Analysis for Turtle Games

Predict loyalty points, define customer segments and understand sentiment

Technical report

Prepared by Lilliana Golob

Contents

Background 3

Analytical approach 4

Data exploration 6

Insights and recommendations 15

Appendix 32

Background

Turtle Games is a global manufacturer and retailer of games, books and toys. They want to increase sales and seek insights and recommendations to create targeted and effective marketing campaigns and improve their business operations.

Turtle Games wants to understand the following:

- What contributes to customers accumulating loyalty points?
- How to predict loyalty point accumulation.
- How to segment and target customers.
- How can customer reviews inform decisions and improve operations?

Analytical approach

I used Python (Jupyter Notebook) and R (R Studio) for this project. In both IDEs, you'll see sections and comments which summarise the code, explain my intent and specify details such as method, column(s), and train/test ratios for example.

Python and R

Exploratory analysis: reviewed the data to understand customer demographics, and distribution of data and identify patterns and trends.

Linear regression: ran simple and multiple linear regression to understand which factors impact loyalty points and make predictions.

Python

Decision trees: applied DecisionTreeRegressor, pruning and feature importance to identify the most relevant factors to predict loyalty points.

Clustering: identified suitable customer groups for targeted marketing using k-means.

Sentiment analysis: used word frequency, TextBlob and Vader to understand customer feelings from online customer reviews.

*See **Appendix** for packages and libraries imported.*

Functions

I created functions in Python on repetitive tasks: review and validate data, preprocess data for sentiment analysis and simple linear regression.

*See **Appendix** for a summary list of functions.*

Data validation

I imported the CSV file and sense-checked the data to determine data types, unique values and descriptive statistics, and confirm there were no duplicates or missing values. Unnecessary columns were removed and columns were renamed for easier referencing.

For age, income, spend score and loyalty points I created bins grouping data into ranges.

I changed the Education values for 15-20-year-olds to Basic because this age group had degrees or higher.

education age_bin	Basic	PhD	diploma	graduate	postgraduate
15-20	0	30	0	70	20
20-30	0	70	0	230	90
30-40	20	190	70	330	140
40-50	30	70	50	150	60
50-60	0	30	40	70	40
60-70	0	50	30	50	40
70+	0	20	0	0	10

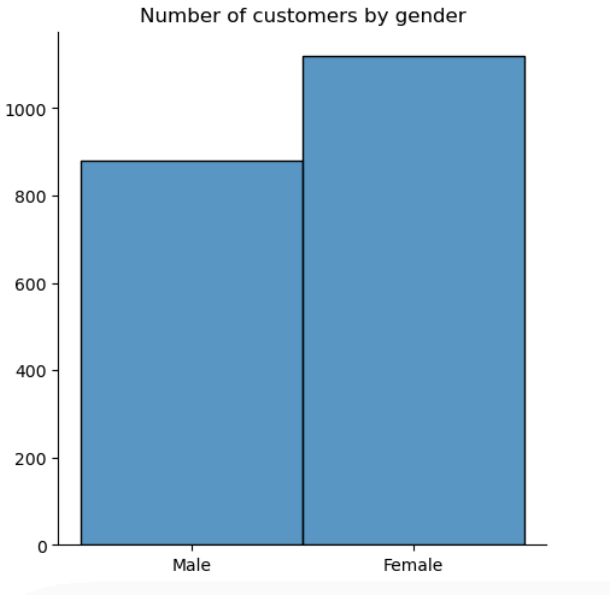
Data exploration

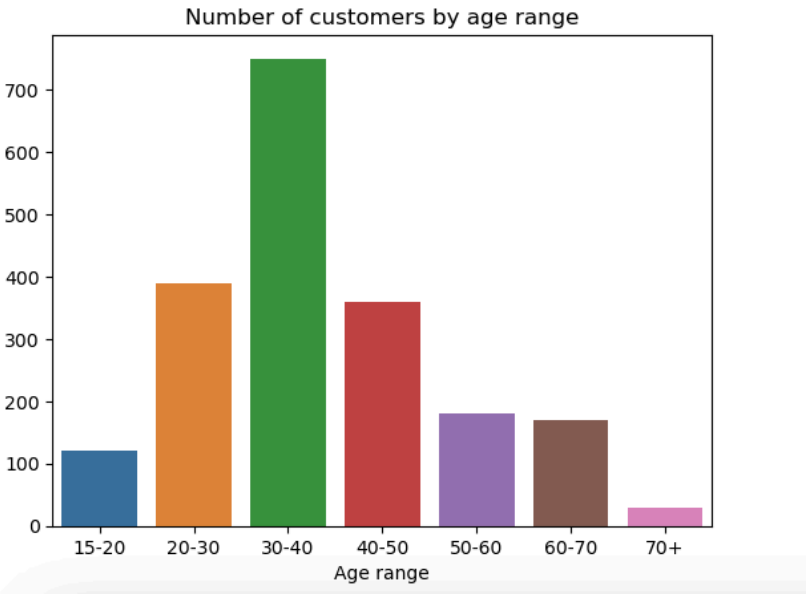
Demographics

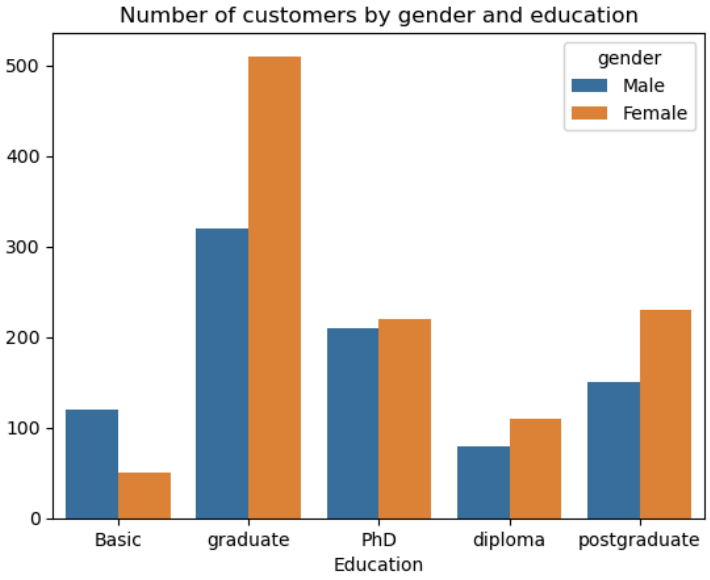
Customers are nearly equally split between males (880) and females (1120). 38% are aged between 30 and 40 years old. Most customers have an above basic education.

