AutonomousShipment business analysis report

Introduction

AutonomousShipment, a Leeds-based start-up, uses robot drones for parcel delivery. This project will evaluate the outcome within one month, aiming to encompass a wide range of potential customers across various stores. This project decides to choose the best robot prototype from four based on requirements. Additionally, it is essential to find the number of robots to allocate across the three stores to meet the trial's goal and constraints.

The details of weight and criteria

In terms of selecting the robot prototypes, the company developed four robot prototypes: Archer (Robot A032), Bowler (Robot B23), Corner (Robot CJKL), and Deviant (Robot DSXX). To find the best robot to operate this project based on five criteria provided in Table 1 with different weights.

Criteria of Robots	**Meaning of Criteria**	**The Maximum or Minimum value is preferred**
Carrying Capacity	The robot individual carrying capacity, measured in litres.	Maximum
Battery Size	The robots' battery capacity, measured in hours of operation.	Maximum
Average Speed	The average velocity of robot, measured in kilometres per hour.	Maximum
Cost per Unit	Cost per unit of each robot in GBP.	Minimum
Reliability	Estimate operating hours until breakdown.	Maximum

Table 1: Evaluation Criteria for Autonomous Delivery Robots

The management team provides a level of importance for all criteria. That information could be used to assume the weight of each criterion in Table 2, which focuses on robot reliability and a minor focus on average speed.

Criteria of Robots	**Level of Important**	**Weight (%)**
Carrying Capacity	4th	15
Battery Size	3rd	20
Average Speed	5th	10
Cost per Unit	2nd	25
Reliability	1st	30

Table 2: Weight Distribution for Robot Selection Criteria

Method of business analysis (finding the best robot prototypes)

Multiple Criteria for Decision-making Analysis (MCDA) might be used in robot selection because it can assist users in choosing the best options in various object criteria with different weights. MCDA also has many decision-making and problem-solving methods that can be selected to solve different problems.

The Technique for Order of Preference by Similarity to the Ideal Solution (TOPSIS) was selected to find the best robot prototypes. TOPSIS (one of the MCDA methods) is the decision-making method that could identify the best option based on how close it is to the best choice (positive ideal solution) and how far it is from the worst (negative ideal solution).

The reason for choosing TOPSIS is that it can be used to balance the best and worst scenarios. Moreover, TOPSIS could reduce bias from the process of normalisation. Furthermore, TOPSIS is more complex but can provide a more nuanced view than the Weight Sum Method (WSM). Therefore, TOPSIS might be the best choice to handle this robot-selection process.

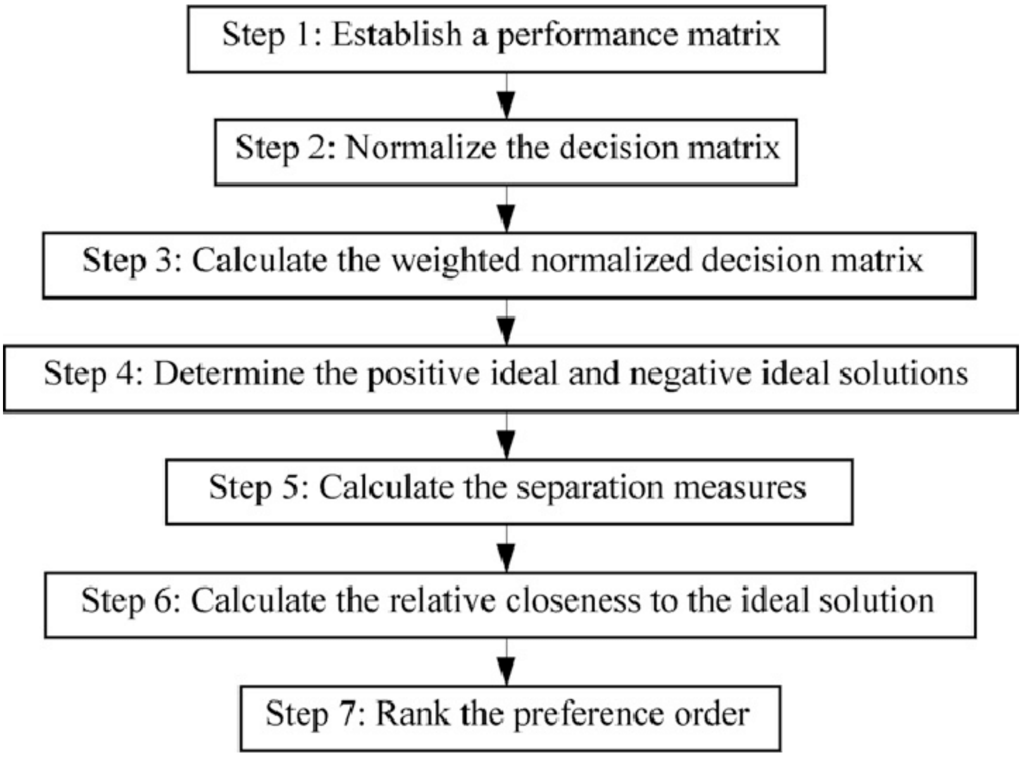

Step 1: Establish a performance matrix
Step 2: Normalize the decision matrix
Step 3: Calculate the weighted normalized decision matrix
Step 4: Determine the positive ideal and negative ideal solutions
Step 5: Calculate the separation measures
Step 6: Calculate the relative closeness to the ideal solution
Step 7: Rank the preference order

