Programming Project: Decision Support System at Fuji Food Services

Project Overview

Fuji Food Services delivers food to businesses and corporations in the metropolitan area of Hartford. Fuji's business model is based on Tamago-Ya's model, a Japanese company that produces and delivers high-quality lunch (bento) boxes at a low price of about $5.60 per box to office workers in the Tokyo metropolitan area.

The idea is that orders for lunchboxes will be received every day from 9 to 10:30 am, and then delivered by noon. Notice that the lead time for production and delivery of the last order could be as little as 90 minutes. Hence, all orders should be quickly entered and once they are compiled, Fuji must promptly decide the best way to deliver the orders, taking into consideration that its marketing strategy relies on the promise of delivery on time. To cope with demand, Fuji has divided the Hartford area into eight disjoint regions and setup distribution centers in each region to prepare and deliver the lunch boxes. These centers are located at Bloomfield, Farmington, Glastonbury, Hartford, Manchester, Southington, West Hartford, and Windsor, respectively.

Orders are compiled according to the region where they are placed. Each lunch box contains six items, most of which are made from organic and natural ingredients. Fuji publishes its menu both on its website and on a sheet of paper handed out to customers. Typical customers are corporate, which would buy many lunch boxes, or individuals, who would only buy one box.

Decision Support System

Fuji wishes to create a decision support system (DSS) to help with two important aspects in their daily planning:

Demand monitoring: To better plan for ingredient procurement from suppliers, Fuji wants to keep track of daily historic demand. Concretely, the system should provide a friendly user-interface to enter orders into a database to record and aggregate the usage of different ingredients.

Demand forecasting: Since Fuji promises high quality, made of fresh ingredients, lunchboxes, it is very important to estimate the ingredients required for the next seven days and timely order the appropriate amounts from the suppliers. Demand is sensitive to weather changes. For example, they receive more orders on rainy or very cold days since people do not want to go out for lunch. To forecast demand, Fuji has prepared a data set consisting of the number of boxes ordered per region/distribution center for the past four months. Concretely, the data set contains the following fields:

[1] Adapted from Whang et al. (2010).

- **Date**: Calendar date when the observation was taken, in mm/dd/yy format.
- **Day**: Name of the observation occurrence day.
- **Temperature**: Categorical variable that classifies the observation as cold, cool, mild, warm, hot, or very hot, depending on the average temperature from 9 a.m. to 12 m. every day.
- **Humidity**: Categorical variable that classifies the observation as very dry, dry, humid, or very humid, depending on the average humidity from 9 a.m. to 12 m. every day.
- **Festive Day?**: Categorical variable that classifies the observation as yes or no; "yes" means that the corresponding day was a holiday or a festive day when workers are not likely to order food delivered at work. Otherwise, it is classified as "no".
- **Boxes per Region**: boxes ordered per region on the corresponding day.

The data can be found in the Excel file entitled "*fujidata2016.xlsx*," available on Husky CT. The goal is to develop a system to forecast demand for boxes for each region for the next seven days.

Project Specifications/Requirements

1. Design a user-form interface to let Fuji enter a summary of the daily orders as indicated in the file entitled "fujidata2016.xlsx", on the "Orders Summary" tab. Your interface will use a user form whose input fields and required VBA controls are specified in Table 1 from Appendix 1. The program will add the number of boxes ordered on one day in each of the eight regions to the data set on the "Orders Summary" tab. It should also allow the user to input the temperature, humidity, and whether the day is festive. The new data must be entered in the same format as the data on the tab. The date associated with the new data (row) must be the next calendar day following the last entry in the data set (this should be done automatically by the program, the user will not enter the date or day name). The form must unload after entering the data for one day. Your program should check that the user inputs correct values into the data fields as described in Appendix 1. If an invalid value is entered, you should display an error message. You may also use the properties of the VBA controls on your form to enforce that only valid values are input.