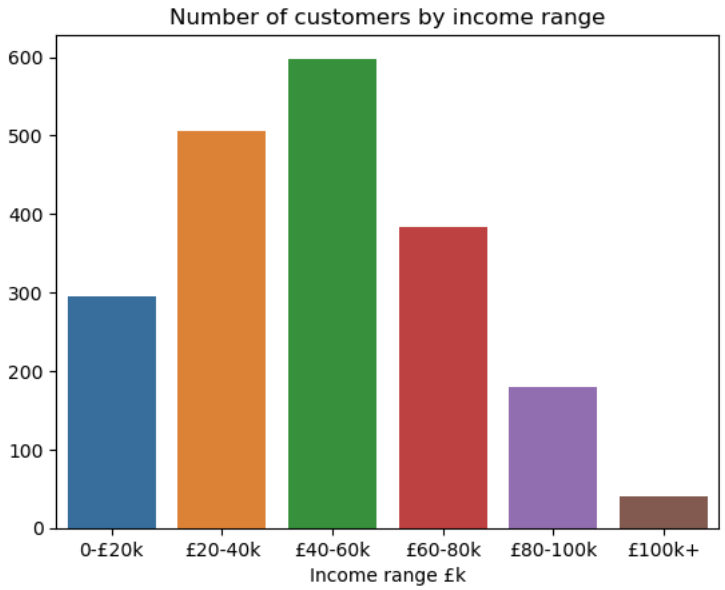
55% earn between £20k and £60k, and 36% have a spend score between 40 and 60.

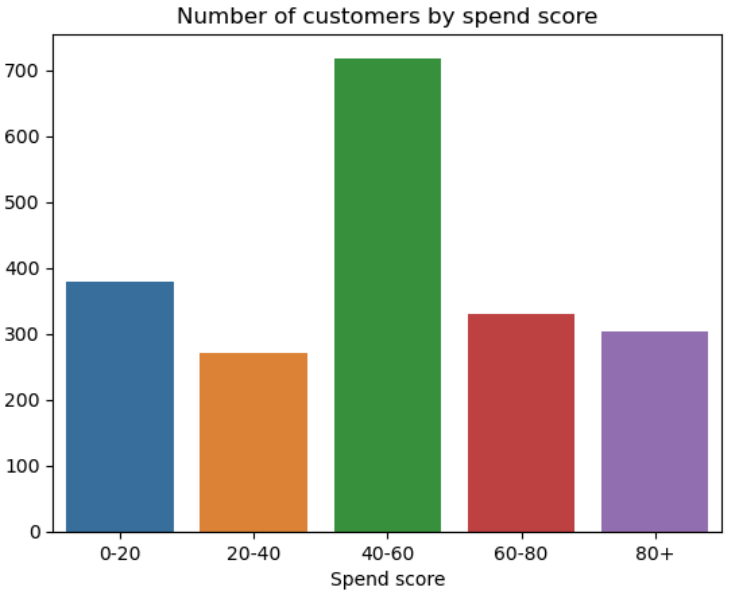
80% of customers have between 0 to 2000 loyalty points, with 46% having 1000 to 2000.

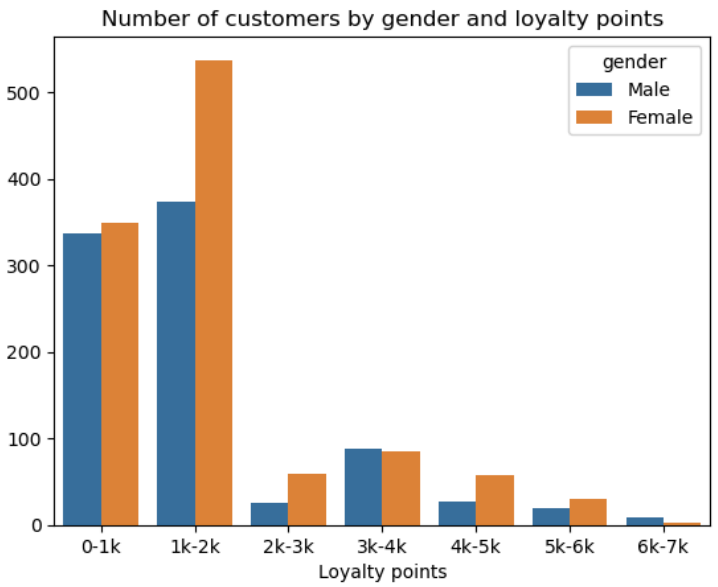



Number of customers by age range
700
600
500
400
300
200
100
0
15-20
20-30
30-40
40-50
50-60
60-70
70+
Age range


Number of customers by gender and education
gender
Male
Female
500
400
300
200
100
0
Basic
graduate
PhD
diploma
postgraduate
Education


Number of customers by income range
600
500
400
300
200
100
0
0-£20k
£20-40k
£40-60k
£60-80k
£80-100k
£100k+
Income range £k


Number of customers by spend score
700
600
500
400
300
200
100
0
0-20
20-40
40-60
60-80
80+
Spend score



