**Abstract**

Food sales prediction is concerned with estimating future sales of companies in the food industry, such as supermarkets, groceries, restaurants, bakeries and patisseries. Accurate short-term sales prediction allows companies to minimize stocked and expired products inside stores and at the same time avoid missing sales. In order to be able to build a system to predict the demand we have to be able to take design decisions of a data analyst working on food sales prediction, such as the temporal granularity of sales data, the input variables to use for predicting sales and the representation of the sales output variable. In addition, we have to be able to decide which machine learning algorithms suites better food sales prediction and appropriate measures for evaluating their accuracy.

Keywords Food · Sales prediction · Demand forecasting · Machine learning · Regression · Time series forecasting

**Business problem:**

In today’s highly competitive and constantly changing business environment, the accurate and timely estimation of future sales, also known as sales prediction or sales forecasting, can offer critical knowledge to companies involved in the manufacturing, wholesale or retail of products. Short-term predictions mainly help in production planning and stock management, while long-term predictions can help in business development decision making. In our specific case we have a fast-food chain in Brazil with 400 stores with difficulty in predicting its short term production.

Sales prediction is particularly important for this particular company due to the short shelf-life of many of its products, which leads to loss of income in both shortage and surplus situations. Producing too many leads to waste of products, while producing too few leads to opportunity loss. Therefore we have a situation where predicting correctly how much you have to produce of each item each day is important.

Moreover, food consumer demand is constantly fluctuating due to factors such as price, promotions, changing consumer preferences or weather changes. Sales prediction is typically done arbitrarily by managers. However, skilled managers are hard to find and they are not always available. In our specific case this forecast is done based on experience however it remains far from accurate. Average loss (too many or too few) spins around 10%.

Here it is important to put the management perspective: It is their view today they relay too much in the managers (e.g. they may get sick or take a leave) and would like to have a computer systems that can play the role of a skilled manager. Over time the expectation was to have some tool which would free the company from the human dependence. In addition of that they believe that the level of error is high and could be reduced.

Therefore, from the management perspective a system capable of predicting the sales would be worth having even if at the beginning it doesn´t perform better than the current process. Equal performance would be acceptable. In addition of that there was an understanding that the system would be able to improve its performance overtime as more and more historical data is added to its reference database. (Machine learning effect) .

**Analytical problem**

The problem is how to build a model which effectively predicts the demand with a level of assertiveness equal or superior to the current one and improves overtime. One way to build such a system would be to try and model the expert knowledge of skilled managers within a computer system. Alternatively, we could exploit the wealth of sales data and related information to automatically construct accurate sales prediction models via machine learning techniques. The latter is a much simpler process, it is not biased from the particularities of a specific sales manager and it is dynamic, meaning it can adapt to changes in the data. Furthermore, it has the potential to outweigh the prediction accuracy of a human expert, who typically is imperfect.

Nevertheless, we listened to the thoughts of the people currently in charge of making this forecast and we were informed that they believe that the demand is correlated to the following factors:

1. Day of the month (payment days usually have bigger demand)
2. Day of the week (Fridays, Saturdays, Sundays and holidays usually have a big demand)
3. Month (Holidays months usually have bigger sales – In Brazil Dez-Jan-jun-Jul)
4. Weather (temperature, rain and sun have an impact in what people eat)

This particular company sales its food through several channels: 1) directly from its stores, 2) through a web deliver service and 3) through a call-center. In our study we are not going to differentiate these channels just counting the total volume sold of each item each day.

Here it is worth mentioning that the insigns provided by the people in charge of the process today should be seen with a grain of salt given the fact that they know that a system like that would be built to replace them. Therefore all these assumptions must be checked against the hard data.

**Datasets**

To be able to build our model we decided to use as a sample the sales in the city of São Paulo which alone responds for almost 40% of the total. This is a simplifying strategy given the fact that if the process works for this city we easily deploy it in the others.

**Getting the data**

Initially we managed to get from the company the sales by type of item per day for thirteen months (jan – 2018 and Jan-2019) – 396 registers .

Secondly we managed to get the weather stations measurements in São Paulo for the whole year of 2018 and January of 2019. It is public information available at the website:

<http://www.inmet.gov.br/projetos/rede/pesquisa/>

In sequence we prepared this data crossing these two files unifying them by date.

**Evaluating the data**

Accuracy and completeness: A visual inspection showed that the sales data per day was basically correct, although some values seem to be too high or too low. The weather measurements had some problems of completeness. There were several days without the insolation, temperature and humidity recorded. In addition of that there are several days where the level of rain is zero, this is a problem because we don´t know it happens because it wasn´t recorded or because in fact didn´t rain in these days.

**Cleaning the data**

To deal with the inconsistences we defined three strategies:

1. Regards the missing values in precipitation we managed to see the average rain in each month (public information) and check it against the sum of each day/month in the database. Through this process we managed to identify that in fact the zero represented days without rain (the data was right).
2. The registers with temperature min, max or med and humidity equals zero were filled with the mean of these parameters (just two samples fall into this scenario).
3. In the case of the sun intensity we had a situation where 196 out of 396 samples were equal zero. Considering that there is no possibility that the sun didn´t appear for so many days it was assumed we had a problem with the data. We managed to check the average sun intensity per month in the city of São Paulo (public information) and fill the gaps manually. Subsequently we identify that the mean of the registers with measurement represent the average therefore it was possible to implement a code to correct it automatically.
4. The values of sales which diverge too much from what is typical were treaded by a bell distribution were the values whose frequency were smaller than 1% were eliminated from the sample. Note that we did not eliminate the whole row ( Each row had sales for each one of the eight types of products) we just didn´t count the line when predicting the specific item.

The evaluation was important because allowed us to create a cleaning layer in the R code where we check these factors (2 and 3) and adjust it automatically – It is important because we assume that a new samples will be added to the training data as time goes by and this new data probably will suffer from the same problems.