2. Using the provided data set, prepare eight multiple regression models, one for each of the eight regions, respectively. In each model the dependent (Y) variable is the boxes ordered in the corresponding region and the independent (X) variables are "date", "temperature", "humidity", and "festive day?". Notice that you have to transform the variable "date" into a time period index, and the other three independent variables must be transformed into dummy variables. Since for the "temperature" field there are no values for "hot" and "very hot", you may ignore them in your regression models. To summarize your results, report the regression coefficients, the significance F value, and the MSE for each of the eight models.

3. Use the "day" variable and the regression models from the previous item to compute seven seasonal indices, one for each day from Monday to Sunday, and for each region. Notice that you will end up with eight sets of seven seasonal indices each, one set per region. Adjust your regression models using the corresponding set of indices and report the eight sets and the corresponding revised MSE for each region.

4. Prepare a user form that will show forecasting intervals for the expected number of boxes on each of the following seven days after the last day from the "Orders Summary" sheet. To do so, the user will first input temperature and humidity forecasts for the next seven days. The user will also indicate if any of the next seven days will be a festive day. The form will have a button entitled "Compute Forecasts". When the user presses this button, the program will ask the user to select a region and then, a new form will show a list box where the seven-day forecast for the chosen region will be listed. Each row of the list box will contain the date and the lower and upper bounds of the forecast interval for that date. The forecast intervals must be computed by using the regression models of the data as indicated in items 1-2 above. You are free to design the controls on these forms any way you want, but your program should check that the user inputs correct values into the data fields as described above. If an invalid value is entered, you should display an error message. You may also use the properties of the VBA controls on your form to enforce that only valid values are input.

5. Add a button entitled "Revise Models" to your form from the previous item. When the user presses this button, your program must re-run all the eight regression models created in items #2 and #3 above (also re-computing all the seasonal indices and MSEs) by using the entire data set in the "Orders Summary" tab (including the original data and the newly added data). The goal is to update the regression coefficients and seasonal indices as new data is added.

6. After completing the computations from the previous item, the revised coefficients, significance F value, seasonal indices, and MSE for each of the eight regions must be displayed in a table on a sheet that must be named "Models Summary". Notice that the table must have eight rows, one row per region. The form must unload after completing this table.

7. Add two buttons to any sheet to start and execute the two main forms from item #1 and item #4 above.

Project Grading

The project will be graded according to the following parameters:

Meets specifications	**40%**
User-friendly	**40%**
Creativity/Aesthetic	**10%**
Code clarity/modularity	**10%**
Total	**100%**

Specifications: the program meets the requirements indicated above such as correct handling of events, correct updating of tables, correct outputs, etc.
User-friendly: the program is reasonably easy and fast to use for an operator not very familiar with Excel.
Creativity/Aesthetic: the layout of the user forms and spreadsheets is pleasing to the eye, elegant, functional, and exhibits creativity.
Code clarity/modularity: the program code is easy to understand and exhibits some modularity.

References

1. J. Whang, S. Fushimi, J. Kaminsky, V. Rocha, and J. Tsou. *Tamago-ya of Japan: Delivering Lunch Boxes to Your Work*. Stanford Graduate School of Business, September, 2010.

Appendix 1. Data to Be Collected

Field	Subfield	Definition	Required VBA Control
Date		mm/dd/yy format	Spin buttons or combo boxes
Day		Monday, Tuesday, … , Sunday	Combo box
Type of day			
	Temperature	Cold, cool, mild, warm, or hot	Option button
	Humidity	Very dry, dry, humid, or very humid	Option button
Holiday?		yes/no	Check box
Boxes per Region			
	Bloomfield	Numerical value ranging from 1 to 300	Scroll Bar
	Farmington	Numerical value ranging from 1 to 300	Scroll Bar
	Glastonbury	Numerical value ranging from 1 to 300	Scroll Bar
	Hartford	Numerical value ranging from 1 to 300	Scroll Bar
	Manchester	Numerical value ranging from 1 to 300	Scroll Bar
	Southington	Numerical value ranging from 1 to 300	Scroll Bar
	West Hartford	Numerical value ranging from 1 to 300	Scroll Bar
	Windsor	Numerical value ranging from 1 to 300	Scroll Bar

Table 1. Data field descriptions to input daily orders summary.