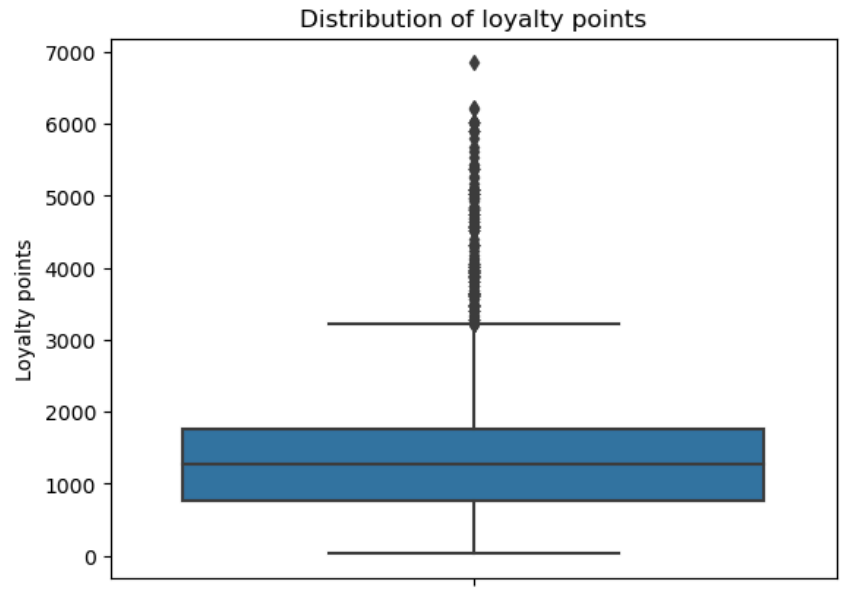
Loyalty points

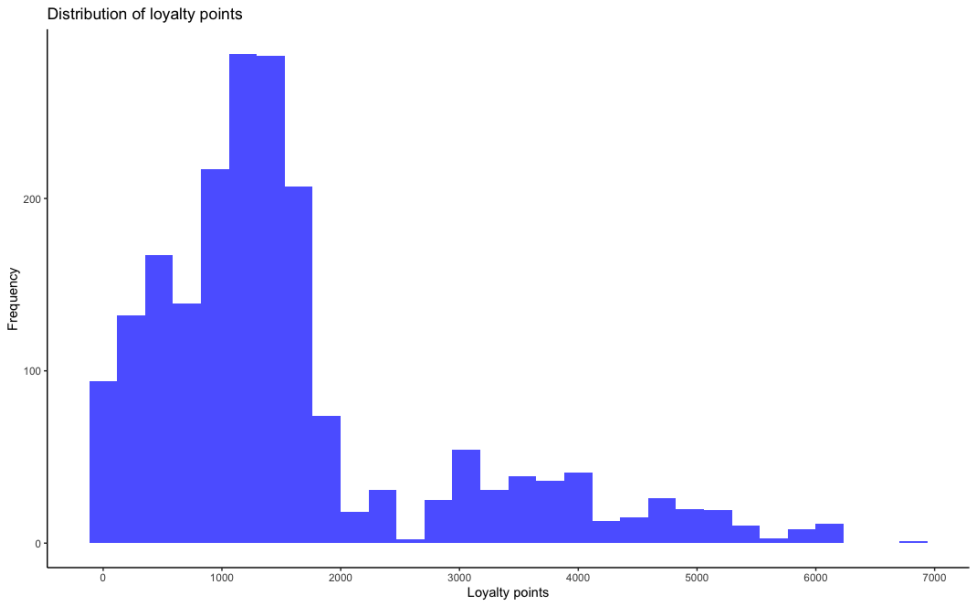
Range from 25 to 6847, with a mean of 1578 and a standard deviation of 1283.

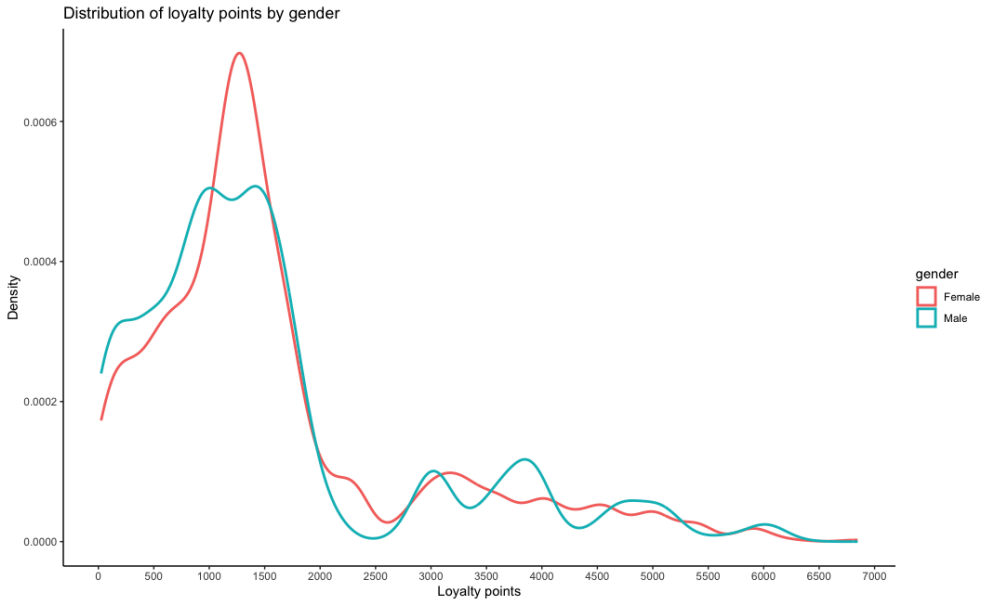
Data is skewed to the right and the higher values appear as outliers on the boxplot. I kept all data points because the data set is small. A high deviation and wide range may impact the performance of predictive models.

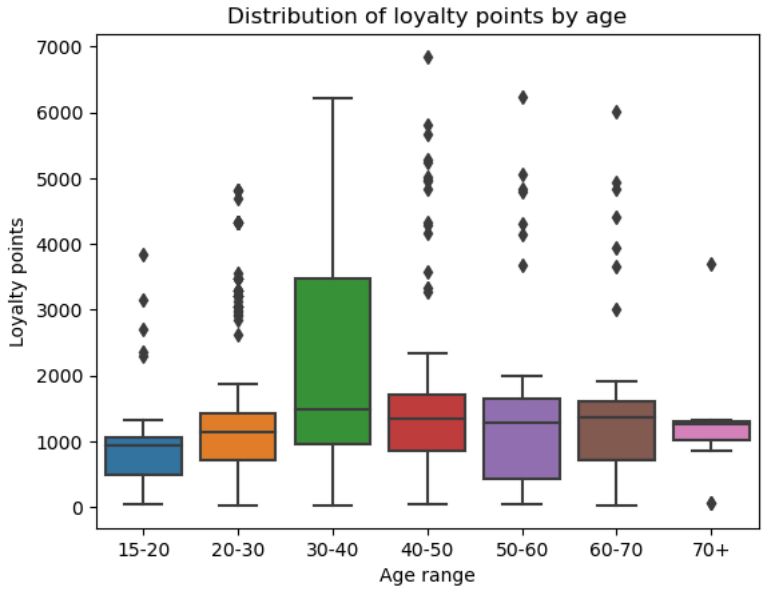
Customers aged 30-40 have an above-average number of loyalty points, and those aged between 30 to 70 have the highest maximum loyalty points, indicating age could be a factor.

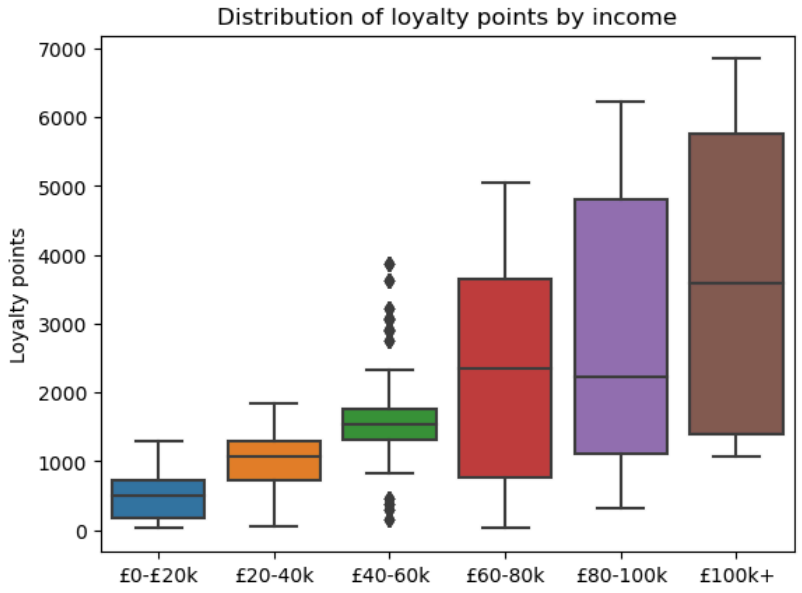
As expected, customers with low spend scores and income have fewer loyalty points.