**Selecting the training data and the test data**

The idea is to use 95% of the measurements as our training data and 5% as our test data.

**The meaning of the columns (data dictionary)**

Data – Day of the weather measurements and sales

Precipitac – Volume of rain in millimetres per square meter during the day

Tempmax – Max temperature during the day

Tempmin – Minimum temperature during the day

Tempmed – Average temperature during the day

Umidade – Level of humidity in the air

Insolacao – Level of sun

Diasemana – day of the week abreviation

Diasem – weight of the day of the week considering the average volumes sold

Mes – Weight of the month considering the average sales volumes

Desserts – number of desserts sold in the day

Pizzas – Number of pizzas sold in the day

Beverage – Number of Beverage sold in the day

Cbmaker – Number of a special dish sold in the day

Combos – Number of combos sold in the day

Sfiha – Number of sfihas sold in the day

kit1 – Number of kits 01 sold in the day (Is a dish with a gift)

kit2 - Number of kits 02 sold in the day (Is a dish with a gift)

snack – Number of snacks sold in the day

pastas – Number of pastas sold in the day

dishes – Number of lunches sold in the day

promotion – Number of promotions sold in the day (This is episodic and may not be counted)

savory – Number of savory sold in the day

salads – Number of salads sold in the day

total – Total number of items sold in the day

The meaning of the numbers in the field diasem is:

1. Monday, Tuesday, Thursday
2. Friday
3. Saturday and Sunday
4. Holidays

The number represents the weight of the day regards sales. This weight reflects the view of the current planners regards sales and we need to check if the assumption holds. Analyzing the sales per day of the week we have the following graphic:



The current assumption of separating the days of week into four categories seems to be a bit wrong , the graphic shows three ranges as follows:

1. Monday – Tuesday – Wednesday-Thursday – Range 11% - 13% of the sales
2. Friday and Saturday – Range 17% and 19% of the sales
3. Sunday – Range 14% a 15% of the sales

The holidays match the volumes of FRI and SAT falling into the Range 2. The distribution by type of item goes as shown:



Here we can see that the division of training data by type of the day would be better if done conjugate with the type of product, That means the variation in sales along the week is not uniform for all products. We didn´t do that and surely it is an aspect of this model which can be improved.

Our analysis suggested that we group the days by type as follows:

1. Monday – Tuesday – Wednesday-Thursday – Range 11% - 13% of the sales
2. Sunday – Range 14% a 15% of the sales
3. Friday and Saturday – Range 17% and 19% of the sales

The same process of grouping the days by sales profile was done for the months, the current method was as follows:

1. March, April, May, August, September, October, November
2. February and July
3. January ,June and December

The number represents the weight of the month regards sales, this weight reflects the view of the current planners regards sales and we checked if the assumption holds. Analyzing the actual sales monthly by month classification we saw the following:



That suggests that there is a differentiation among the months as follows:

1. 1.500.000 items sold monthly (8,61% above the baseline)
2. 1.380.000 items sold monthly (Baseline)
3. 2.079.000 items sold (50% above the baseline)

That suggests the change 1 with 2 as a classification for the month to keep it aligned with the sales volumes:

1. 1.380.000 items sold monthly (Baseline)
2. 1.500.000 items sold monthly (8,61% above the baseline)
3. 2.079.000 items sold (50% above the baseline)

As we can see, although the weather do have some influence regards the amount sold of each type of product the main aspects influencing the demand are the days of week and month of the year.

Note that the objective is first to identify if the listed factors are in fact defining the demand and identify which ones are the most relevant.

Stablishing a correlation using the Kendall method between the demand for specific items. Here we are going to limit ourselves to the following items:

Desserts – number of desserts sold in the day

Pizzas – Number of pizzas sold in the day

Beverage – Number of Beverage sold in the day

Sfiha – Number of sfihas sold in the day

snack – Number of snacks sold in the day

pastas – Number of pastas sold in the day

dishes – Number of lunches sold in the day

savory – Number of savory sold in the day

salads – Number of salads sold in the day

The reason for that is the fact that the others are not regular items but some sort of promotion only made available for limited span of time. Items expurgated from the analysis:

Cbmaker – Number of a special dish sold in the day

Combos – Number of combos sold in the day

kit1 – Number of kits 01 sold in the day (includes a gift)

kit2 - Number of kits 02 sold in the day (Includes a gift)

promotion – Number of promotions sold in the day (This is episodic and may not be counted)

Seasonal parameters:

Diasem – weight of the day of the week considering the average volumes sold (1,2 or 3)

Mes – Weight of the month considering the average sales volumes (1,2 or 3)

Using the following code in R:

“setwd("/Curso-ML/Assignment-1/")

variables <- read.csv("data-06.csv")

summary(variables)

variables1 <- variables[17:32]

summary(variables1)

library(corrplot)

cor.test(variables1$diasem, variables1$pizzas)

cor.test(variables1$diasem, variables1$pizzas)

forcorrplot <-cor(variables1)

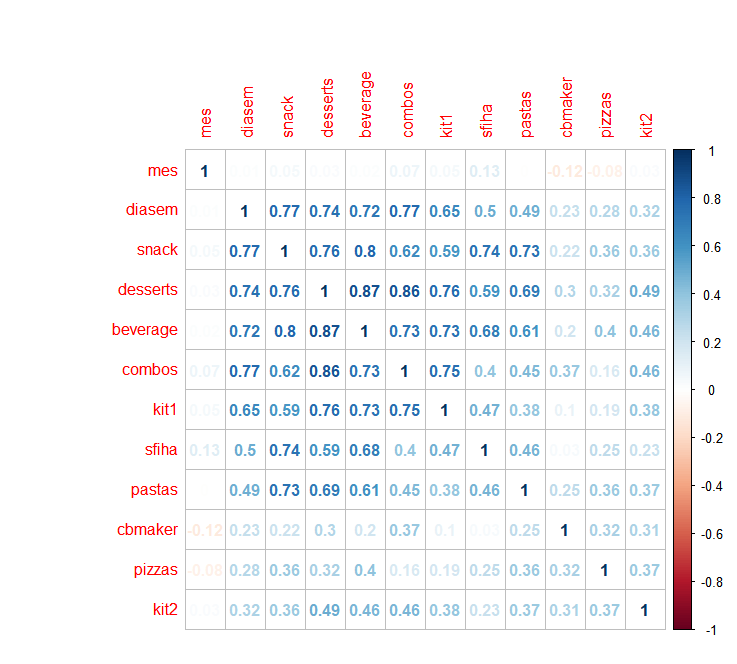
corrplot(forcorrplot)