Figure 1: The Step of Operating TOPSIS Prototypes (Lozano, 2013)

Robot Prototypes	**Carrying Capacity (liters)**	**Battery Size (hours)**	**Average Speed (km/h)**	**Cost Per Unit (GBP)**	**Reliability (hours)**
Weight	0.15	0.2	0.1	0.25	0.3
Archer	45	18	6	5210	22
Bowler	50	18	4	6250	24
Corner	60	12	4	4500	24
Deviant	40	24	10	7100	32

Table 3: Decision Metrics for Evaluating Robot Prototypes

Figure 1's TOPSIS steps begin with listing all alternatives and criteria and creating the decision matrix in Table 3. Secondly, use vector normalisation to compare values across criteria and multiply by weights to get a Weighted Normalised Matrix. Thirdly, calculate distances from positive ideal solution (PIS) and negative ideal solutions (NIS) demonstrated in Table 4. Finish by calculating the score of each robot based on its proximity to the ideal solution and ranking it.

Robot Prototypes	**separation from PIS**	**separation from NIS**
Archer	0.078	0.055
Bowler	0.084	0.042
Corner	0.092	0.064
Deviant	0.063	0.099

Table 4: The Separation Distance from PIS and NIS

Result (finding the best robot prototypes)

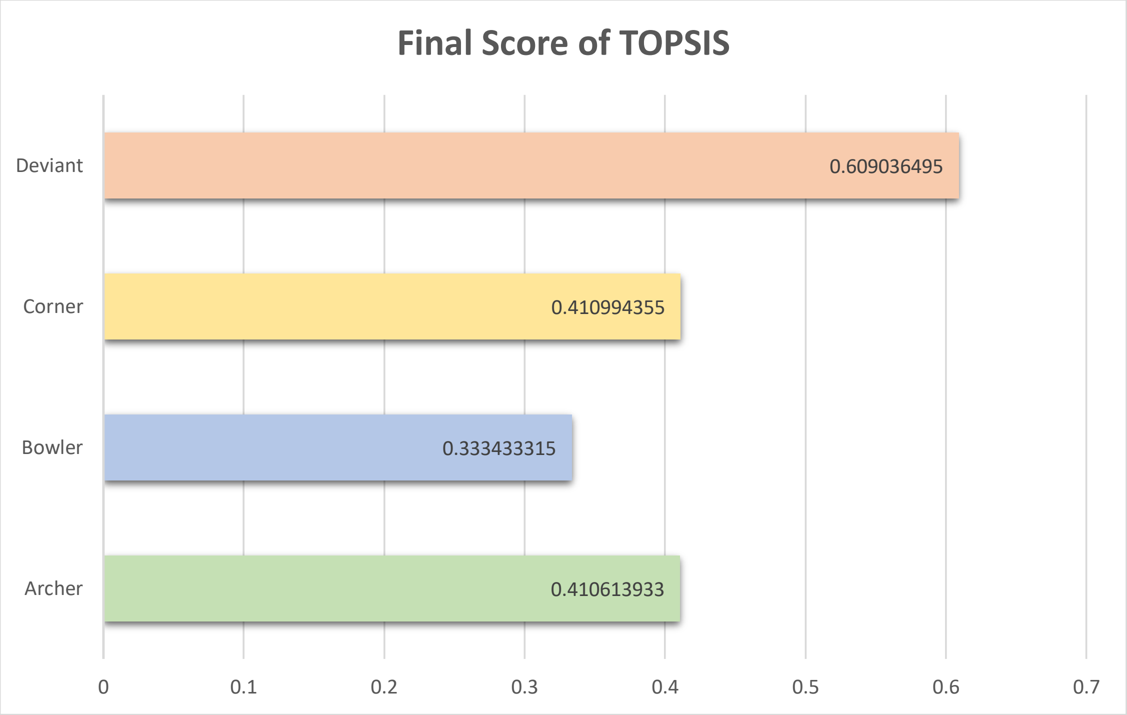


Figure 2: Bar Chart of TOPSIS Final Scores for Robot Prototypes

After applying TOPSIS, Figure 2's bar chart, showcasing all four robot prototypes, reveals the scores. The 'Deviant' prototype leads with a score of about 0.609, while the 'Bowler' prototype trails with the lowest score, approximately 0.333. The 'Corner' and 'Archer' prototypes have similar scores, both around 0.410.

In summary, the TOPSIS method assists decision-making by comparing options against ideal and negative solutions. For this project, the 'Deviant' prototype scores highest in TOPSIS, suggesting it is closest to the ideal and potentially the best choice among the prototypes. Its details, including a Cost Per Unit of 7100 GBP, are in Table 3. This cost data is crucial to determine the number of 'Deviant' robots for the trial, ensuring effective allocation across the three stores to meet the trial's goals and constraints.

Method of business analysis (finding the number of robots)

Regarding finding the number of robots, the company's primary aim is to ensure the project's testing covers all objectives, with less emphasis on profit. This involves determining the number of robots needed for three store types: Grocery, Clothing, and Sports Equipment Stores.