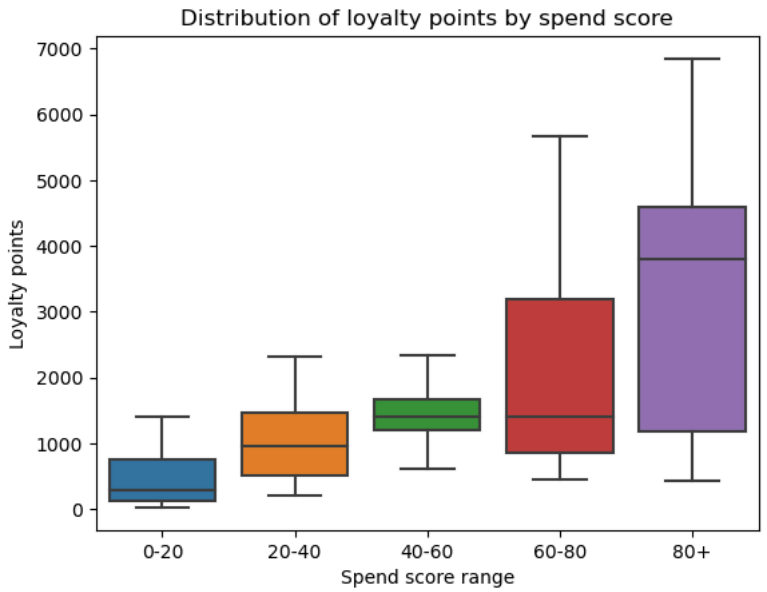

Distribution of loyalty points
Loyalty points
7000
6000
5000
4000
3000
2000
1000
0


Distribution of loyalty points
Frequency
200
100
0
0
1000
2000
3000
4000
5000
6000
7000
Loyalty points


Distribution of loyalty points by gender
Density
0.0006
0.0004
0.0002
0.0000
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000
5500
6000
6500
7000
Loyalty points
gender
Female
Male


Distribution of loyalty points by age
Loyalty points
7000
6000
5000
4000
3000
2000
1000
0
15-20
20-30
30-40
40-50
50-60
60-70
70+
Age range


Distribution of loyalty points by income
Loyalty points
7000
6000
5000
4000
3000
2000
1000
0
£0-£20k
£20-40k
£40-60k
£60-80k
£80-100k
£100k+


Distribution of loyalty points by spend score
Loyalty points
7000
6000
5000
4000
3000
2000
1000
0
0-20
20-40
40-60
60-80
80+
Spend score range

Average loyalty points (mean = 1578)

age_bin	15-20	20-30	30-40	40-50	50-60	60-70	70+
loyalty_points	889.783333	1264.666667	2133.878667	1378.344444	1177.516667	1246.517647	1186.533333

income_bin	0-£20k	£20-40k	£40-60k	£60-80k	£80-100k	£100k+
loyalty_points	509.99661	982.409901	1623.792642	2291.788512	2877.357542	3641.7

spend_score_bin	0-20	20-40	40-60	60-80	80+
loyalty_points	424.775132	1024.03321	1429.993036	2175.551515	3212.273927

Maximum loyalty points (max = 6847)

age_bin	15-20	20-30	30-40	40-50	50-60	60-70	70+
loyalty_points	3850	4814	6208	6847	6232	6020	3695

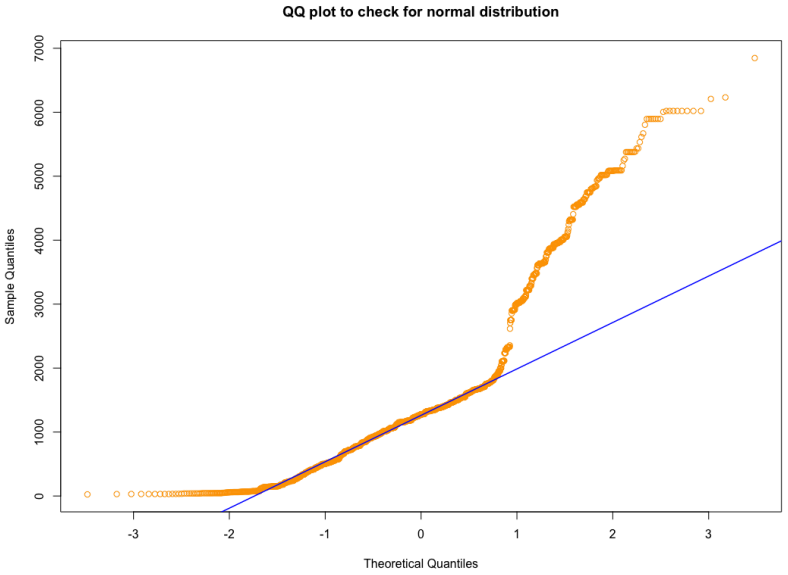
income_bin	£0-£20k	£20-40k	£40-60k	£60-80k	£80-100k	£100k+
loyalty_points	1292	1851	3866	5059	6232	6847

spend_score_bin	0-20	20-40	40-60	60-80	80+
loyalty_points	1414	2325	2332	5669	6847

Normal distribution

Loyalty points data is not normally distributed. Removing outliers, log transformation or adding data could correct this.

```
        Shapiro-Wilk normality test

data:  turtle$loyalty_points
W = 0.84307, p-value < 0.00000000000000022

>
> # Skewness
> skewness(turtle$loyalty_points)
[1] 1.463694
>
> # Kurtosis
> kurtosis(turtle$loyalty_points)
[1] 4.70883
```



Insights and recommendations

What contributes to customers accumulating loyalty points?

From the simple and multiple linear regressions and the decision tree regressor results, I concluded **income and spend score together are strong predictors of loyalty points**.

*See **Appendix** for simple linear regression results.*

Multiple linear regression

The initial multiple linear regression model using spend score and income together significantly improved results from individual simple regression.

The multiple regression model predicts 82.7% of the dependent variable. Looking at the coefficient, for every unit increase in spend score, loyalty points increase by 33. And for every £1k increase in income, loyalty points increase by 34.

OLS Regression Results

Dep. Variable:	y	**R-squared:**	0.827
Model:	OLS	**Adj. R-squared:**	0.827
Method:	Least Squares	**F-statistic:**	4770.
Date:	Fri, 28 Jun 2024	**Prob (F-statistic):**	0.00
Time:	20:39:40	**Log-Likelihood:**	-15398.
No. Observations:	2000	**AIC:**	3.080e+04
Df Residuals:	1997	**BIC:**	3.082e+04
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-1700.3051	35.740	-47.575	0.000	-1770.396	-1630.214
X[0]	32.8927	0.458	71.845	0.000	31.995	33.791
X[1]	33.9795	0.517	65.769	0.000	32.966	34.993