corrplot(forcorrplot, upper="number", lower="color", order="hclust")

corrplot(forcorrplot, method="number",order="hclust" )

cormat(variables)”

We managed to see the correlation between the day of the week and month and the consumption of the several items:



As we can see the consumption of some items vary more them others as the days of the week change.

Now we have to see if there is correlation between the weather factors and the consumption by item:

Precipitac – Volume of rain in millimetres per square meter during the day

Tempmax – Max temperature during the day

Tempmin – Minimum temperature during the day

Tempmed – Average temperature during the day

Umidade – Level of humidity in the air

Insolacao – Level of sun

Remembering that doing the correlation we are going to identify the variable R which can varies from +1 to -1 indicating that a relationship exists between the two variables from absolute direct correlation (+1), no correlation (0) to inverse correlation (-1). Using R studio and the command “corrplot” we managed to see the following:

“setwd("/Curso-ML/Assignment-1/")

variables <- read.csv("data-06.csv")

summary(variables)

variables$diasemana <- NULL

variables1 <- variables[10:31]

summary(variables1)

library(corrplot)

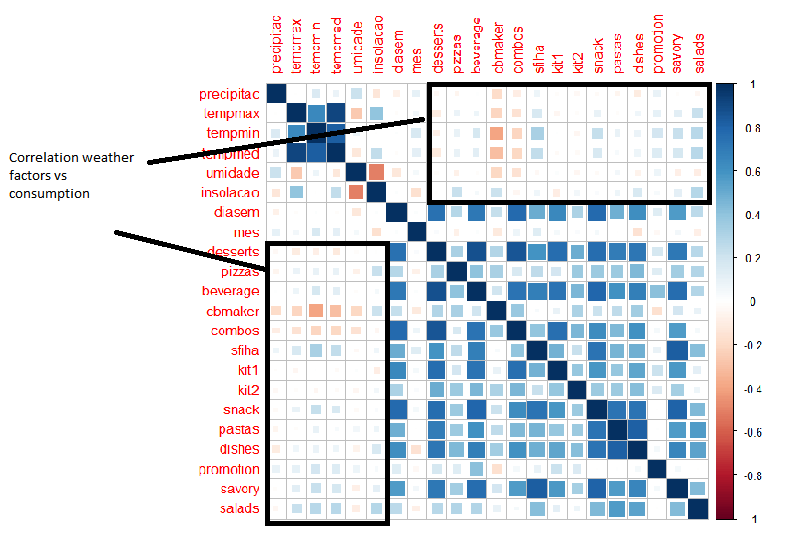
cor.test(variables1$diasem, variables1$pizzas)

cor.test(variables1$diasem, variables1$pizzas)

forcorrplot <-cor(variables1)

corrplot(forcorrplot)

corrplot(forcorrplot, method = "square")”



As we can see in the graphics there are some correlation between weather parameters and the consumption. However the weather conditions are not so determining as the day of the week and the month of the year.

Therefore building a model to guess the demand will imply the definition of the average demand of the day of the week in a given month and in sequence adjust this demand by the weather conditions.

To archive that we grouped the demand by the combination of type of day vs type of month with nine categories:



This analysis grouped the data as follows:



This would be the demand if only these two factors were influencing the actual sales, therefore the challenge is to identify how the weather conditions make the average demand deviate (up and down) from these values baseline factors.

**Identifying the demand**

Once we sliced the dataset by type of week and type of month we have to identify which one of the six weather parameters is more influential in adjusting the demand for each item (nine types).

To be able to do that we have to identify the correlation factor R² between each weather parameter and the volume sold. The factor with bigger R² is the ones to be used as “predictor” of the demand. We are going to use linear regression to predict the demand, identifying the parameters A and B in the formula Ax + B = Y where x is the chosen weather parameter and Y de expected demand for the product. This calculation will be done for each one of the nine types of products. For each row in the data\_test we do the following combinations of tests:





**The results**

When analysing how we should evaluate the results at the beginning the answer seemed to be very strait forward: How many times the system got the prediction right. However thinking a bit more some additional considerations appear:

1. Very unlikely that the prediction would be exactly right (How far from the mark seems to be a best measure.
2. We have nine predictions per cycle and have several cycles (Each day is a cycle in this context) , how many times it got within a giving limit seems to be a best measurement.
3. We have a current process to compare with, how many times it got better than the current one.

With all this in mind we produced the following spreadsheet:



As we can see the system got an uneven performance, being much better predicting some types of products, than others. In general this first version didn´t perform better than the manual process today in place.

**What could be done to improve the results**

Analyzing the results we start understanding the reasons why the performance wasn´t the one we expected. We identified at least four initiatives which if taken surely would improve the results :

1. We are segmenting the training data by type of day and type of month, doing that we create training subsets which sometimes were very small: E.g. the combination type of day 2 (Sunday) and month 2 (Dec and June) have only 8 samples (maybe less if part of them were segmented to the test data). Adding more historical data to the training data surely will improve the system performance. This problem tends to be solved overtime as new data is added to the system.
2. There is a problem with some of the sales volumes of our sample. Some products have a completely abnormal volumes of sales (too high or too low) . We did coded an Bell-curve tail with the objective of expurgating the samples which would distort the results. However, giving the problem 1 (lack of samples) we were unable to eliminate the adequate number of such cases; otherwise we would end up with even less samples.
3. We believe that there are some cross correlation between the weather parameters which maybe more effective as predictors than the parameters themselves. Due time constrains we didn’t explore this avenue.