This start-up company aims to maximize daily order completion using Deviant robots, with target delivery rates of 9, 6, and 4 orders per day for Grocery, Clothing, and Sports Equipment Stores, respectively, as detailed in Table 5.

Goal to maximise the daily order completion rate using Deviant robots.	
Type of store	**The estimate number of order for each robot (order per robot per day)**
Grocery Stores	9
Clothing Stores	6
Sports Equipment Stores	4

Table 5: The Description of the Goal in Different Store Types.

The project's constraints, shown in Table 6, include the three main constraints. The challenge is determining the feasible number of robots per store under these constraints.

Constraints	**Store type**	**Number that needs to cover the constrain**
The number of robots **(robots)**	Grocery Stores	>= 5
	Clothing Stores	>= 5
	Sports Equipment Stores	>= 5
The number of technician staff should not more than 250 hours per week **(hours per robot per week)**	Grocery Stores	10
	Clothing Stores	7
	Sports Equipment Stores	5
The robot operation cost sum with the cost of robot should not more than 250000 GBP per month **(GBP per robot per month)**	Grocery Stores	8700
	Clothing Stores	8100
	Sports Equipment Stores	8700

Table 6: The Description of the Constraints in Different Store Types.

Linear optimisation could be the best option to solve this problem because it is the method that can be used to solve the MCDA problem to maximise or minimise the target. The linear optimisation method could solve the function to maximise robot daily orders to solve this business problem. Linear optimisation can handle the trial's constraints, including budget, minimum number of robots per store, and technician staff hours.

Goal programming is the adaptation of linear optimisation, which uses summed deviational variables, and the linear solver minimises the sum of the deviations to find a strategy closest to the goals. Goal programming could be the best option for this problem because this business problem does not focus on weight or goal priority levels. Therefore, goal programming may be used instead of Weighted and Lexicographic Goal Programming.

Type of variables	**Definition of variables**
Decision Variable	X_1 = Number of robots in Grocery Stores
	X_2 = Number of robots in Clothing Stores
	X_3 = Number of robots in Sports Equipment Stores
Deviation Variables	d_{1-}, d_{1+}, d_{2-}, d_{2+}, d_{3-}, d_{3+}, d_{4-}, d_{4+}, d_{5-}, d_{5+}, d_{6-}, d_{6+} = deviational variables

Table 7: The Explanation Variable Which Used to Perform Goal Programming Analysis

Goal and Constraints	**Linear Optimisation Equation**
Goal	Max Order number: $9 * X_1 + 6 * X_2 + 4 * X_3 + d_{1-} - d_{1+} >= 95$ (Assume the number of minimum order might be >= 95 from input the number of robot in each store as 5 in equation)
Constraints	Technician hours: $10 * X_1 + 7 * X_2 + 5 * X_3 + d_{2-} - d_{2+} <= 250$
	Budget: $8700 * X_1 + 8100 * X_2 + 7700 * X_3 + d_{3-} - d_{3+} <= 250000$
	Number of robots operating in Grocery Stores: $X_1 + d_{4-} - d_{4+} >= 5$
	Number of robots operating in Clothing Stores: $X_2 + d_{5-} - d_{5+} >= 5$
	Number of robots operating in Sports Equipment Stores: $X_3 + d_{6-} - d_{6+} >= 5$
	While $d_{1-}, d_{1+}, d_{2-}, d_{2+}, d_{3-}, d_{3+}, d_{4-}, d_{4+}, d_{5-}, d_{5+}, d_{6-}, d_{6+} >= 0$

Table 8: The Linear Optimisation Equation Used to Perform Goal Programming Analysis

The decision, goals, constraints, and deviational variables must be defined first to solve this problem (Table 7). Secondly, create the goal programming model parameters and model (Table 8). Thirdly, solve the problem and review the results.

Result (finding the number of robots)

According to Table 9, the goal programming result (solved by R) provides that the number of robots to use in this trial is 29. Moreover, this shows that the maximum number of orders in one day could be 221.

The number of robots which can cover all goal and constraints of the trail.	
Type of store	**The estimate number of robot (robots)**
Grocery Stores	19
Clothing Stores	5
Sports Equipment Stores	5

Table 9: The Result from Solving Goal Linear Programming by “goalp” Function in R

Conclusion

To conclude, business analytic methods provide the Deviant prototype, a robot prototype that may be the best choice, considering the positive and negative ideal solutions of the Deviant prototype. This project's goal and constraints can be accomplished with 19, 5, and 5 robots in the Grocery Store, Clothing Store, and Sports Equipment Store, respectively. The maximum number of orders per day is 221 orders. The limitation of this analysis is that it uses assumptions in the calculation, such as weight, which might lead to the wrong result.

Drinks@home.uk customer analysis report

Introduction

Drinks@home.uk is an e-commerce website that sells alcoholic and non-alcoholic beverages from around the entire world. The data obtained from this organisation has been documented and analysed regarding 400 customers to understand them better and plan the following marketing campaign. The primary issues that require resolution involve identifying the key factors that impact customer spending behaviour on the website and determining the optimal choice from the available options for the upcoming marketing campaign to improve profitability.