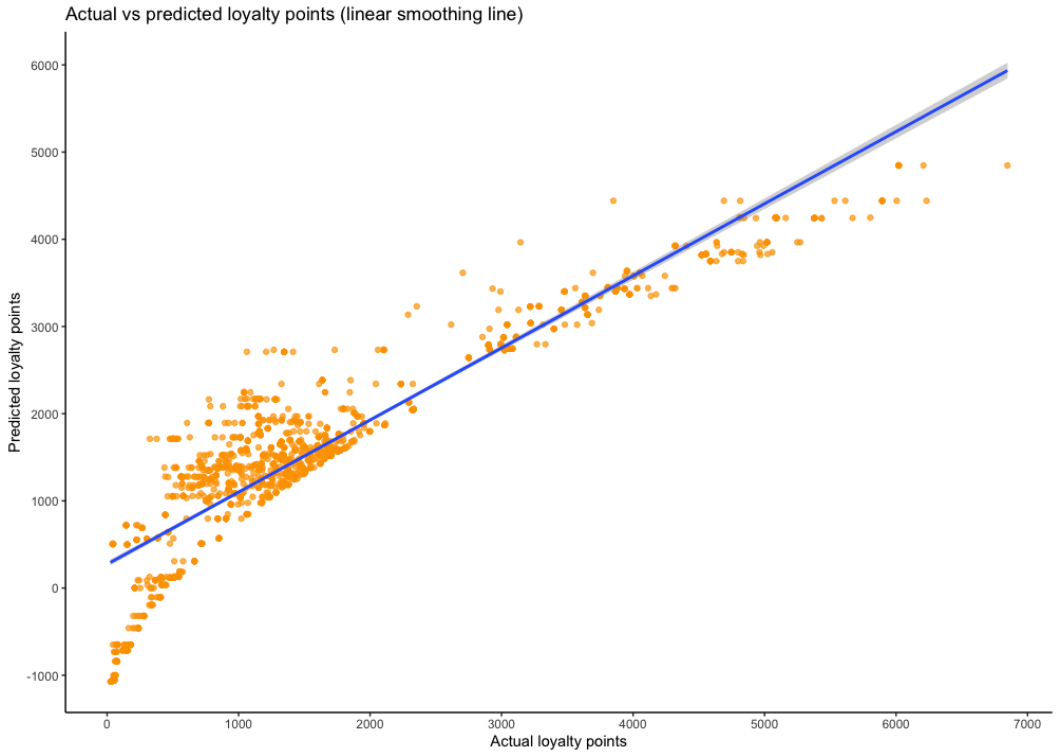
Omnibus:	4.723	**Durbin-Watson:**	3.477
Prob(Omnibus):	0.094	**Jarque-Bera (JB):**	4.650
Skew:	0.103	**Prob(JB):**	0.0978
Kurtosis:	3.115	**Cond. No.**	220.

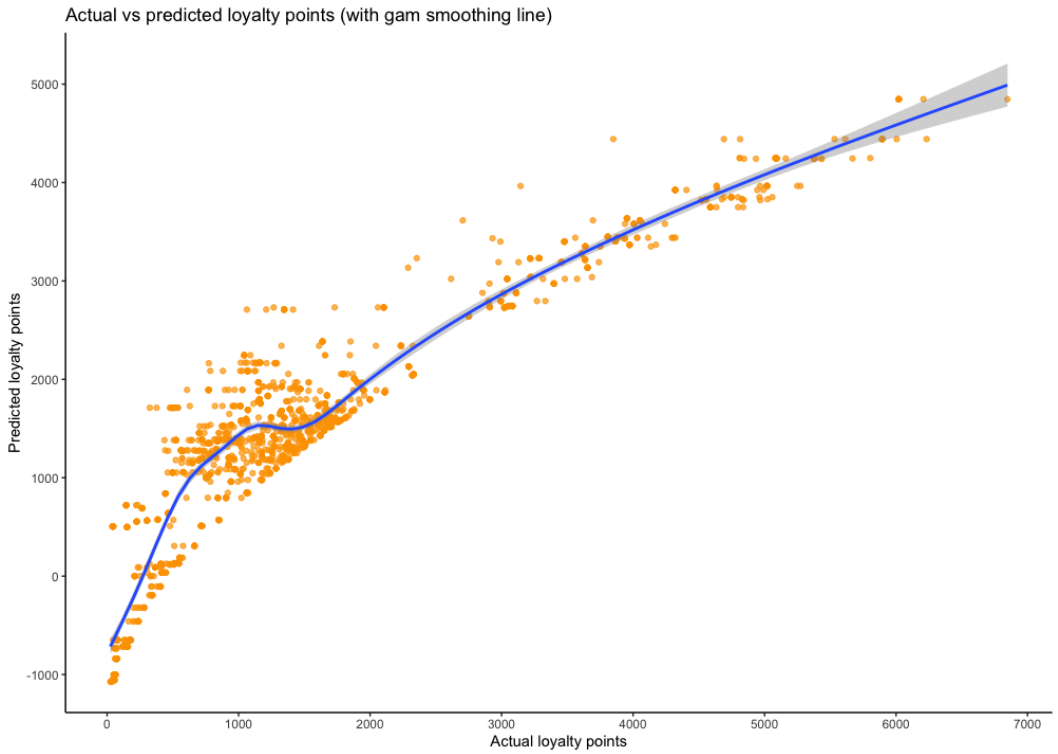
Model accuracy

Running the regression on the training subset yielded 424.31 mean absolute error, which indicates that the model's predictions are off by 424.31 from actual values. I anticipated a high error value due to a high standard deviation, and distribution of points on the boxplot.

However, these scatterplots show the model is not bad at predicting since many of the points are close-ish to the blue line.