A secondary issue which may or may not have relevance in this process is the bias regards under-producing. We realized that the real sales may hide a demand which wasn´t met due unavailability of products. That means we may have a day with an unusually low sales of a specific product. We don´t know and didn´t figured out a way of guessing how frequent this phenomenon may happens.

The bell-tail elimination process may deal with this if it was effectively implemented. It would work as long the phenomenon isn´t too much frequent.

**Annex 1**

Using this code in R we manage to get the average volumes of sales per type of product per range:

“setwd("/Curso-ML/Assignment-1/")

data <- read.csv("data-06.csv")

data1 <- read.csv("dia-sem-me.csv")

summary(data)

summary(data1)

print(data1)

len3=nrow(data1)

print(len3)

print(data1[1,4])

print(data[1,17])

library(stringr)

len2=nrow(data)

print(len2)

contador2 = 1

while (contador2<=len2){

print(contador2)

print(contador2)

print(contador2,17)

len3=nrow(data1)

print(len3)

contador1 = 1

while (contador1<=len3){

print(contador1)

print(data1[contador1,2])

print(data[contador2,17])

if (data1[contador1,2] == data[contador2,17] && data1[contador1,3] == data[contador2,18]){

print(data1[contador1,1])

data1[contador1,4]<-data1[contador1,4]+data[contador2,19]

data1[contador1,5]<-data1[contador1,5]+data[contador2,20]

data1[contador1,6]<-data1[contador1,6]+data[contador2,21]

data1[contador1,7]<-data1[contador1,7]+data[contador2,22]

data1[contador1,8]<-data1[contador1,8]+data[contador2,23]

data1[contador1,9]<-data1[contador1,9]+data[contador2,24]

data1[contador1,10]<-data1[contador1,10]+data[contador2,25]

data1[contador1,11]<-data1[contador1,11]+data[contador2,26]

data1[contador1,12]<-data1[contador1,12]+data[contador2,27]

data1[contador1,13]<-data1[contador1,13]+data[contador2,28]

data1[contador1,14]<-data1[contador1,14]+data[contador2,29]

data1[contador1,15]<-data1[contador1,15]+data[contador2,30]

data1[contador1,16]<-data1[contador1,16]+data[contador2,31]

data1[contador1,17]<-data1[contador1,17]+data[contador2,32]

data1[contador1,19]<-data1[contador1,19]+1

}

contador1 = contador1 + 1

}

contador2 = contador2 + 1

}

len4 = nrow(data1)

contador3 = 1

while (contador3<=len4){

print(contador3)

print(data1[contador3,2])

print(data1[contador3,1])

data1[contador3,4]<-data1[contador3,4]/data1[contador3,19]

data1[contador3,5]<-data1[contador3,5]/data1[contador3,19]

data1[contador3,6]<-data1[contador3,6]/data1[contador3,19]

data1[contador3,7]<-data1[contador3,7]/data1[contador3,19]

data1[contador3,8]<-data1[contador3,8]/data1[contador3,19]

data1[contador3,9]<-data1[contador3,9]/data1[contador3,19]

data1[contador3,10]<-data1[contador3,10]/data1[contador3,19]

data1[contador3,11]<-data1[contador3,11]/data1[contador3,19]

data1[contador3,12]<-data1[contador3,12]/data1[contador3,19]

data1[contador3,13]<-data1[contador3,13]/data1[contador3,19]

data1[contador3,14]<-data1[contador3,14]/data1[contador3,19]

data1[contador3,15]<-data1[contador3,15]/data1[contador3,19]

data1[contador3,16]<-data1[contador3,16]/data1[contador3,19]

data1[contador3,17]<-data1[contador3,17]/data1[contador3,19]

contador3 = contador3 + 1

}

print(data1)

##write.csv(data1,'/Curso-ML/Assignment-1/dia-sem-me.csv')

class(data$diasem)

print(nrow(data))”

**Annex 2**

This annex shows how the analysis was done for each type of product in order to identify which weather parameters is better to be used to predict it demand. We did each step for sfihas as a way to illustrate the process but it is done by the code for all types. Note that we did the comparison of which weather parameter suites better the prediction for each type of product for each type of combination day of week vs day of the month.

Day of week 1 – month 1

Sfihas

Coefficients:

Coefficients:

Estimate Std. Error t value Pr(>|t|)

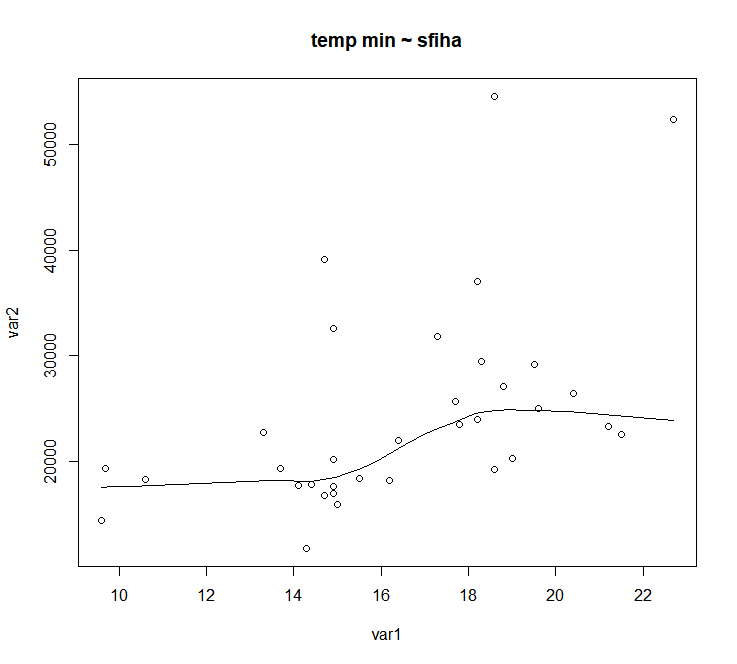
(Intercept) -664.1 7771.5 -0.085 0.93244

var1 1524.0 464.2 3.283 0.00249 \*\*

---

Here we have an linear equation = 1524 x temp med -664.1 = Volume of sfihas sold

That generates a scenario where when we apply the formula and compare the results with the real sales we have that 7 out of 34 samples show results with an error below 10% (Better than the current process). Therefore we have a model that got it right in 20% of the guesses (at least as right as today).