The detail of customer data

The data of 400 customers is provided in the file "Transactions_Customer.csv", including the six factors of each customer on the website. The details of each factor are demonstrated in Table 10.

Factor of Transactions Customers	**Explanation**
Revenue (GBP)	The revenue from the last order that purchased by this customer.
Advertisement Channel (Channel 1, 2,3 or 4)	The Advertisement Channel that brought the customer to website Including 4 channel - Channel 1: Leaflet - Channel 2: SocialMedia - Channel 3: SearchEngine - Channel 4: Influencer
Estimated Age (years)	The estimated age of customer from website tracking software.
Estimated Income (GBP)	The estimated income of customer from website tracking software.
Time on website per week (Seconds)	The estimated average time that customers spending time on website per week.
Seen Voucher (Yes or No)	Do customers has seen any discount voucher popup.

Table 10: The Factor of Transactions Customer on 'Drinks@home.uk' Website

The method of solving the problem

The linear regression model may be the most suitable approach for identifying the determinants influencing customers' purchasing decisions on the website. This is because linear regression can function as an explanatory model, revealing positive and negative relationships between explanatory variables and the dependent variable. The linear regression model is used in this case to identify the direction and relationship between customer revenue and potential influencing factors.

- **Data understanding and preparation**

The first step for creating the linear regression model is rechecking the data quality of the "Transactions_Customer.csv" data to find some missing values. After rechecking the dataset, it was found that there are no missing values present. Consequently, this dataset appears suitable for subsequent analysis.

The second step in creating this business analysis model is setting the dummy variables. These variables can take values of either 0 or 1 and are employed to symbolise the categorical variables within this model. In this case, two categorical variables exist: Advertisement Channel and Seen Voucher. These variables are displayed in Table 11. Regarding the Advertisement channel, it is necessary to divide the variable into four columns and assign the values 0 or 1, representing "no" or "yes", respectively. As regards the Seen Voucher, the value is already set as 0 and 1.

Categorical variable	**Dummification of categorical variable**
Advertisement Channel	Separate 4 channel to 4 new columns and add to data set that are: Advertisement_Channel_1 Advertisement_Channel_2 Advertisement_Channel_3 Advertisement_Channel_4
Seen Voucher	The variables are already provided in form of dummy variables (0 and 1)

Table 11: The explanation of Dummification of Categorical Variable

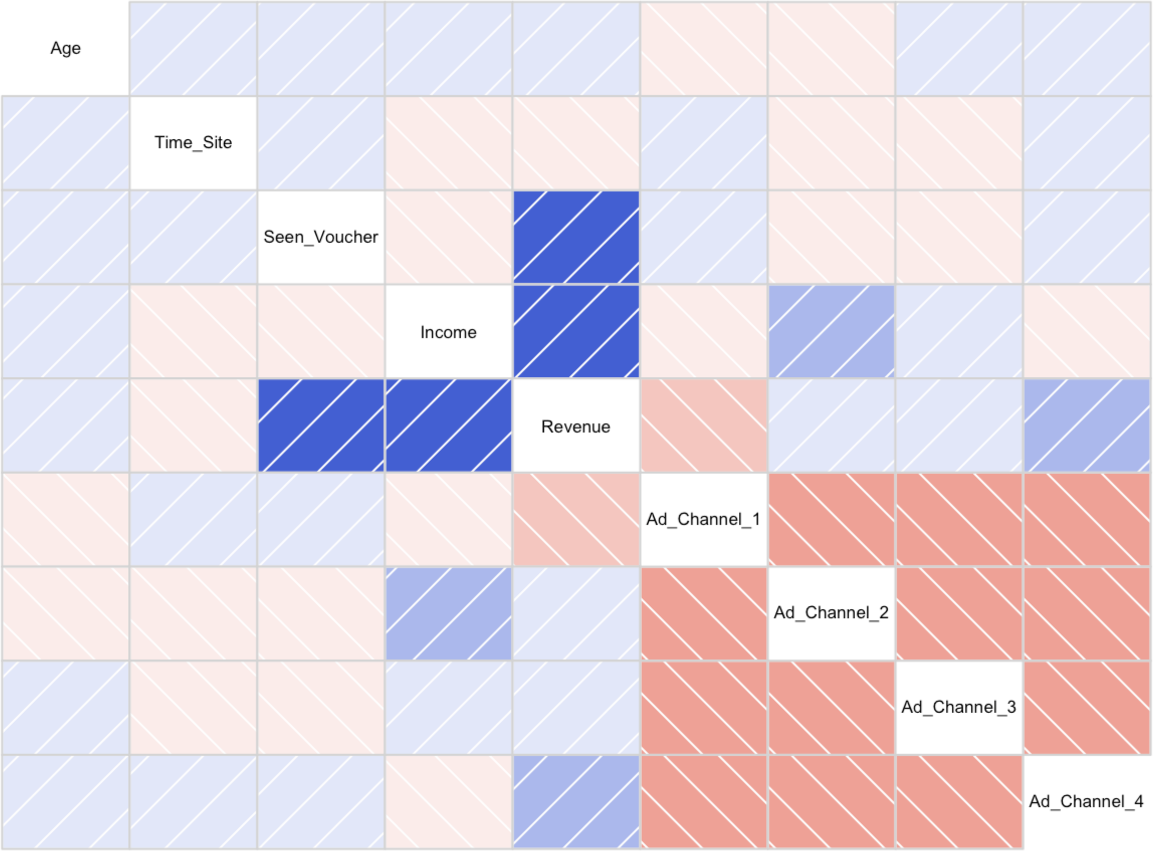


Figure 2: The Correlation of All Numerical Variables in the Dataset.

