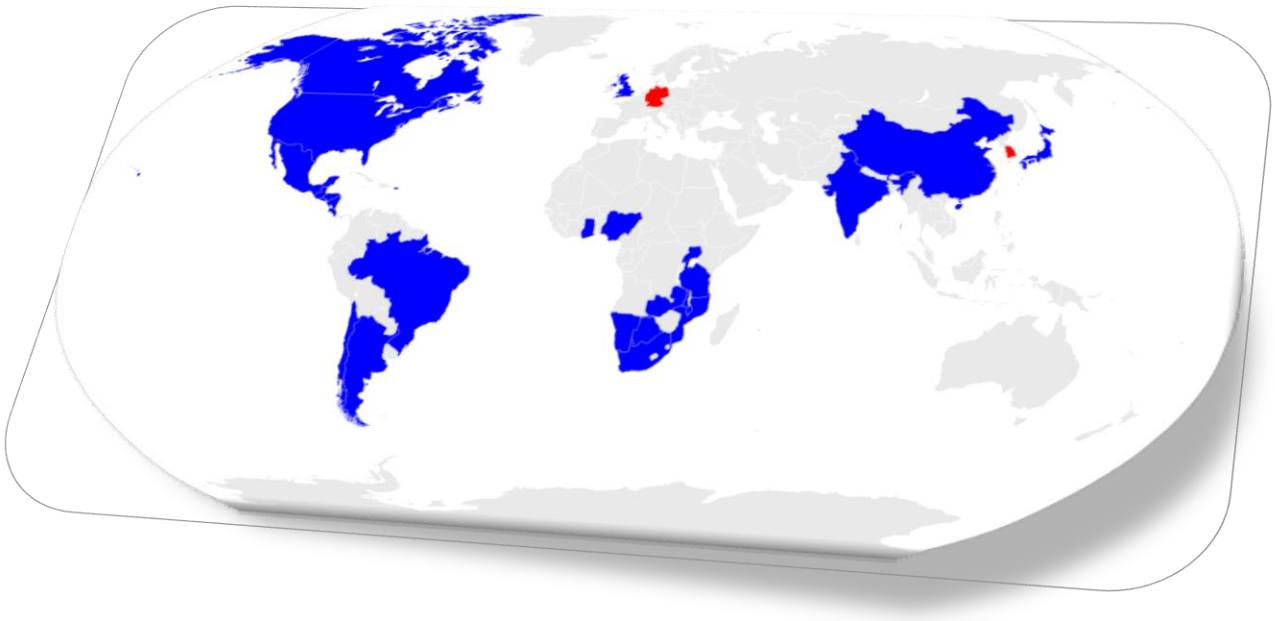


Walmart locations in various countries across the world

How Weather Forecast Predicts Walmart's Sales Outlook

Introduction:

Walmart operates 11,450 stores in 27 countries, managing inventory across varying climates and cultures. Extreme weather events, like hurricanes, blizzards, and floods, can have a huge impact on sales at the store and product level. Walmart relies on a variety of vendor tools to predict sales around extreme weather events, but it's an ad-hoc and time-consuming process that lacks a systematic measure of effectiveness. So I have created my version of prediction here. We have chosen 2 products across 6 stores to do our predictions.

Literature Review:

Like most retailers, Walmart has been basing sales and marketing decisions on weather data for years in obvious ways, such as putting up umbrella or snow-shovel displays in advance of rain or snow.

But now, in the second year of an extensive partnership with the Weather Company, the Earth's largest retailer is delving far deeper into sometimes unlikely correlations between weather and store sales.

Martha Starr wrote a paper on Retails sales affected by weather conditions. She explains how monthly fluctuations in consumer spending is often attributed to the weather conditions.

Dataset

key.csv - the relational mapping between stores and the weather stations that cover them **sampleSubmission.csv** - file that gives the prediction format
train.csv - sales data for all stores & dates in the training set
test.csv - stores & dates for forecasting (missing 'units', which you must predict) **weather.csv** - a file containing the NOAA weather information for each station and day **noaa_weather_qclcd_documentation.pdf** - a guide to understand the data provided in the weather.csv file

Data set and Field Descriptions:

Dataset 1:

date - the day of sales or weather

store_nbr - an id representing one of the 45 stores

station_nbr - an id representing one of 20 weather stations

item_nbr - an id representing one of the 111 products

units - the quantity sold of an item on a given day

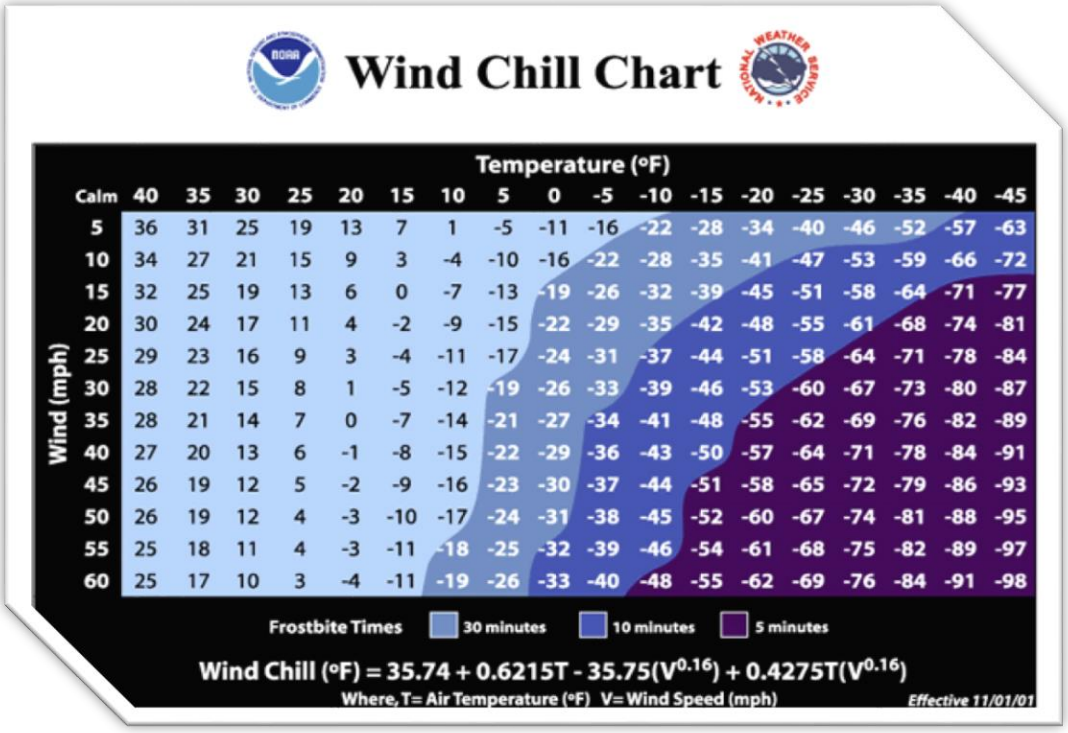
id - a triplet representing a store_nbr, item_nbr, and date. Form the id by concatenating these (in that order) with an underscore. E.g. "2_1_2013-04-01" represents store 2, item 1, sold on 2013-04-01.