Day of week 2 – month 1

Sfihas

Coefficients:

Estimate Std. Error t value Pr(>|t|)

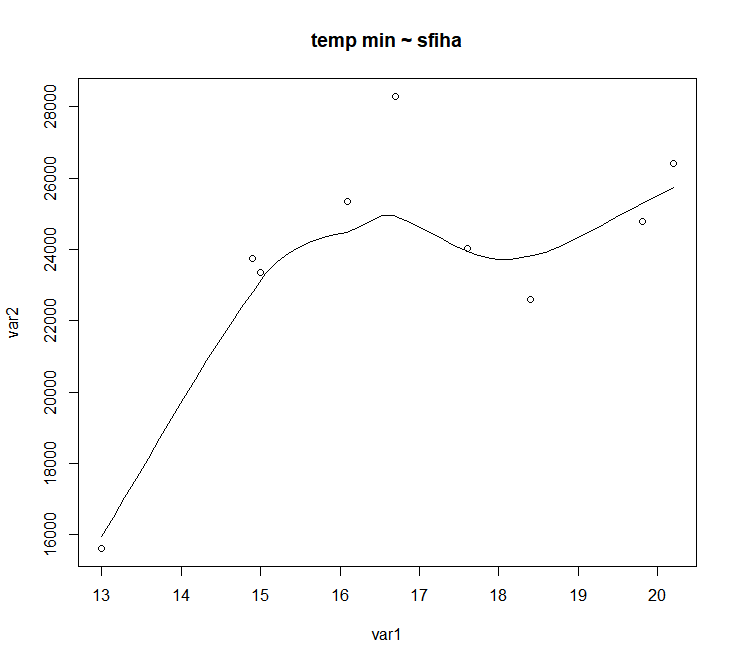
(Intercept) 8780.8 7531.3 1.166 0.2818

var1 890.9 442.9 2.012 0.0842 .

---

Here we have an linear equation = 890.9 x temp min + 8780,8 = Volume of sfihas sold

That generates a scenario where when we apply the formula and compare the results with the real sales we have that 6 out of 9 samples show results with an error below 10% (Better than the current process). Therefore we have a model that got it right in 67% of the guesses (at least as right as today).



Day of week 3 – month 1

Sfihas

Coefficients:

Estimate Std. Error t value Pr(>|t|)

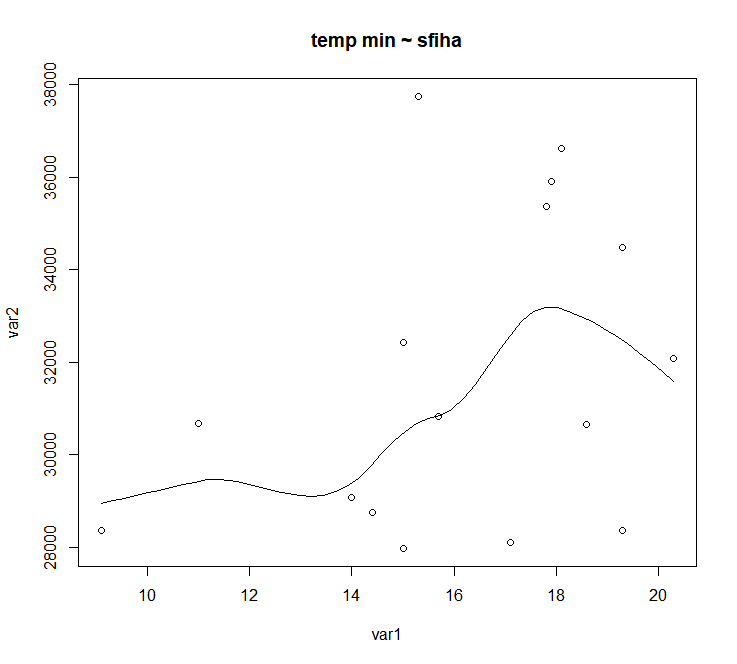
(Intercept) 25417.8 4455.2 5.705 5.44e-05 \*\*\*

var1 390.8 271.8 1.438 0.173

---

Here we have an linear equation = 390.8 x temp min + 25417.8 = Volume of sfihas sold

That generates a scenario where when we apply the formula and compare the results with the real sales we have that 10 out of 16 samples show results with an error below 10% (Better than the current process). Therefore we have a model that got it right in 62% of the guesses (at least as right as today).