The third step is performing a correlation analysis, typically employed to examine the relationships between numerical variables. The variables exhibited a strong positive correlation, indicating that when one value is high, the other is also high. The negative correlation operates in opposite directions. In this dataset, the correlation checking is provided in Figure 2, in which the red indicates the negative correlation and the blue demonstrates the positive correlation; the intensity of colour represents the strong correlation. This report concentrates on the Revenue factor and presents the correlation between the Revenue factor and other variables in Table 12. The time customers spend on the site and the influence of Advertisement Channel1 (Leaflet) display negative relationships, while the other variable demonstrates a positive relationship.

The correlation of revenue to other variable	
Variable	**Correlation value**
Estimated Age	0.0263
Time on website per week	-0.0283
Seen Voucher	0.4660
Estimated Income	0.5317
Revenue	1.0000
Advertisement Channel 1	-0.2264
Advertisement Channel 2	0.0512
Advertisement Channel 3	0.0283
Advertisement Channel 4	0.1468

Table 12: The Correlation of Revenue that Affects the Other Values

The fourth step is reviewing the two theoretical assumptions of the linear regression model to determine if our data fits the linear regression model. The outcome of the testing is presented in Table 13. Overall, two assumptions do not fit as well as a theory assumption to be an excellent linear regression model. Nevertheless, it is reasonable to assume that this dataset can be analysed using a linear regression model because it is an appropriate technique to identify the relationship between data factors.

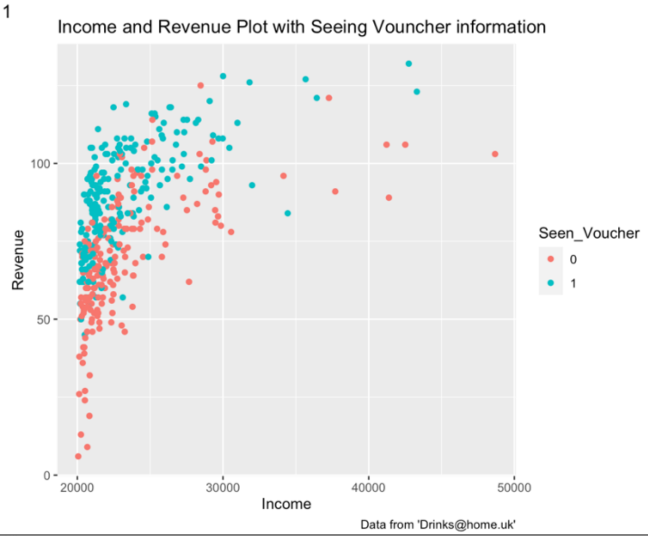
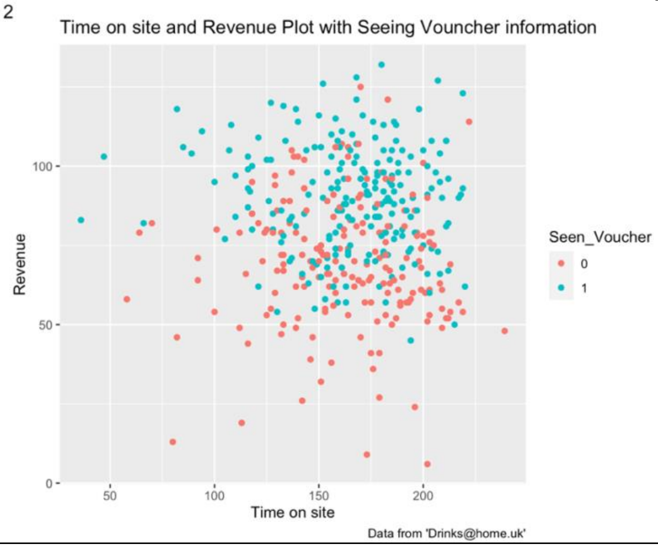
The assumption of liner regression	**The visualization of assumption**
Assumption 1: The expected value of the dependent variable has a linear relation with the explanatory variable(s).	Assumtion 1  Income and Revenue Plot with Seeing Vouncher information Data from 'Drinks@home.uk'
Assumption 2: The variance of the dependent variable should be the same for all values of the explanatory variable.	Assumtion 2  Time on site and Revenue Plot with Seeing Vouncher information Data from 'Drinks@home.uk'

Table13: The Checking Assumption Theory of Linear Regression Model

- **Modelling**

The fifth step involves applying a linear regression model to the data, with Revenue as the dependent variable and the other as the selected independent variable. The results from the model are provided in Table 14, which represents the coefficients of the linear regression model.