*See **Appendix** for attempts to improve the model.*

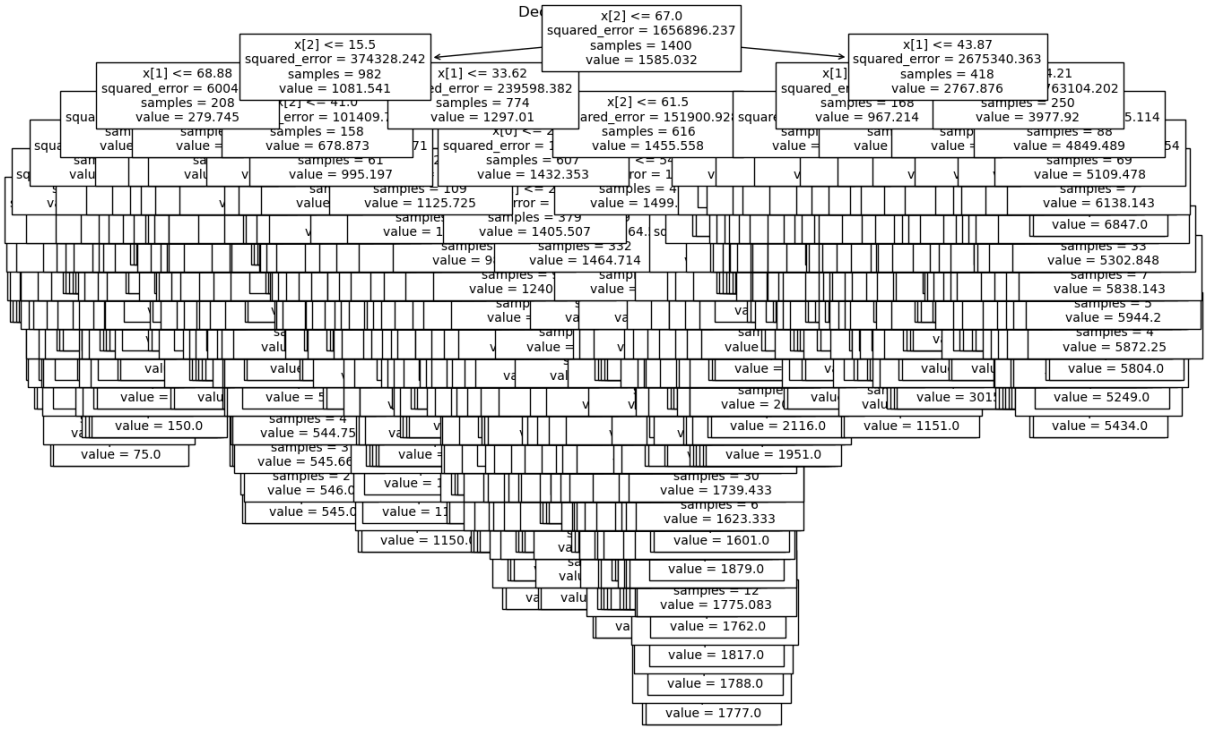




Decision tree regressor

Initial results indicate the model needs pruning due to overfitting because the R-squared for the training set is 1 (perfect predictions) and the tree is visually complicated.

```
Test data
Mean absolute error MAE:  32.12
Root Mean Squared Error RSME:  90.98
Difference:  58.86
R-squared:  0.99

Train data
Mean absolute error MAE:  0.0
Root Mean Squared Error RSME:  0.0
Difference:  0.0
R-squared:  1.0
```



Improving the model

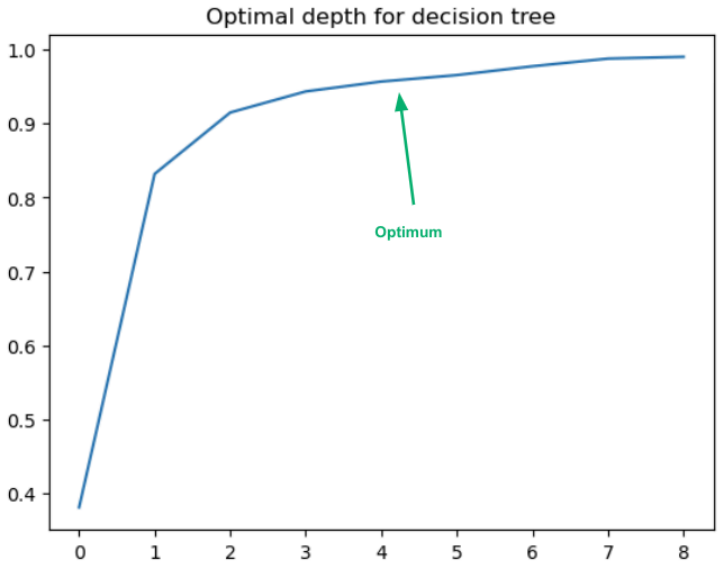
To identify optimal depth, I used GridSearchCV and plotted the accuracy score.

```
# Use GridSearchCV to find best parameter values

parameters = {'max_depth': list(range(1, 5))}

# Create new model
regressor_gs = DecisionTreeRegressor(random_state = 42)
regressor_gs = GridSearchCV(regressor_gs, parameters)

# Fit new model
regressor_gs.fit(X_train, y_train)
```

GridSearchCV
estimator: DecisionTreeRegressor
DecisionTreeRegressor

```
# Determine best parameters
regressor_gs.best_params_
```

```
{'max_depth': 4}
```



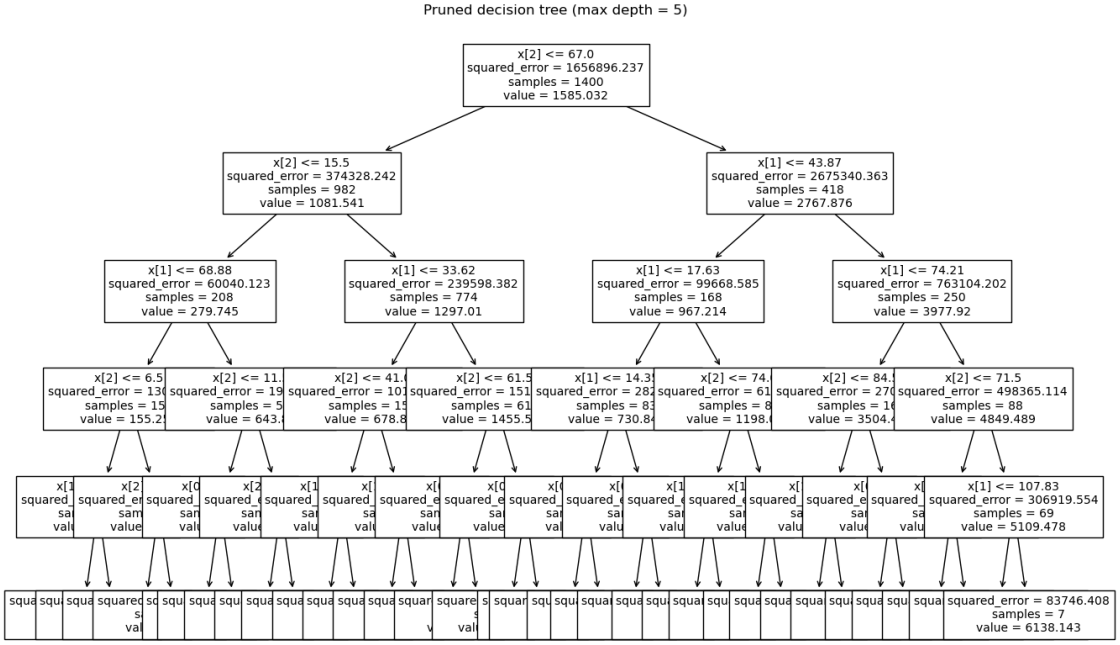
While both tests showed that a max depth of 4 is optimal, pruning to 5 gave the best results.

R-squared for test and train data is 0.96, and the difference between error values is lower.