Week 1 – month 2

Sfihas

Coefficients:

Estimate Std. Error t value Pr(>|t|)

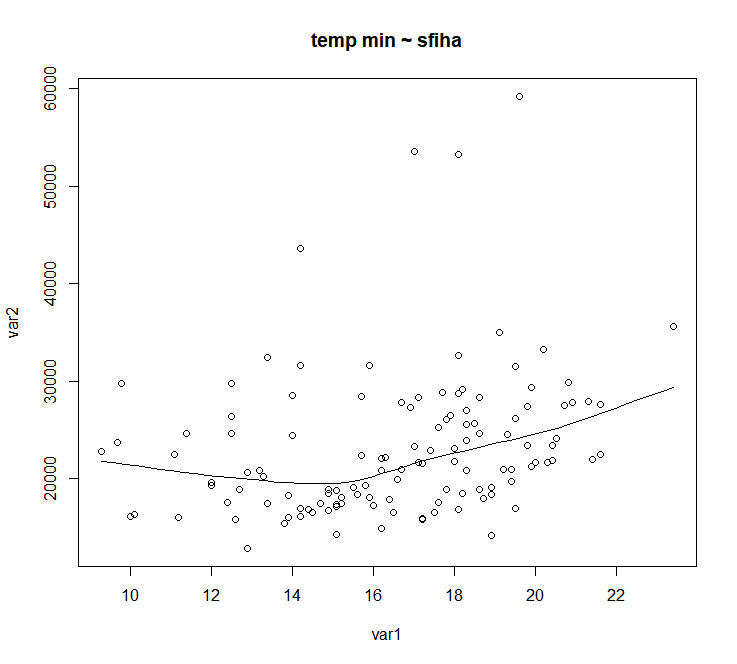
(Intercept) 12182.0 3665.8 3.323 0.00118 \*\*

var1 664.0 218.2 3.043 0.00287 \*\*

---

Here we have an linear equation = 664 x temp min + 12182 = Volume of sfihas sold

That generates a scenario where when we apply the formula and compare the results with the real sales we have that 30 out of 123 samples show results with an error below 10% (Better than the current process). Therefore we have a model that got it right in 25% of the guesses (at least as right as today).



Week 2 – month 2

Sfihas

Coefficients:

Estimate Std. Error t value Pr(>|t|)

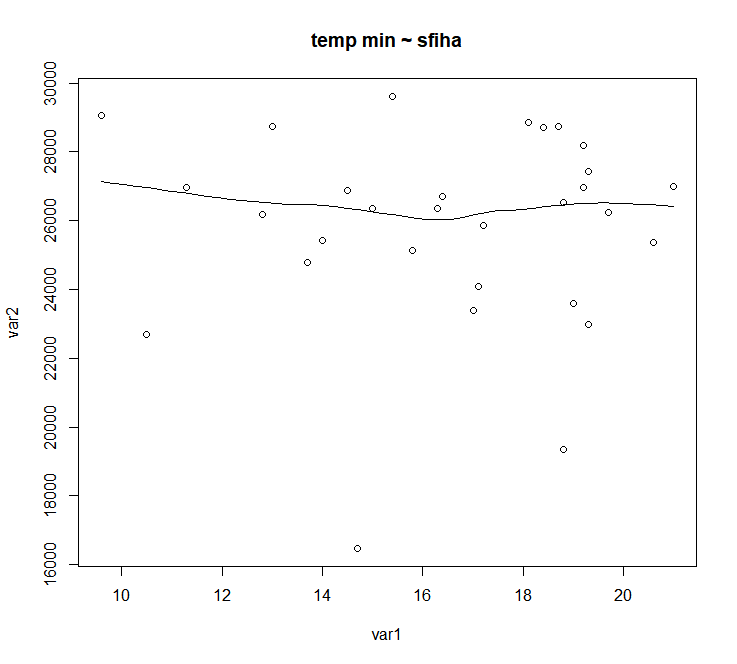
(Intercept) 25865.055 2978.734 8.683 1.97e-09 \*\*\*

var1 -2.627 177.843 -0.015 0.988

---

Here we have an linear equation = -2.627 x temp min + 25865 = Volume of sfihas sold

That generates a scenario where when we apply the formula and compare the results with the real sales we have that 19 out of 30 samples show results with an error below 10% (Better than the current process). Therefore we have a model that got it right in 55% of the guesses (at least as right as today).



Week 3 – month 2

Sfihas

Coefficients:

Estimate Std. Error t value Pr(>|t|)

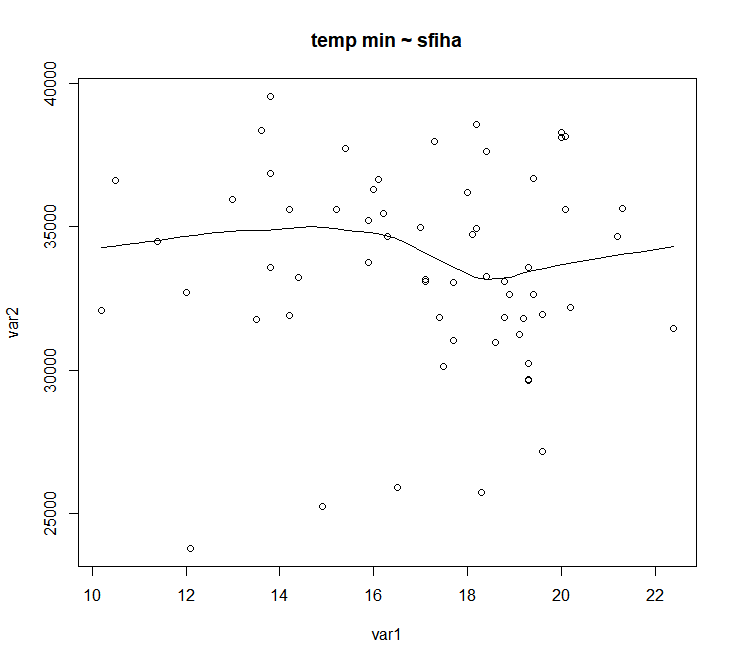
(Intercept) 34328.14 2776.22 12.365 <2e-16 \*\*\*

var1 -45.79 160.84 -0.285 0.777

---

Here we have an linear equation = -45.79 x temp min + 34328 = Volume of sfihas sold

That generates a scenario where when we apply the formula and compare the results with the real sales we have that 43 out of 61 samples show results with an error below 10% (Better than the current process). Therefore we have a model that got it right in 70% of the guesses (at least as right as today).