Dataset 2:

Weather data set contains weather information of the weather station which is located closer to the stores.

These are some of the fields in the weather dataset:

date pid sid units station_nbr tmax tmin tavg dewpoint wetbulb heat stnpressure sealevel resultspeed resultdir avgspeed



This is a description of Wind chill charts

Approach and Block diagram

Downloading the dataset is the first approach once you downloaded them.
The dataset contains 45 stores and 111 products sales information in the rows.
Each store has a store ID, Product ID, Date and number of units sold.

This is a screen shot of one of the 1 datasets before merging with the weather dataset.

date	store_no	product_no	units
21/03/2013	1	9	34
18/07/2013	1	9	20
7/5/2013	1	9	45
20/05/2012	1	9	25
7/9/2013	1	9	30
21/05/2014	1	9	6
4/12/2012	1	9	13
26/02/2014	1	9	9
23/09/2014	1	9	22

Data set after bins are created and daily sales consolidated.

date	store_no	product_no	units	volume_sale
21/03/2013	1	9	34	low
18/07/2013	1	9	20	low
7/5/2013	1	9	45	low
20/05/2012	1	9	25	low
7/9/2013	1	9	30	low
21/05/2014	1	9	6	low
4/12/2012	1	9	13	low
26/02/2014	1	9	9	low
23/09/2014	1	9	22	low

Weather data is merged with the store data and cleaned for any missing values

units	volume_sales	tmax	tmin	tavg	dewpoint	wetbulb	stnpressure	sealevel	resultspeed	resultdir	avgspeed
621	high	39	26	33	16	28	30.03	30.19	5.5	31	8.1
555	high	43	31	37	15	30	29.83	30.01	10.4	25	11.4
345	medium	48	32	40	27	35	29.68	29.88	5.5	25	8
312	medium	38	23	31	13	26	30.14	30.32	6.5	31	7.7
258	medium	36	17	27	8	22	29.96	30.13	2.5	28	3.7
258	medium	92	63	78	62	68	29.82	29.97	2.1	21	3.4
243	medium	78	60	69	56	61	29.94	30.08	7.2	21	7.5
240	medium	27	13	20	0	14	29.51	29.72	11.8	31	12.4
230	medium	64	45	55	48	51	29.97	30.09	5.1	20	5.5
228	medium	43	30	37	26	32	29.88	30.07	4.4	4	7.1
226	medium	52	46	49	38	44	30.13	30.31	6.6	6	6.9
219	medium	39	25	32	15	27	29.91	30.12	6.6	28	7.7
216	medium	52	35	44	30	38	29.77	29.92	6.6	25	8
210	medium	53	29	41	31	37	29.64	29.77	1.7	20	2.9
207	medium	45	23	34	15	28	29.89	30.08	2.5	21	3
206	low	52	41	47	33	40	29.54	29.72	7.9	29	8.3
206	low	46	32	39	22	33	29.62	29.78	6.2	31	7.4
201	low	48	42	45	38	42	29.77	29.89	2.8	5	3.5
200	low	93	70	82	71	74	29.94	30.07	6.7	20	7.4
195	low	80	42	61	37	50	29.92	30.09	5.3	26	6.4
194	low	18	5	12	-7	8	30.01	30.18	7.9	26	9.5
186	low	47	22	35	19	31	29.98	30.1	10	22	10.3
186	low	73	59	66	63	64	30.05	30.2	1.2	16	2.9
186	low	80	54	67	57	61	29.98	30.14	0.7	14	2
183	low	53	45	49	43	46	30.04	30.22	5.4	6	6.1

Now this data set is ready for **predictions.**

I have done three iterations where I include and exclude certain columns to find out the outcomes

Step 1: Data Cleaning

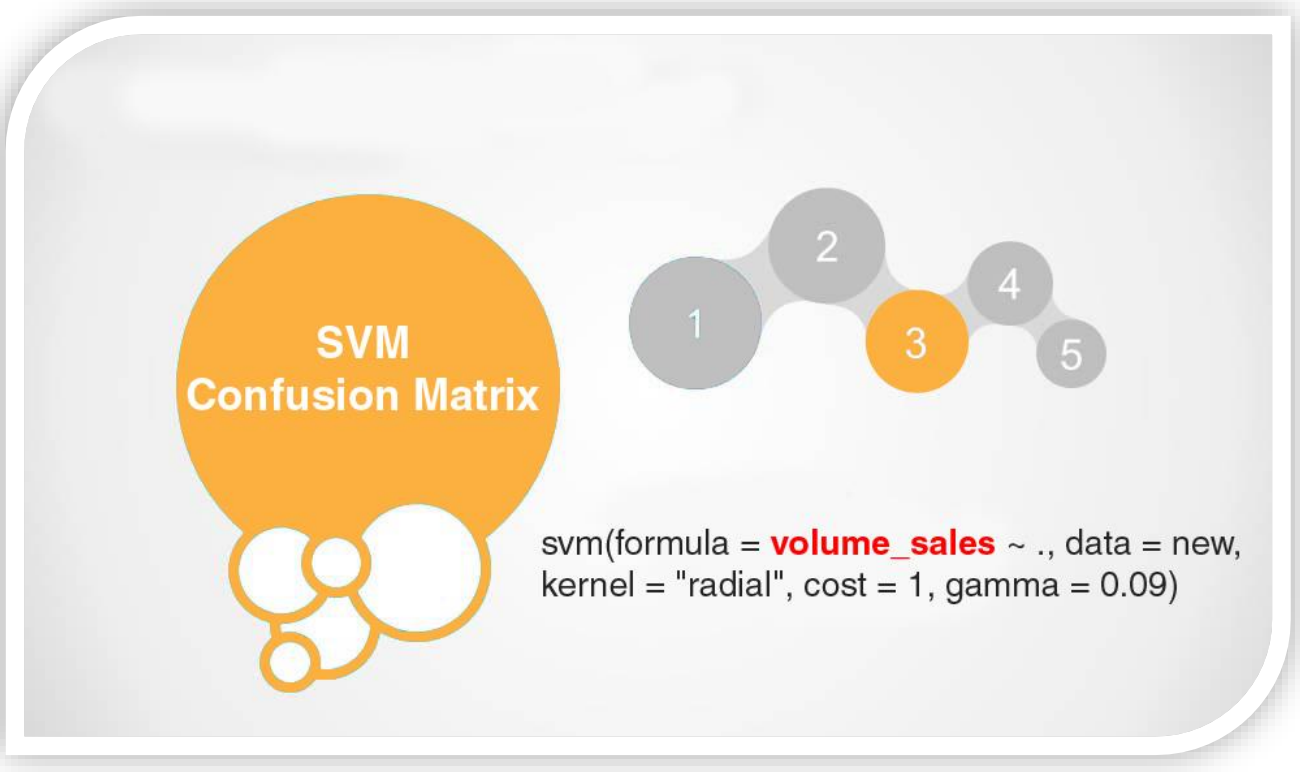
1. Remove the rows which contains zero values (where there is zero sales).
   Example: Store 1
   Total rows=100242
   Non zero rows =**2877**
2. Some stores have one product sold multiple times so we need to calculate per day sales for each products.
3. Once we calculated this we need to separate each product sold in one store since they are distinguished by product ID.
   Example: Store1 has products **{'51', '89', '93', '9', '40', '99', '28', '47'}**
4. Once you find out the products which are in the store you separate them in to individual csv files.
5. This is a very important step where you create bins based on values of units sold
   We do this by finding the total units sold and then calculate its bins.
   30% and less gets the bin value **'low'**
   30-60% gets **'medium'**
   60% and above gets **'high'**

Step 2: Merging the sales data with the weather data.