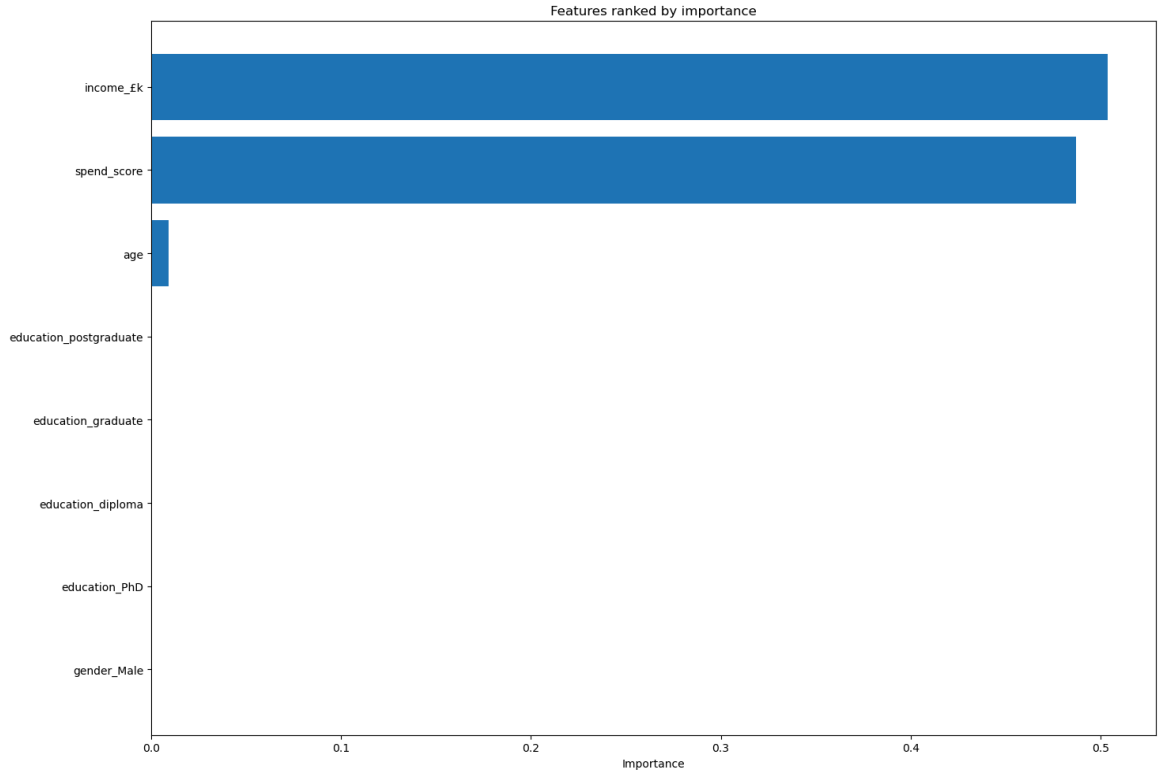
```
Max depth = 5

Test data
Mean absolute error MAE:  176.81
Root Mean Squared Error RSME:  264.8
Difference:  87.99
R-squared:  0.96

Train data
Mean absolute error MAE:  173.81
Root Mean Squared Error RSME:  253.33
Difference:  79.52
R-squared:  0.96
```

```
Max depth = 3

Test data
Mean absolute error MAE:  267.13
Root Mean Squared Error RSME:  371.43
Difference:  104.3
R-squared:  0.91

Train data
Mean absolute error MAE:  277.9
Root Mean Squared Error RSME:  385.31
Difference:  107.41
R-squared:  0.91
```

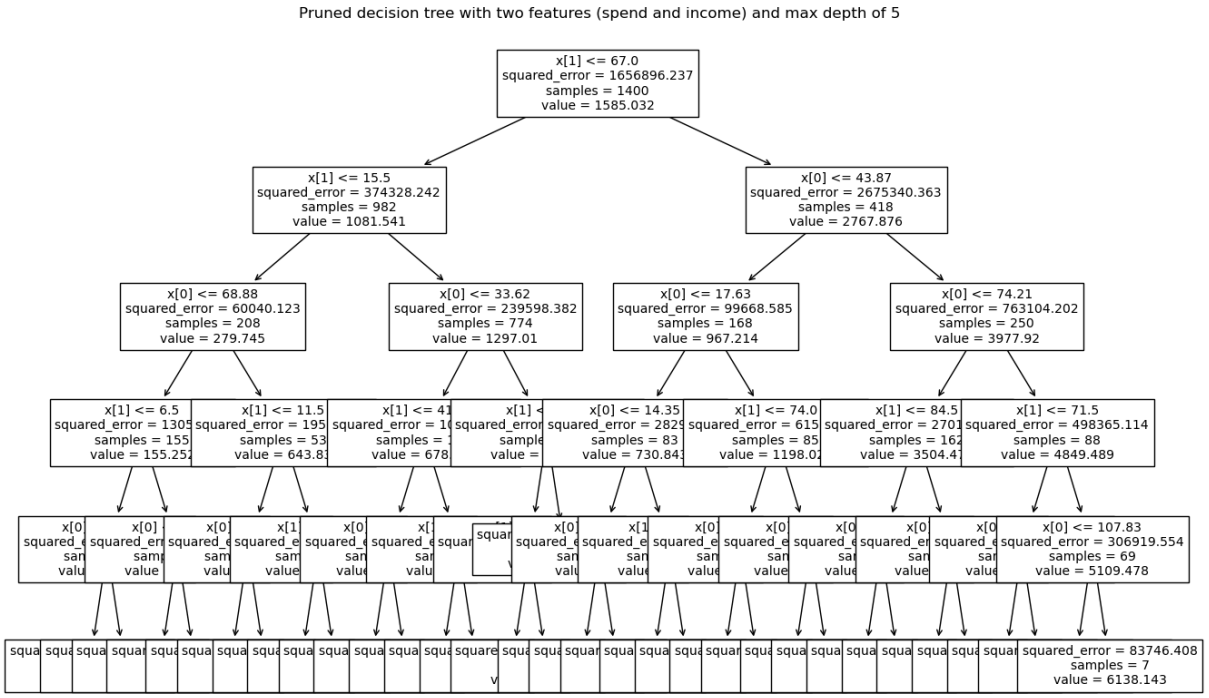
```
Max depth = 4

Test data
Mean absolute error MAE:  215.36
Root Mean Squared Error RSME:  302.93
Difference:  87.57
R-squared:  0.94

Train data
Mean absolute error MAE:  211.22
Root Mean Squared Error RSME:  298.87
Difference:  87.65
R-squared:  0.95
```



Feature importance confirmed income and spend score are the most important predictors. When I removed the other features, the model improved.




```
Results using only 2 features (spend score and income)

Test data
Mean absolute error MAE:  83.27
Root Mean Squared Error RSME:  161.55
Difference:  78.28
R-squared:  0.98

Train data
Mean absolute error MAE:  69.16
Root Mean Squared Error RSME:  135.85
Difference:  66.69
R-squared:  0.99
```



Recommendation

I would use multiple linear regression (spend score and income) to make predictions since I find the model easier to understand and the results are similar to the decision tree regressor. I would like to get additional data to help train the model since the data set is relatively small.

While income and spend score are good predictors of loyalty points, I would like to delve into demographics and products and do further analysis using classification decision trees to see how categorical variables (e.g. gender and product categories) affect loyalty points.

How to predict loyalty point accumulation?

I used a multiple linear regression model with spend-score and income as the predictors. As expected, the predictions show that **spend score greatly impacts loyalty points accumulation for the same income**.

For example: for an income of £50k (approx mean) and spend score of 25, predicted loyalty points are 821. If spend score is increased to 75, the predicted loyalty points are 2466.