Week 1 – month 3

Sfihas

Coefficients:

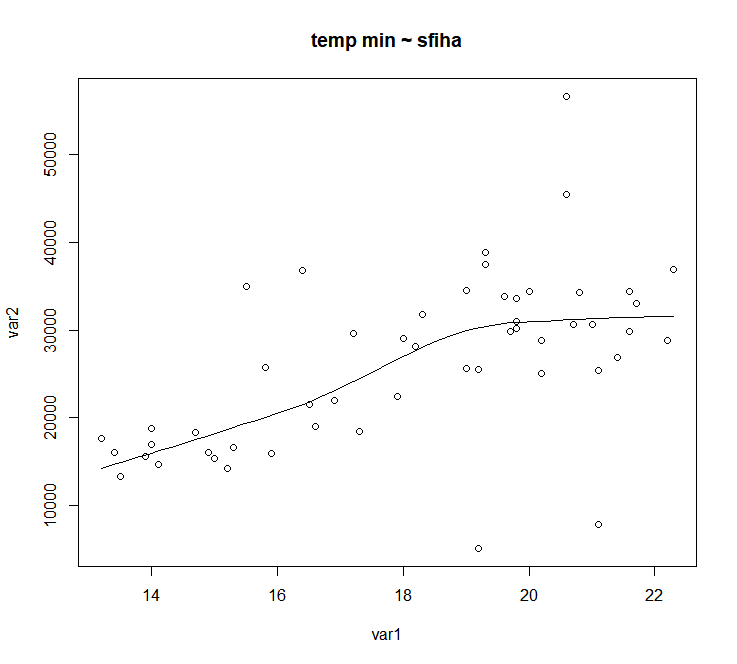
Estimate Std. Error t value Pr(>|t|)

(Intercept) -11381.9 7489.5 -1.520 0.135

var1 2075.5 408.7 5.078 5.67e-06 \*\*\*

Here we have an linear equation = 2075 x temp min – 11.381,9 = Volume of sfihas sold

That generates a scenario where when we apply the formula and compare the results with the real sales we have that 18 out of 52 samples show results with an error below 10% (Better than the current process). Therefore we have a model that got it right in 34% of the guesses (at least as right as today).



“scatter.smooth(x=var1, y=var2, main="temp min ~ sfiha") # scatterplot”

Week 2 – month 3

Sfihas

Coefficients:

Estimate Std. Error t value Pr(>|t|)

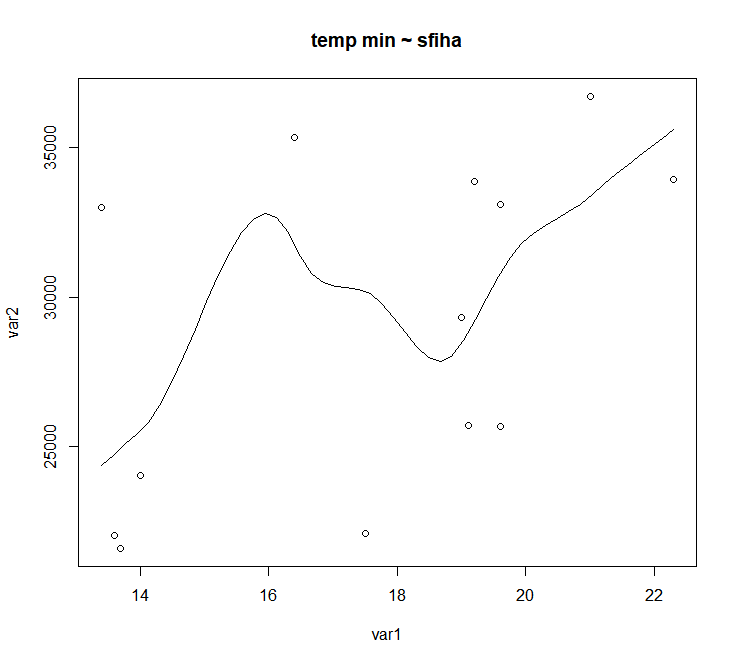
(Intercept) 12413.3 8488.7 1.462 0.1716

var1 941.6 476.6 1.976 0.0738 .

---

Here we have an linear equation = 941.6 x temp min +12413.3 = Volume of sfihas sold

That generates a scenario where when we apply the formula and compare the results with the real sales we have that 5 out of 13 samples show results with an error below 10% (Better than the current process). Therefore we have a model that got it right in 38% of the guesses (at least as right as today).



Week 3 – month 3

Sfihas

Coefficients:

Estimate Std. Error t value Pr(>|t|)

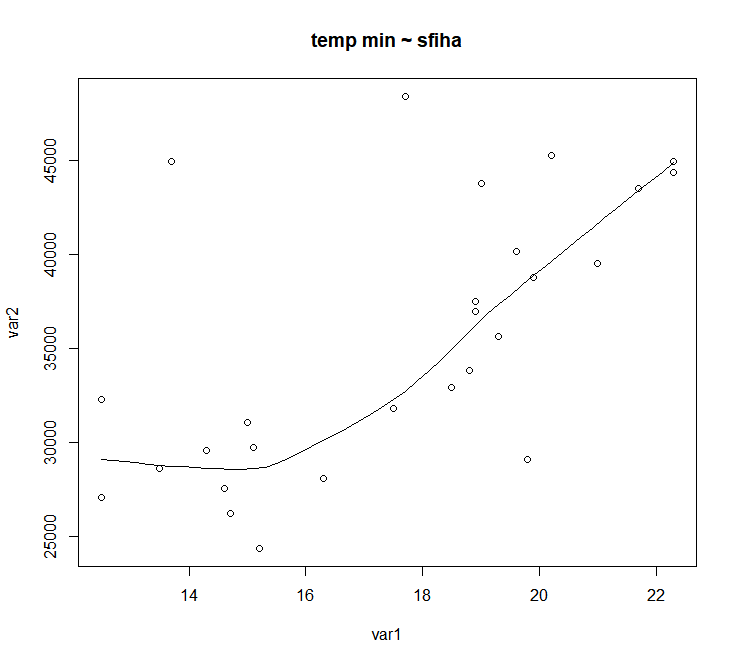
(Intercept) 8939.2 6418.1 1.393 0.175944

