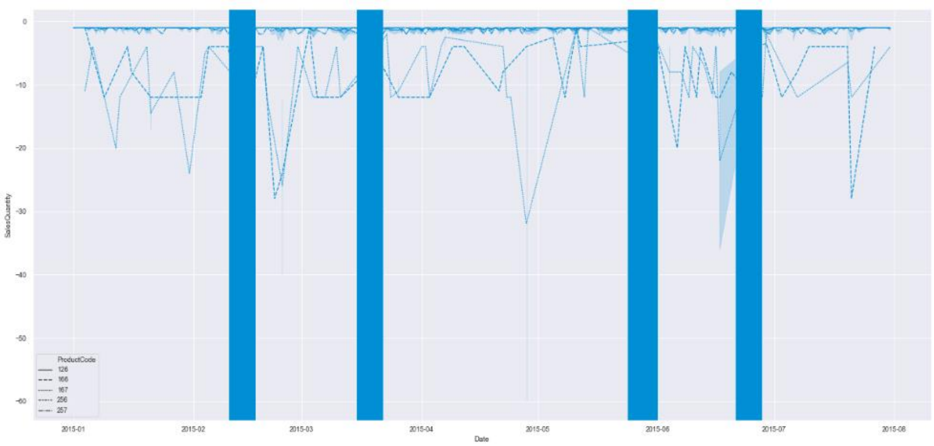
Depending on this issue, I conducted an analysis on Excel (because it is interactive and gives you more flexibility in this case):

IFPromo	1			
StoreType	ProductType	Sum of SalesQuantity	AvgDaily	Change
Fast	Fast	31,141	865	0%
Fast	Medium	20,797	578	24%
Fast	Slow	29,495	819	18%
Medium	Fast	49,150	1,365	10%
Medium	Medium	34,640	962	21%
Medium	Slow	53,581	1,488	17%
Slow	Fast	53,923	1,498	12%
Slow	Medium	35,170	977	21%
Slow	Slow	62,830	1,745	15%
Grand Total		370,727	10,298	15%

IFPromo	0		
StoreType	ProductType	Sum of SalesQuantity	AvgDaily
Fast	Fast	151,612	861
Fast	Medium	82,038	466
Fast	Slow	121,842	692
Medium	Fast	217,759	1,237
Medium	Medium	139,866	795
Medium	Slow	223,011	1,267
Slow	Fast	234,959	1,335
Slow	Medium	141,709	805
Slow	Slow	267,047	1,517
Grand Total		1,579,843	8,976

As seen, the change percentage changes with the type of stores and products.

Also, we can see that there is an increase in returns after the promotions.



Future Works and Improvements

- Creating different models for different types of products or stores will increase predictability during promotion days.
- More sophisticated models, like CatBoost I used, may be used in these kinds of cases but will require more computing power.
- The other sheet shows a different analysis which explains the percent increase including the product groups, store types. We could use these 48 instances and create models.
- Going further: Creating different models for different products or stores. (Also, causes more computing power but beneficial)
- We could prepare a feasibility analysis including the above statement and use the one giving us the optimal output.
- We could add national holidays to the data. So, that we could explain the bumps or diminishes along with the data.
- Also, more information about the stores and items will help.
- Stock data of the products and the availability in the stores of the products would help the model explain the sales quantity.