Factor	Coefficient Value	Standard Deviation Errors
Estimated Age	-0.015	0.089
Time on website per week	-0.022	0.022
Seen Voucher	19.696	1.415
Estimated Income	0.003	0.000
Advertisement Channel 2	6.828	2.017
Advertisement Channel 3	8.091	2.000
Advertisement Channel 4	12.974	2.000

Table 14: The Coefficient Value of Linear Regression Model

- **Data evaluation**

The final stage is evaluating the model and providing a clear summary of the model's outcome. Regarding model evaluation, it will focus on the R-Square and Adjusted R-Square. These two metrics indicate the proportion of variation in the dependent variable that can be explained by the model, with a value ranging from 0 to 1. From this model, the R Square and Adjusted R-Square is approximately 0.55, which means this model can cover 55% of the variation in Revenue value.

- **The model result**

The result from the model, overall, shows that the factors that positively impact the revenue are the vouchers seen, the estimated income, and the advertisement. On the other hand, the factors that negatively affect revenue are estimated age and time spent on websites by customers. A detailed analysis of each variable will be provided in the following paragraph.

Table 14 demonstrates that the estimated age factor coefficient is -0.015, suggesting revenue decreases by 0.015 GBP per year of additional years of age. The result showed that revenue decreased by 0.022 GBP per unit of increase in customer spending on-site time. According to the coefficient and standard deviation of the result of seeing vouchers, it increases revenue by 19.696 GBP per unit with a deviation of 1.415 GBP. The coefficient for estimated income is 0.003, meaning that revenue increases by 0.003 GBP per unit increase in income, which has little effect on the Revenue Factor. Advertisement Channel 1 is the reference for channels 1, 2, 3, and 4 advertising influence. The other channels (2,3,4) increase revenue by 6.828, 8.091, and 12.974 GBP, respectively, with a value gap that can increase or minus around 2.000 GBP.

The critical ideas derived from the model are the estimated weekly age time on the website and estimated income, which slightly impact the revenue value. In contrast, the voucher factor and the influence of advertisement channels significantly affect the revenue.
The result from the linear regression model could be used to analyse the three options to run this business's next campaign, as demonstrated in Table 15.

The option of next operating campaign	
1	Launch a marketing campaign aimed at customers aged 45 and above, because these people are more inclined to make larger purchases.
2	Offer a discount voucher for 20 GBP on customers' next purchase.
3	Invest additional funds in advertising by an influencer.

Table 15: The Option for Running the Following Project of 'drinks@home.uk' Business

Regarding option 1, the coefficient of linear regression of factor estimate age (as -0.015) minimally impacts revenue. Also, it provides a negative way that running the project is inappropriate because it might lead to a loss of profit.

In terms of option 2, this option requires investing 20 GBP in discount vouchers for customers. In comparison, the coefficient of linear regression of the voucher factor is 19.696 GBP, which is lower than the investment value of around 0.304, which might lead to a loss of profit and seem not to be the appropriate choice.

As regards option 3 in Table 15, this could be the best option to select due to the coefficient value in Table 16; the coefficient value of Advertisement channels leads to positive ways and also provides the significance associated with an increase in revenue. Therefore, this one might be the best option for me to select.

- Conclusion

To summarise, the factors that might positively affect revenue are the voucher seen, the estimated income, and the advertisement. By contrast, the factors that negatively impact revenue are the estimated age and time of customer spending on the website. Additionally, selecting one of the best options from the three choices to run the next marketing campaign to increase profit could be investing additional funds in advertising by an influencer.

The limitation of this analysis is that more than 400 data might be needed to make an accurate analysis, but it could improve the model's accuracy by providing more customer data. Additionally, the linear regression model is assumed by the data might fit the three assumptions of the linear regression model. Moreover, this model can cover 55% of the variation in revenue value, which is not the high score of the excellent model.

References

Palma, D., 2022. goalp: Weighted and Lexicographic Goal Programming Interface_. R package version 0.3.1. [online]. Available from: https://CRAN.R-project.org/package=goalp

Sánchez-Lozano, J.M., Teruel-Solano, J., Soto-Elvira, P.L., and García-Cascales, M.S., 2013. Geographical Information Systems (GIS) and Multi-Criteria Decision Making (MCDM) methods for the evaluation of solar farms locations: Case study in south-eastern Spain. Renewable and Sustainable Energy Reviews, **24**, pp.544-556.

Wickham, H., 2016. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York.

Wickham, H., François, R., Henry, L., Müller, K. and Vaughan, D., 2023. dplyr: A Grammar of Data Manipulation. R package version 1.1.3. [online]. Available from: https://CRAN.R-project.org/package=dplyr

Wickham, H., Hester, J., and Bryan, J., 2023. readr: Read Rectangular Text Data. R package version 2.1.4. [online]. Available from: https://CRAN.R-project.org/package=readr

Wright, K., 2021. corrgram: Plot a Correlogram. R package version 1.14. [online]. Available from: https://CRAN.R-project.org/package=corrgram