var1 1511.7 361.4 4.183 0.000309 \*\*\*

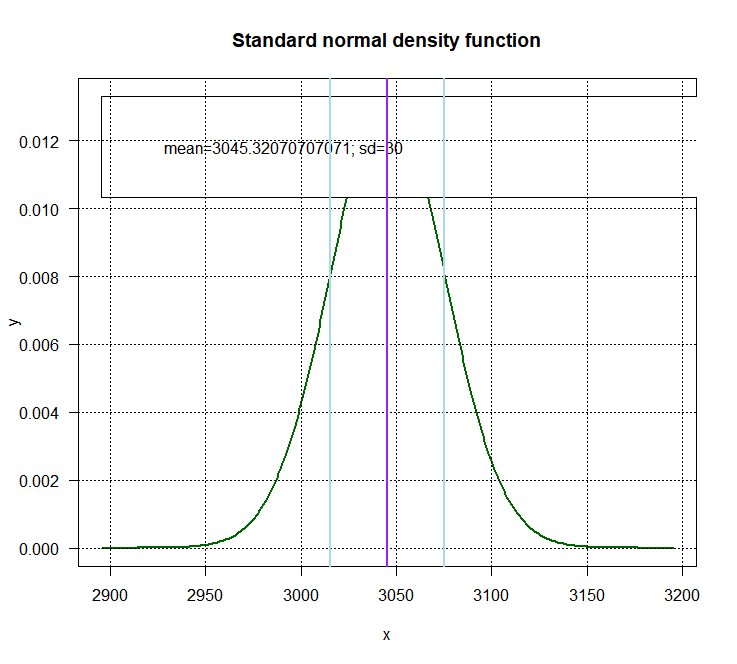
---

Here we have an linear equation = 1511.7 x temp min + 8939.2 = Volume of sfihas sold

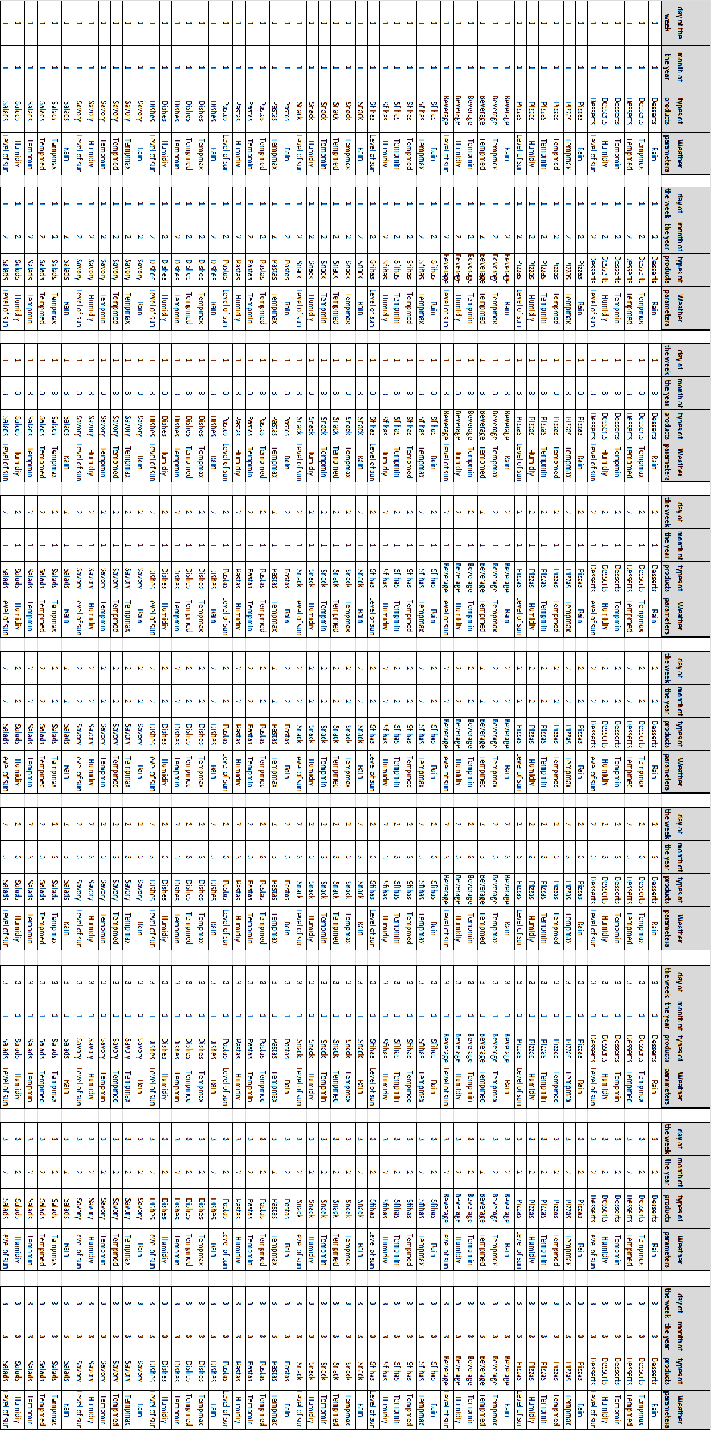
That generates a scenario where when we apply the formula and compare the results with the real sales we have that 5 out of 27 samples show results with an error below 10% (Better than the current process). Therefore we have a model that got it right in 18% of the guesses (at least as right as today).



**Annex 3**



**Annex 4 (combinations)**

****