The weather data is associated with the store data in a separate file called key.csv where each store is matched with each weather station. Some stores who are located close together will share a weather station.

1. We select all the data from the store and product csv file and merge them with the weather dataset.
2. We have the individual products file in which date they primary key and the weather dataset has weather information and we can use the date in the weather dataset to merge both of them together.
3. Once its merged the weather dataset has some values with **'M'** which means missing value. Since the amount of M's located is less than 1% I have removed the missing values from the dataset.
4. In order to run Support Vector Machine Algorithm, the values of the data cannot be constant so we removed the values which are not scaling. Example: store number, product number and we need to remove the date as well.

Step 3: Running SVM and Confusion Matrix



This step is where we write algorithms to predict our outcomes.

I have selected 2 products on 6 stores to compare with them.

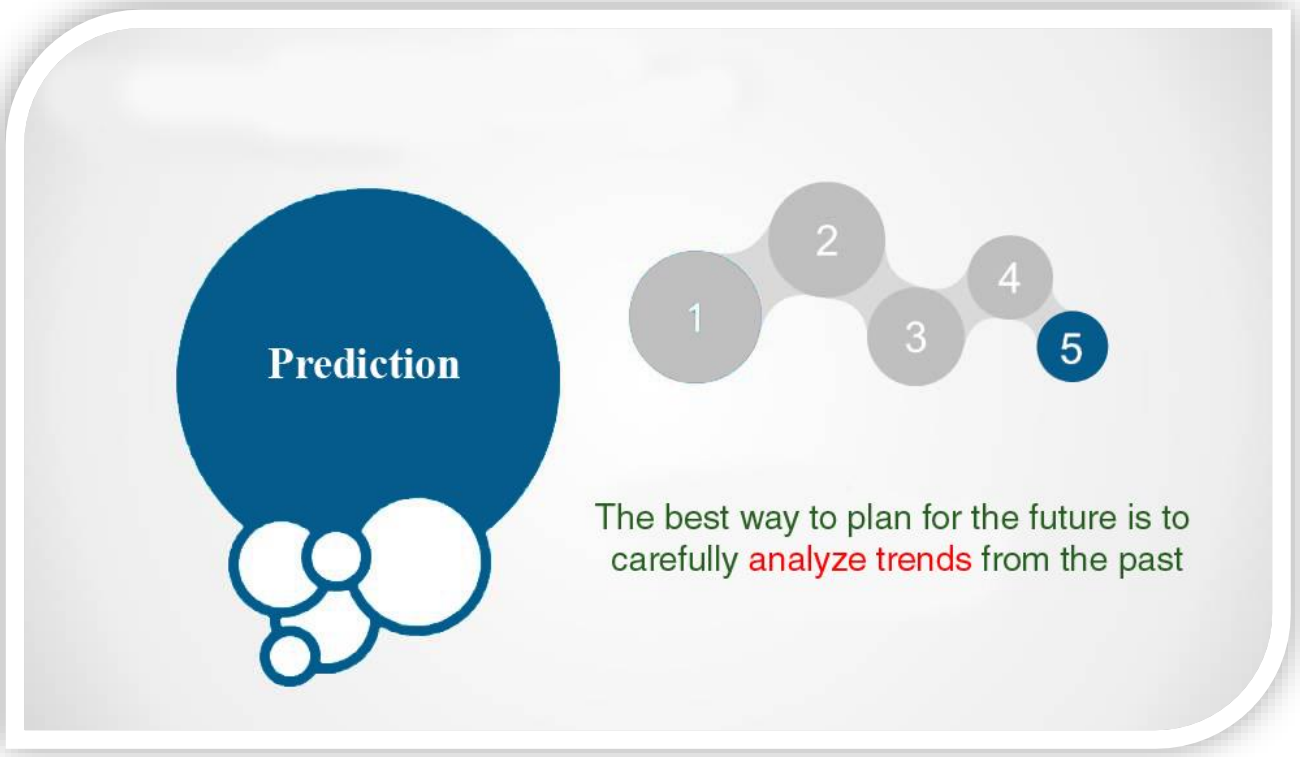
Store_1_product 9	Store_3_product 45
Store_14_product 9	Store_6_product 45
Store_32_product 9	Store_15_product 45

We run SVM and Confusion Matrix in each of the six datasets to figure out the predictions.

Step 4: k-nearest neighbor classification

In k'th nearest neighbor classification we need to find a value for k. We should choose it to minimize predication error, and to measure prediction error we need a loss function. That is, a function which takes as input the truth and the prediction and returns a value that is large when the prediction is far from the truth and 0 when it matches the truth.