Appendix

Appendix 1: Processing TOPSIS analysis on Excel

Robot_Prototype	Carrying Capacity (max)	Battery Size (max)	Average Speed (max)	Cost Per Unit (min)	Reliability (max)
Weight	0.15	0.2	0.1	0.25	0.3
Archer	45	18	6	5210	22
Bowler	50	18	4	6250	24
Corner	60	12	4	4500	24
Deviant	40	24	10	7100	32
sum x square	9725	1368	168	136866600	2660
Root sum x square	98.61541462	36.98648402	12.9614814	11698.99996	51.57518783

normalisation

Robot_Prototype	Carrying Capacity (max)	Battery Size (max)	Average Speed (max)	Cost Per Unit (min)	Reliability (max)
Weight	0.15	0.2	0.1	0.25	0.3
Archer	0.456318114	0.486664263	0.46291005	0.44533721	0.426561704
Bowler	0.507020127	0.486664263	0.3086067	0.534233697	0.465340041
Corner	0.608424152	0.324442842	0.3086067	0.384648262	0.465340041
Deviant	0.405616101	0.648885685	0.77151675	0.60688948	0.620453387

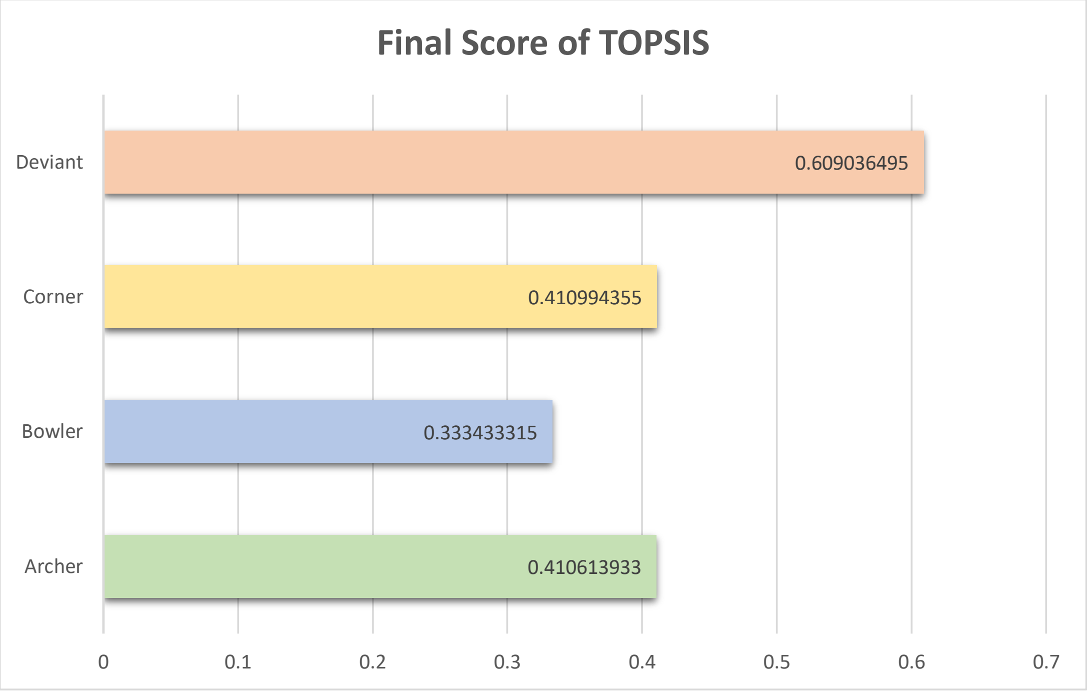
weighted normalised matrix

Robot_Prototype	Carrying Capacity (max)	Battery Size (max)	Average Speed (max)	Cost Per Unit (min)	Reliability (max)
Weight	0.15	0.2	0.1	0.25	0.3
Archer	0.068447717	0.097332853	0.046291005	0.111334302	0.127968511
Bowler	0.076053019	0.097332853	0.03086067	0.133558424	0.139602012
Corner	0.091263623	0.064888568	0.03086067	0.096162065	0.139602012
Deviant	0.060842415	0.129777137	0.077151675	0.15172237	0.186136016

PIS					
Robot_Prototype	Carrying Capacity (max)	Battery Size (max)	Average Speed (max)	Cost Per Unit (min)	Reliability (max)
Weight	0.15	0.2	0.1	0.25	0.3
Archer	0.000520566	0.001052632	0.000952381	0.000230197	0.003383459
Bowler	0.000231362	0.001052632	0.002142857	0.001398488	0.002165414
Corner	0	0.004210526	0.002142857	0	0.002165414
Deviant	0.00092545	0	0	0.003086947	0

NIS					
Robot_Prototype	Carrying Capacity (max)	Battery Size (max)	Average Speed (max)	Cost Per Unit (min)	Reliability (max)
Weight	0.15	0.2	0.1	0.25	0.3
Archer	5.78406E-05	0.001052632	0.000238095	0.001631196	0
Bowler	0.000231362	0.001052632	0	0.000329929	0.000135338
Corner	0.00092545	0	0	0.003086947	0.000135338
Deviant	0	0.004210526	0.002142857	0	0.003383459

Robot_Prototype	Si+	Si-	Final Score
Archer	0.078353261	0.054587209	0.410613933
Bowler	0.083610719	0.041824171	0.333433315
Corner	0.092297329	0.064402917	0.410994355
Deviant	0.063343487	0.098675438	0.609036495