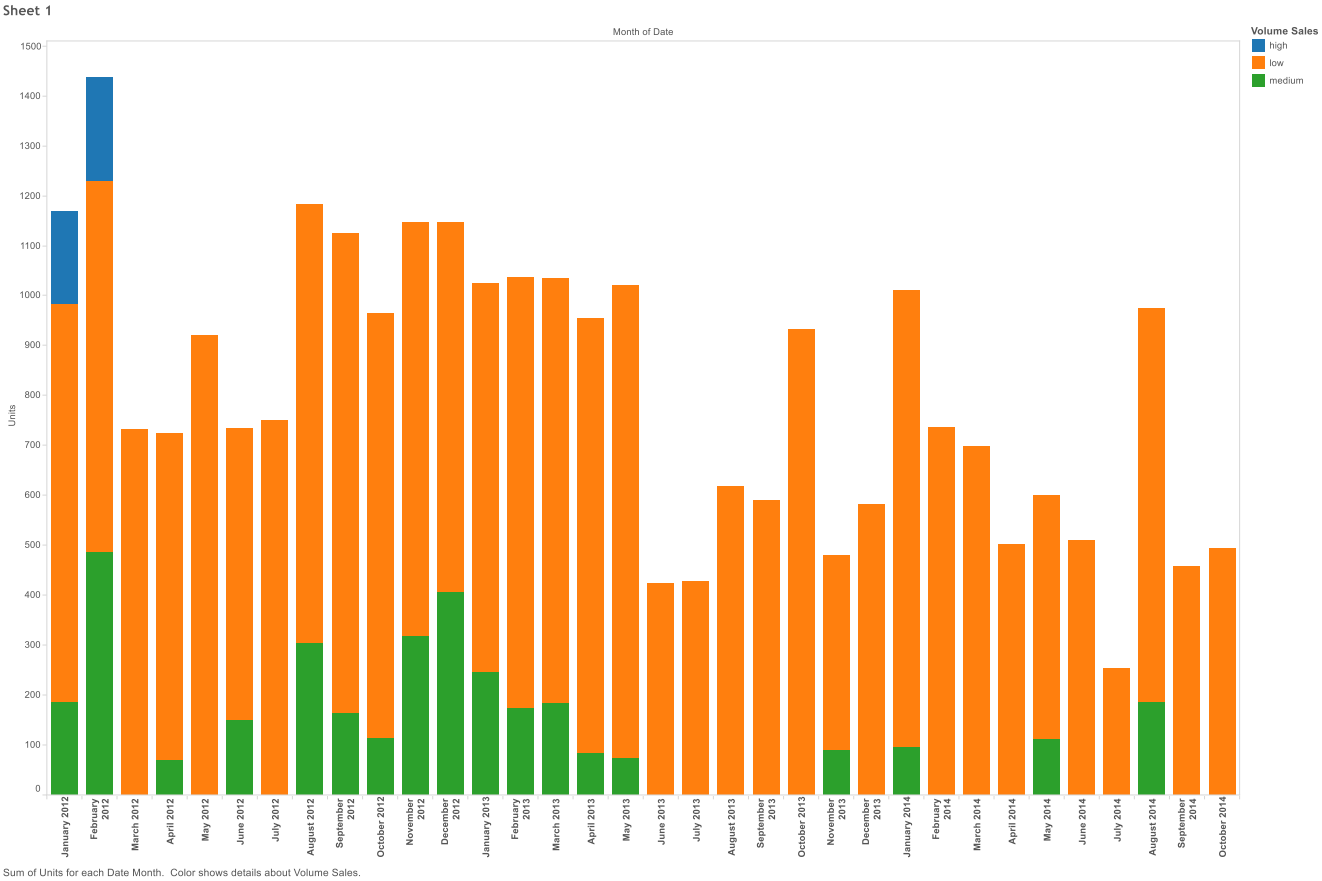
Step 5: Predictions

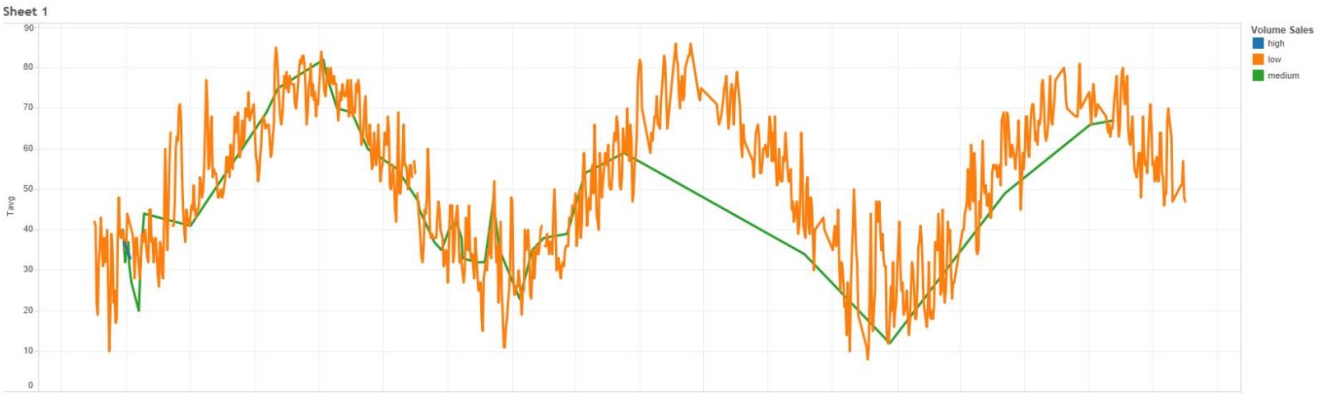


In this step we run the predictions to find out whether our algorithm has predicted correctly and with what degree of accuracy. This step is where we use our 10% testing data on the model we created. In business, predictive models exploit patterns found in historical and transactional data to identify risks and opportunities. Models capture relationships among many factors to allow assessment of risk or potential associated with a particular set of conditions, guiding decision making for candidate transactions

Results

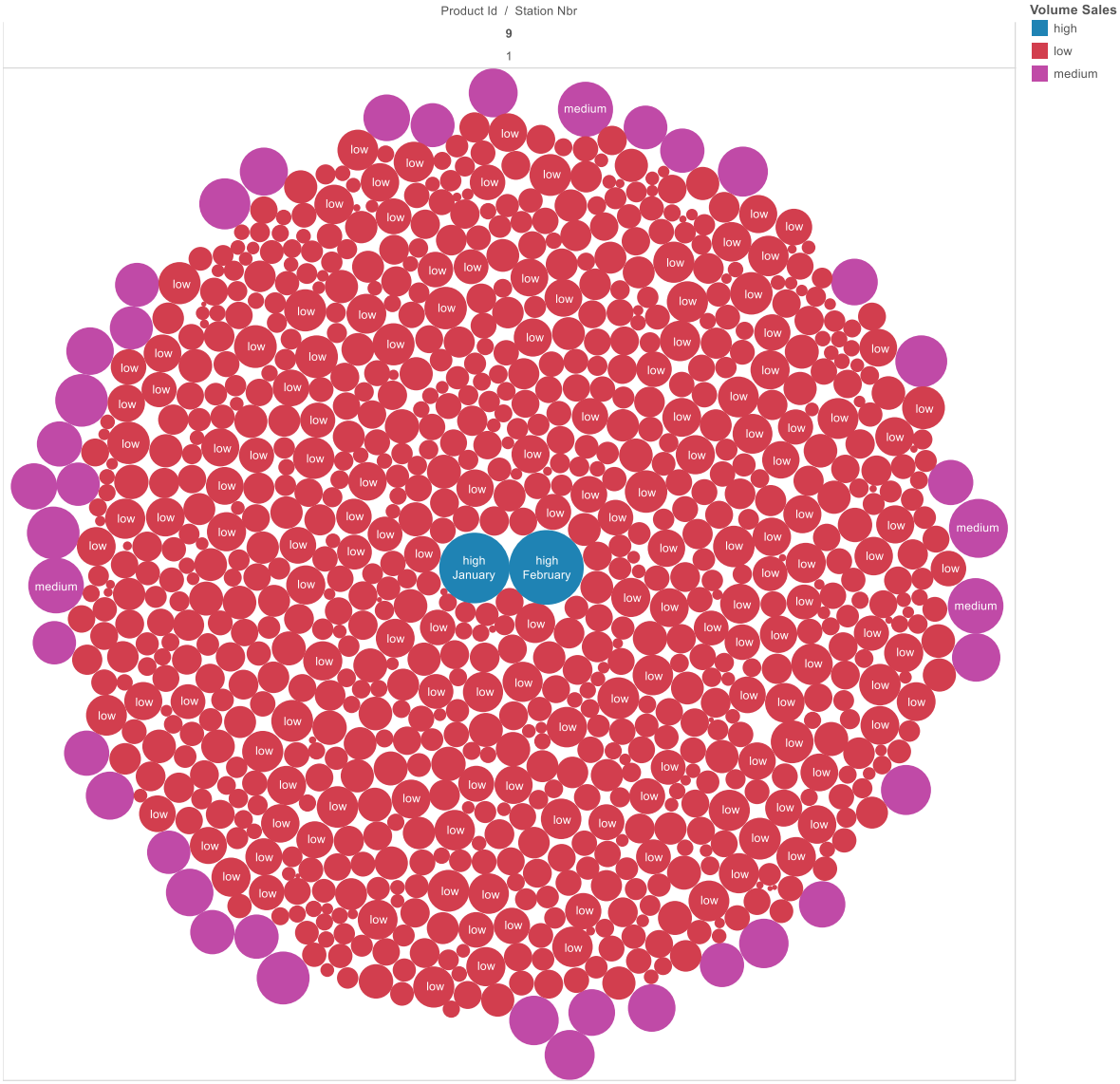
I have selected store 1 and product 9 for visualization:





This product sale goes in parallel with the temperature. I assume this could be a summer product for sure.

Sheet 1



Volume Sales, Date2 Day and Date Year broken down by Product Id and Station Nbr. Color shows details about Volume Sales. Size shows sum of Units. The marks are labeled by Volume Sales, Date2 Day and Date Year.

Svm and confusion matrix results:

The dimensions of this data set is 914 observations and 12 variables:

SVM :

```
Call:
svm(formula = volume_sales ~ ., data = new, type = "C-classification")
Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1
      gamma:  0.09090909
```

```
Number of Support Vectors:  74
 ( 2 13 59 )
Number of Classes:  3
Levels:
 high low medium
```

Sales prediction:

```
> table(sales_predict,y)
             y
sales_predict high low medium
       high      2   0      0
       low       0 899      9
       medium    0   0      4
```

Tuning the Svm model to get the best gamma and doing a 10-fold cross validation:

```
Parameter tuning of ‘svm’:

- sampling method: 10-fold cross validation

- best parameters:
 cost gamma
   10  0.09

- best performance: 0.007608696
```

Svm model After tuning:

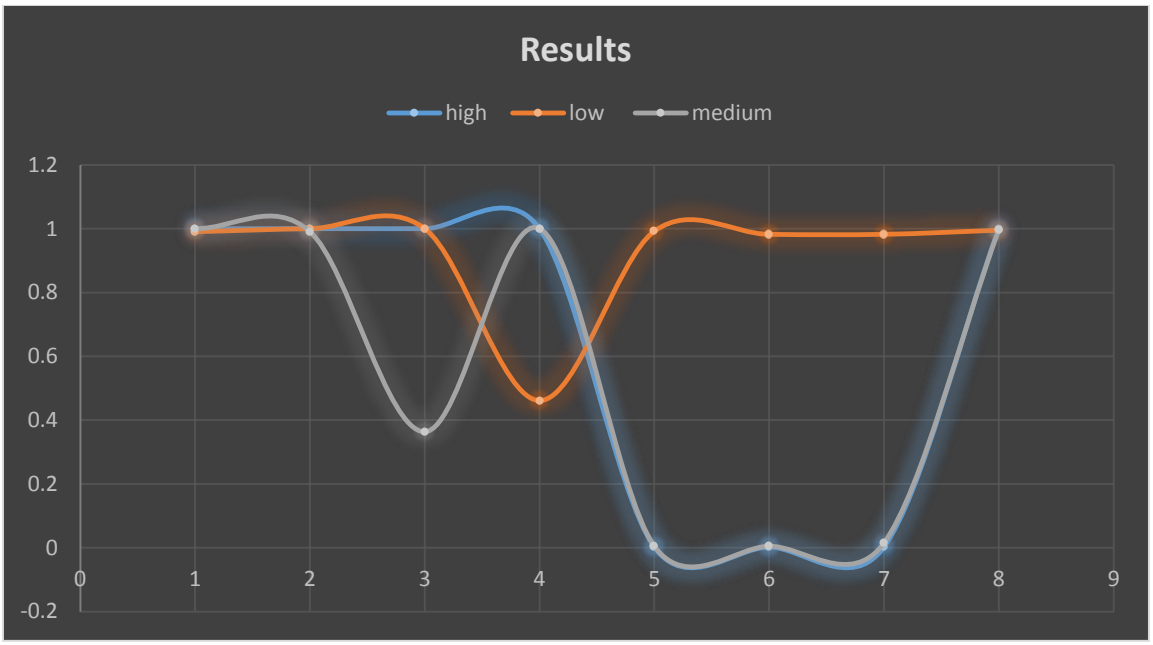
```
Call:
svm(formula = volume_sales ~ ., data = new, kernel = "radial", cost = 1, ga
mma = 0.09)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1
      gamma:  0.09

Number of Support Vectors:  74

 ( 2 13 59 )


Number of Classes:  3

Levels:
 high low medium
```

Prediction tables:

```
> table(pred,y)
        y
pred     high low medium
  high      2   0      0
  low       0 899      9
  medium    0   0      4

> table(sales_predict,y)
             y
sales_predict high low medium
       high      2   0      0
       low       0 899      9
       medium    0   0      4
```

Confusion Matrix:

```
 Confusion Matrix and Statistics

           Reference
 Prediction high low medium
     high      2   0      0
     low       0 720      0
     medium    0   7      4

 Overall Statistics

                Accuracy : 0.9891
                  95% CI : (0.9804, 0.9962)
     No Information Rate : 0.9918
     P-Value [Acc > NIR] : 0.7445

                   Kappa : 0.05521
  Mcnemar's Test P-Value : NA

 Statistics by Class:

                      Class: high Class: low Class: medium
 Sensitivity            1.000000     0.9904      1.000000
 Specificity            1.000000     1.0000      0.990398
 Pos Pred Value         1.000000     1.0000      0.363636
 Neg Pred Value         1.000000     0.4615      1.000000
 Prevalence             0.002729     0.9918      0.005457
 Detection Rate         0.002729     0.9823      0.005457
 Detection Prevalence   0.002729     0.9823      0.015007
 Balanced Accuracy      1.000000     0.9952      0.995199
```



This chart is a plotting of the confusion Matrix. If you see the model predicts low and medium well because of the amount of samples are higher in these two bins.

Knn Cross validations:

In order to do Knn we need to normalize our data set and shuffle the rows to get an even shuffle Once we shuffled we will split them. We will split the dataset into 10% testing and 90% training to conduct our prediction.