Appendix 2: Processing Goal Programming analysis on R studio

```
install.packages('goalp')
library('goalp')

# Define goals and constraints
goals <- "MaxOrders: 9*Grocery + 6*Clothing + 4*Sport >= 95
        TechHours: 10*Grocery + 7*Clothing + 5*Sport <= 250
        Budget: 8700*Grocery + 8100*Clothing + 7700*Sport <=  250000
        Grocery >= 5
        Clothing >= 5
        Sport >= 5"

# Solve the problem
gp <- goalp(goals)

# Output the solution
print(gp)
summary(gp)
```

Appendix 3: Checking the assumption of linear regression in R studio

```
## Assumption 1: The expected value of the dependent variable has a
## linear relation with the explanatory variable(s).
library(dplyr)
library(ggplot2)
library(corrgram)

dataplot <- Transactions_Customer
dataplot$Seen_Voucher <- as.factor(dataplot$Seen_Voucher)

##Assumtion1
ggplot(data=dataplot) + geom_point(aes(Income,Revenue,color=Seen_Voucher))+
  labs(title = "Income and Revenue Plot with Seeing Vouncher information", caption = "Data from
'Drinks@home.uk'",
      tag = "Assumtion 1",x = "Income",y = "Revenue")

## Assumtion2
## The variance of the dependent variable should be the same for all values of the explanatory variable.

ggplot(data=dataplot) + geom_point(aes(Age,Revenue,color=Seen_Voucher))+
  labs(title = "Age and Revenue Plot with Seeing Vouncher information", caption = "Data from
'Drinks@home.uk'",
      tag = "Assumtion 2",x = "Age",y = "Revenue")

ggplot(data=dataplot) + geom_point(aes(Time_Site,Revenue,color=Seen_Voucher))+
  labs(title = "Time on site and Revenue Plot with Seeing Vouncher information", caption = "Data from
'Drinks@home.uk'",
      tag = "Assumtion 2",x = "Time on site",y = "Revenue")

##Assumtion3
mean_revenue <- mean(dataplot$Revenue, na.rm = TRUE) # Adjust for NA values
slope <- 0.5 # Hypothetical slope

ggplot(data=dataplot) +
  geom_point(aes(x = Income, y = Revenue, color=Seen_Voucher)) +
  geom_abline(intercept = mean_revenue, slope = slope, color="blue", linetype="dashed") +
  labs(title = "Income and Revenue Plot with Seeing Voucher information",
      caption = "Data from 'Drinks@home.uk'",
      tag = "Assumption 1",
      x = "Income",
      y = "Revenue")
```

Appendix 4: Processing the linear regression method in R studio

```
###TASK1and2
library(readr)
install.packages('corrgram')
library(dplyr)
library(ggplot2)
library(corrgram)

Transactions_Customer <- read_csv("Transactions_Customer.csv")

##cheching data quality
any_na_in_dataset <- any(is.na(Transactions_Customer))
print(any_na_in_dataset)

##handle with categories variable

Transactions_Customer$Advertisement_Channel_1 <- ifelse(Transactions_Customer$Advertisement_Channel
== 1, 1, 0)
Transactions_Customer$Advertisement_Channel_2 <- ifelse(Transactions_Customer$Advertisement_Channel
== 2, 1, 0)
Transactions_Customer$Advertisement_Channel_3 <- ifelse(Transactions_Customer$Advertisement_Channel
== 3, 1, 0)
Transactions_Customer$Advertisement_Channel_4 <- ifelse(Transactions_Customer$Advertisement_Channel
== 4, 1, 0)

Transactions_Customer$Advertisement_Channel <- NULL

## checking correlation and understanding the data

#change the name of col
names(Transactions_Customer) <- c("Age", "Time_Site", "Seen_Voucher", "Income", "Revenue",
"Ad_Channel_1", "Ad_Channel_2", "Ad_Channel_3", "Ad_Channel_4")
summary(Transactions_Customer)
cor(Transactions_Customer)
corrgram(Transactions_Customer)


##checking only revenue and time spending
cor(Transactions_Customer, Transactions_Customer$Revenue)
cor(Transactions_Customer, Transactions_Customer$Time_Site)
```

```
##fitting the model
# exclude Ad_Channel_1
model1 <- lm(Revenue ~ Age + Time_Site + Seen_Voucher + Income
           + Ad_Channel_2 + Ad_Channel_3 + Ad_Channel_4,
           data = Transactions_Customer)
# exclude Ad_Channel_2
model2 <- lm(Revenue ~ Age + Time_Site + Seen_Voucher + Income
           + Ad_Channel_1 + Ad_Channel_3 + Ad_Channel_4,
           data = Transactions_Customer)
# exclude Ad_Channel_3
model3 <- lm(Revenue ~ Age + Time_Site + Seen_Voucher + Income
           + Ad_Channel_1 + Ad_Channel_2 + Ad_Channel_4,
           data = Transactions_Customer)
# exclude Ad_Channel_4
model4 <- lm(Revenue ~ Age + Time_Site + Seen_Voucher + Income
           + Ad_Channel_1 + Ad_Channel_2 + Ad_Channel_3,
           data = Transactions_Customer)

summary(model1)
summary(model2)
summary(model3)
summary(model4)
```