We create the algorithm and test it. Code followed by results

Reading the Dataset :Code

```
wm<-read.csv("C:/Users/Rahul/Dropbox/CKME 136 CAPSTONE PROJECT/Working Datasets/1_9/1_9.csv")
attach(wm)
table(wm$volume_sales)
head(wm)
dim(wm)
set.seed(9850)
wm[2]
```

Mixing up the data set

```
gp<-runif(nrow(wm))
wm<-wm[order(gp),]
```

Rescale the features normalization

```
normalize<-function(x){
return((x-min(x))/(max(x)-min(x)))}
normalize(c(10,02,30,40,50))
```

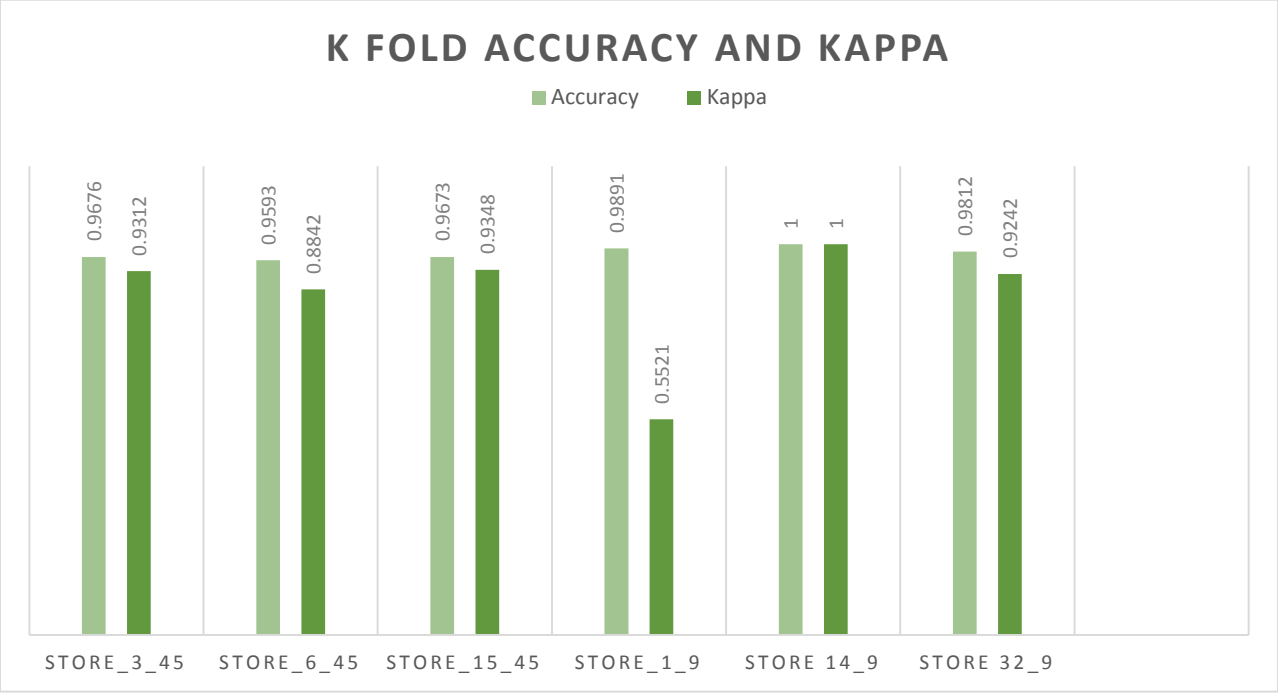
Splitting the Dataset

```
walmart_train<-walmart_n[1:822,]
walmart_test<-walmart_n[823:914,]
walmart_train_target<-wm[1:822,2]
walmart_test_target<-wm[823:914,2]
```

Rescale the features normalization

```
m1<-knn(train=walmart_train,test=walmart_test,cl=walmart_train_target,k=13)
table(walmart_test_target,m1)
summary(m1)
```

Product Results comparison



Store 3 product 45

Statistics by Class:

	Class: high	Class: low	Class: medium
Sensitivity	1	0.9784	0.9615
Specificity	0.98958	0.9777	0.979
Pos Pred Value	0.5	0.9576	0.9883
Neg Pred Value	1	0.9887	0.932
Prevalence	0.01031	0.3402	0.6495
Detection Rate	0.01031	0.3328	0.6244
Detection Prevalence	0.02062	0.3476	0.6318
Balanced Accuracy	0.99479	0.978	0.9702

Store 6 product 45

Statistics by Class:

	Class: high	Class: low	Class: medium
Sensitivity	1	0.9661	0.9563
Specificity	0.997	0.9614	0.9712
Pos Pred Value	0.91304	0.8382	0.9924
Neg Pred Value	1	0.9928	0.8491
Prevalence	0.03052	0.1715	0.798
Detection Rate	0.03052	0.1657	0.7631
Detection Prevalence	0.03343	0.1977	0.7689
Balanced Accuracy	0.9985	0.9638	0.9638

Store 15 product 45

Statistics by Class:

	Class: high	Class: low	Class: medium	
Sensitivity	1	0.9958	0.9516	
Specificity	0.98543	0.9749	0.9964	
Pos Pred Value	0.78846	0.9444	0.998	
Neg Pred Value	1	0.9982	0.9178	
Prevalence	0.05151	0.3003	0.6482	
Detection Rate	0.05151	0.299	0.6168	
Detection Prevalence	0.06533	0.3166	0.6181	
Balanced Accuracy	0.99272	0.9853	0.974	

Store 1 product 9

Statistics by Class:

	Class: high	Class: low	Class: medium
Sensitivity	1	0.989	1
Specificity	1	1	0.989041
Pos Pred Value	1	1	0.272727
Neg Pred Value	1	0.3846	1
Prevalence	0.002729	0.9932	0.004093
Detection Rate	0.002729	0.9823	0.004093
Detection Prevalence	0.002729	0.9823	0.015007
Balanced Accuracy	1	0.9945	0.994521

Store 14 product 9

Statistics by Class:

	Class: high	Class: low	Class: medium
Sensitivity	1	1	1
Specificity	1	1	1
Pos Pred Value	1	1	1
Neg Pred Value	1	1	1
Prevalence	0.006658	0.8735	0.1198
Detection Rate	0.006658	0.8735	0.1198
Detection Prevalence	0.006658	0.8735	0.1198
Balanced Accuracy	1	1	1

Store 32 product 9

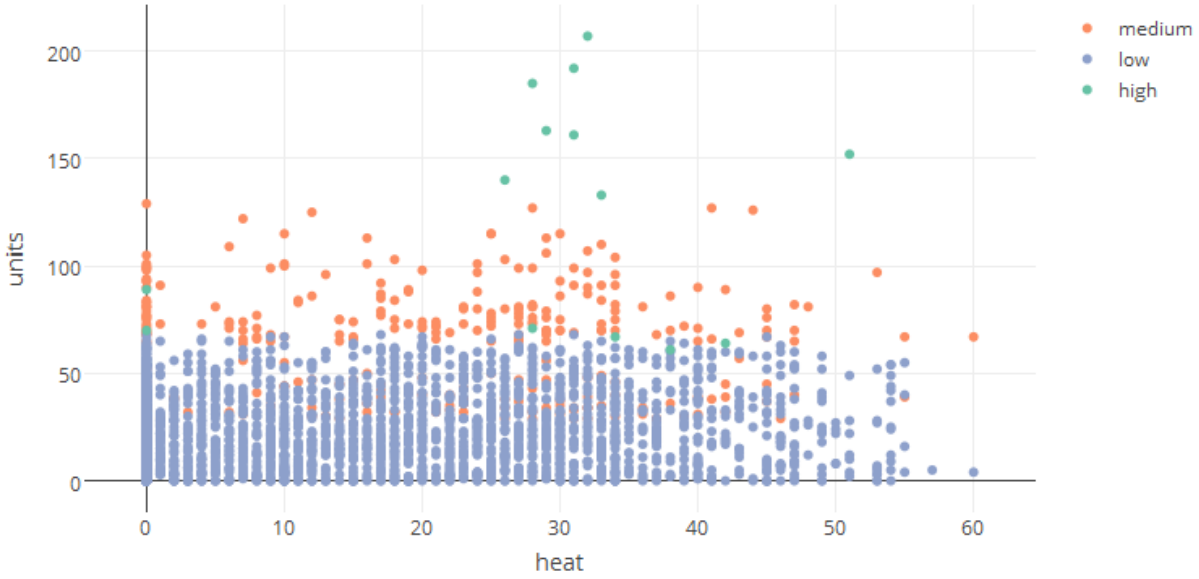
Statistics by Class:

	Class: high	Class: low	Class: medium
Sensitivity	1	0.9825	0.972
Specificity	0.996222	1	0.9826
Pos Pred Value	0.4	1	0.8966
Neg Pred Value	1	0.9008	0.9956
Prevalence	0.002513	0.8631	0.1344
Detection Rate	0.002513	0.848	0.1307
Detection Prevalence	0.006281	0.848	0.1457
Balanced Accuracy	0.998111	0.9913	0.9773

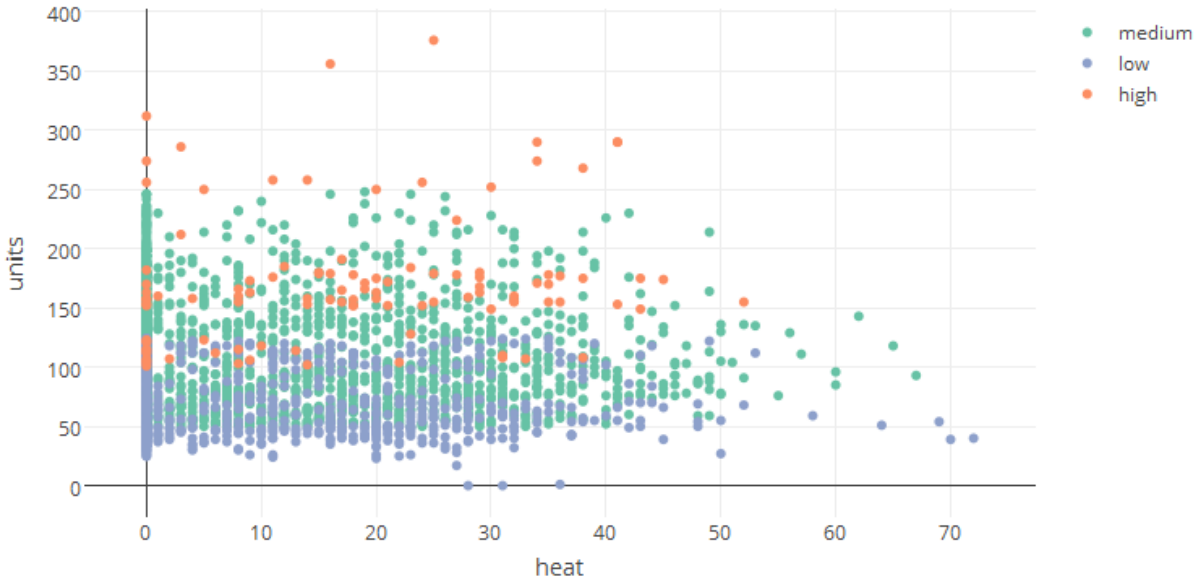
Sales of 2 products in 3 stores each.

Product 9 has been sold in store number 1,14 and 32, Product 45 sold in store number 3,6 and 15

Pid 9



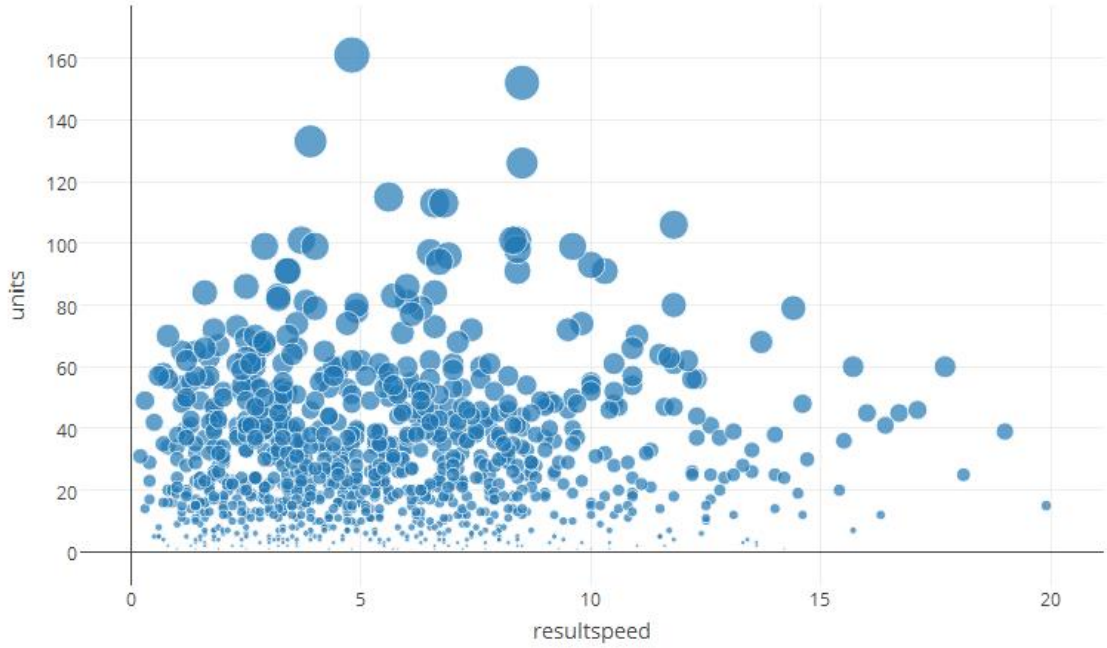
Pid 45



Units sold during windy(Result Speed Variable)

Pid 9

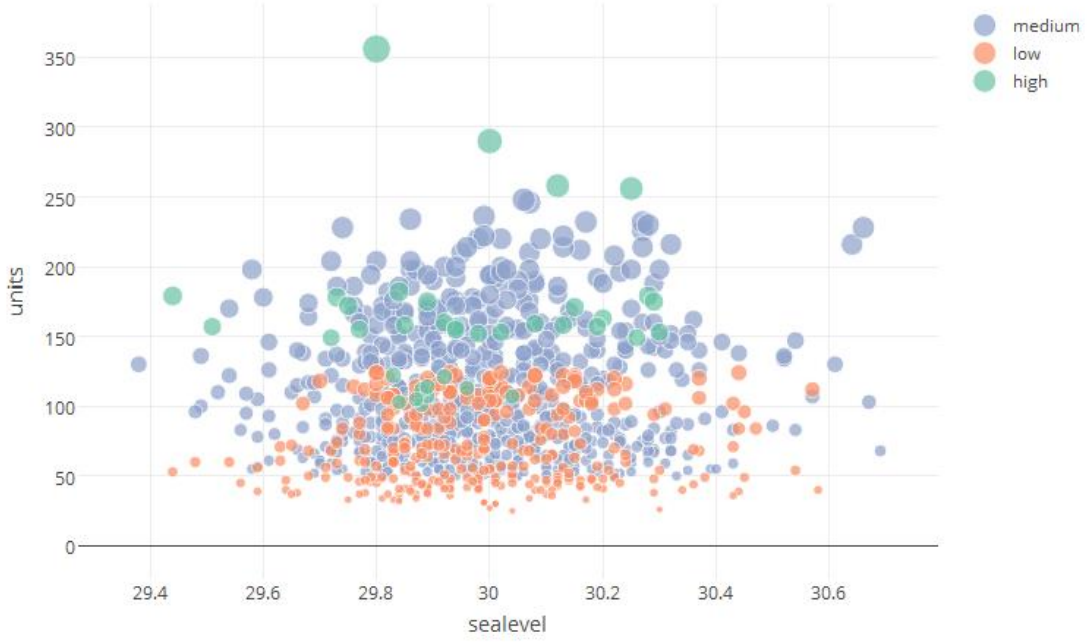
Pid 45



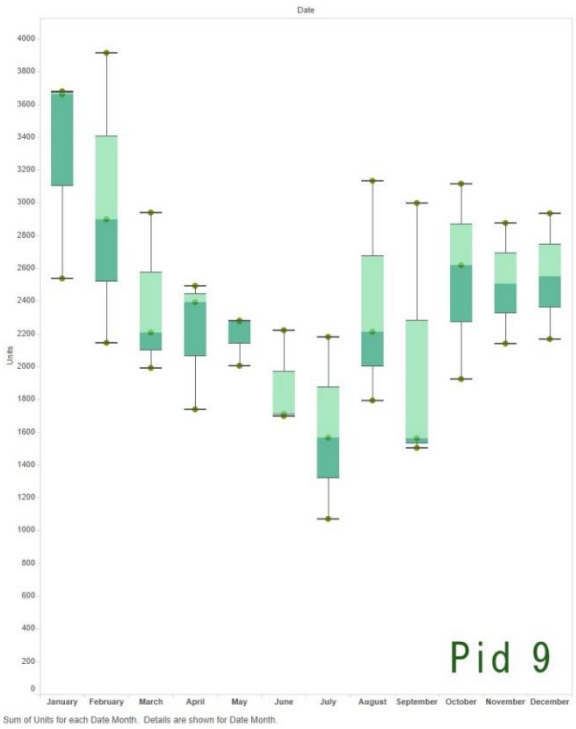
Sea level and product sale comparison Chart

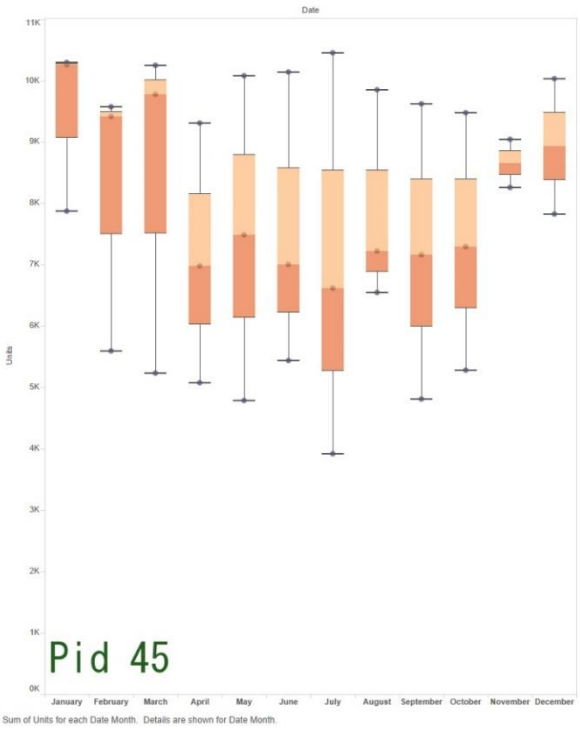
Pid 9

Pid 45



Units sold monthly comparison Chart





Conclusions

Walmart started using big data even before the term big data became known in the industry. With this project it was a huge learning experience and a privilege to work with this dataset. We have not been provided with store locations or product names in the dataset. But we can see what Walmart uncovered with their data.

Some facts what Walmart uncovered with their data:

- People are more likely to eat steak when it's warm out with higher winds but no rain, but not if it gets too hot.
- Ground beef does better with higher temperature, low wind, and mostly sunny conditions.
- Salads sell better when the temperature tops 80 but winds are low.

Walmart has found thousands of such correlations that it's now trying to harness.
Among other ways Walmart is trying to use data to improve its merchandising or marketing is by tracking trends on Pinterest. We can see stuff trending on Pinterest.

Potential Limitations

The distance between the stores and the weather plays a huge role in the prediction. Without knowing the product names adds another level of complication in the analysis. This data provided consists of 45 stores of Walmart but there are lots of other Walmart stores out there so this is just a small part of the Big Data only for the past 2 years, In order to accurately predict what products have to be in the pipeline we need to use a much bigger training set.

Software's used for this project:

Python,R,Tableau,Excel and word

GitHub :

All the codes for these steps are located in this **Git hub link.**

https://github.com/enian/walmart

References

1. http://www.federalreserve.gov/pubs/feds/2000/200008/200008pap.pdf

2. http://www.cisco.com/c/dam/en_us/solutions/industries/retail/downloads/cisco-retail-analytics-wht-paper.pdf

3. http://docplayer.net/2093376-Forecast-of-sales-of-walmart-store-using-big-data-applications.html

4. https://gonewithsuperwind.wordpress.com/

5. http://econweb.ucsd.edu/~brothtra/pdfs/BlameItOnTheRain.pdf

6. http://www.kylemurray.com/papers/MDFP_JRCS2010.pdf

7.https://books.google.ca/books?id=JfELBwAAQBAJ&pg=PA195&lpg=PA195&dq=retail+sales+vs+weather+paper&source=bl&ots=qz2q5qjhEG&sig=Y_l-Ov4AweWxznhN7qzMuFZSo50&hl=en&sa=X&ved=0ahUKEwiOyZTYiN3MAhWK3YMKHZwsB0UQ6AEIVjAL#v=onepage&q=retail%20sales%20vs%20weather%20paper&f=false

8. http://www.emeraldinsight.com/doi/abs/10.1108/09590551211230232

9.http://poseidon01.ssrn.com/delivery.php?ID=729001022114010019090102093029024094026006014001084049030125080018119066064106093099122024127099030044044015002113112104064107021029063058010100127031105021082073072007013054084120113119073125117127111083019064091125119092008125010087082010075116005119&EXT=pdf

10. https://ideas.repec.org/a/zag/market/v25y2013i2p199-211.html