This is even more evident with lower incomes. For example, when income is £30k and spend score is 25, the predicted loyalty points are 141. If spend score is 75, the predicted loyalty points are 1786.

```
# Customer 1: Income = 30, Spend score = 25
# Customer 2: Income = 30, Spend score = 50
# Customer 3: Income = 30, Spend score = 75
# Customer 4: Income = 30, Spend score = 100
# Customer 5: Income = 50, Spend score = 25
# Customer 6: Income = 50, Spend score = 50
# Customer 7: Income = 50, Spend score = 75
# Customer 8: Income = 50, Spend score = 100
# Customer 9: Income = 70, Spend score = 25
# Customer 10: Income = 70, Spend score = 50
# Customer 11: Income = 70, Spend score = 75
# Customer 12: Income = 70, Spend score = 100

data = data.frame(income = c(30, 30, 30, 30, 50, 50, 50, 50, 70, 70, 70, 70),
                  spend_score = c(25, 50, 75, 100, 25, 50, 75, 100,25, 50, 75, 100))
predictions <-predict(model, data)
round(predictions, digits = 0)
```

```
  1    2    3    4    5    6    7    8    9   10   11   12
141  964 1786 2608  821 1643 2466 3288 1501 2323 3145 3968
```

Recommendation

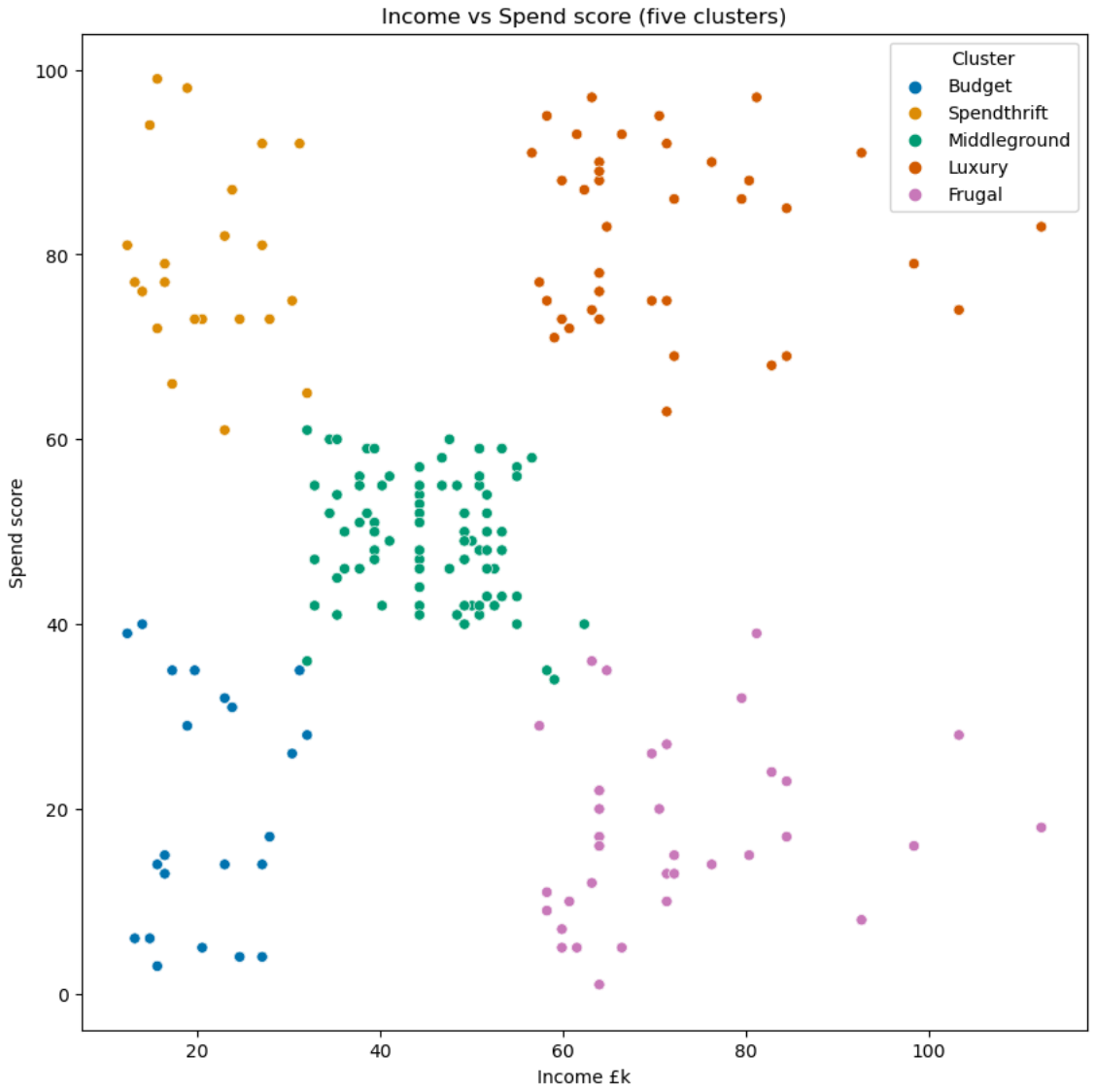
Focus marketing efforts on maximising customer spend scores regardless of of income. Communicate the value of loyalty points (e.g. the benefits) and how easy it can be to accumulate more.

How to segment and target customers?

I recommend Turtle Games **segment customers into five clusters**.

The scatterplot shows the position of the five clusters, and the summary table contains details about each segment including marketing opportunities.

*See **Appendix** for details on k-means clustering steps.*



Luxury	
Income Spend score Number of customers Customer profile	£60k - £100k+ 60 - 100 356 (2nd highest) High earner, high spender
Marketing opportunity: exclusive deals, premium products, premium loyalty benefits, exclusive members club	
Spendthrift	
Income Spend score Number of customers Customer profile	£10k - £40k 60 - 100 269 (least amount of customers) Low earner, high spender
Marketing opportunity: exclusive deals, early access to new products, loyalty discounts, payment plans	
Middleground	
Income Spend score Number of customers Customer profile	£40k - £60k 40 - 60 774 (most customers) Moderate income, moderate spender
Marketing opportunity: mid-range products, loyalty discount	
Frugal	
Income Spend score Number of customers Customer profile	£60k - £100k+ 10 - 40 330 (3rd highest) High earner, low spender
Marketing opportunity: discounts, promos demonstrating value for money, budget bundles	
Budget	
Income Spend score Number of customers Customer profile	£10k - £40k 10 - 40 271 (fourth) Low income, low spender
Marketing opportunity: discounts, end-of-line sales, payment plans, monthly subscriptions, budget bundles	

Recommendation

I recommend targeting customers with higher spend scores (Luxury and Spendthrift) because our predictive model indicates that higher spend scores significantly increase loyalty point accumulation. Focus communication on exclusivity and reward benefits, which may encourage them to spend more.

I would also test a campaign targeted at Frugal (high earner, low spender) because they have the financial means to spend. Communicating value for money and loyalty discounts may increase their spending. You could use the learnings from this campaign and apply them to Middleground segment.

I would also recommend doing further analysis into customers who have higher loyalty points to identify commonalities and see if age or gender impacts the clusters. I would use this info to develop customer personas because they help create personalised and appealing messages which will improve marketing results[1].

[1] The Complete, Actionable Guide to Marketing Personas

How can customer reviews inform decisions and improve operations?

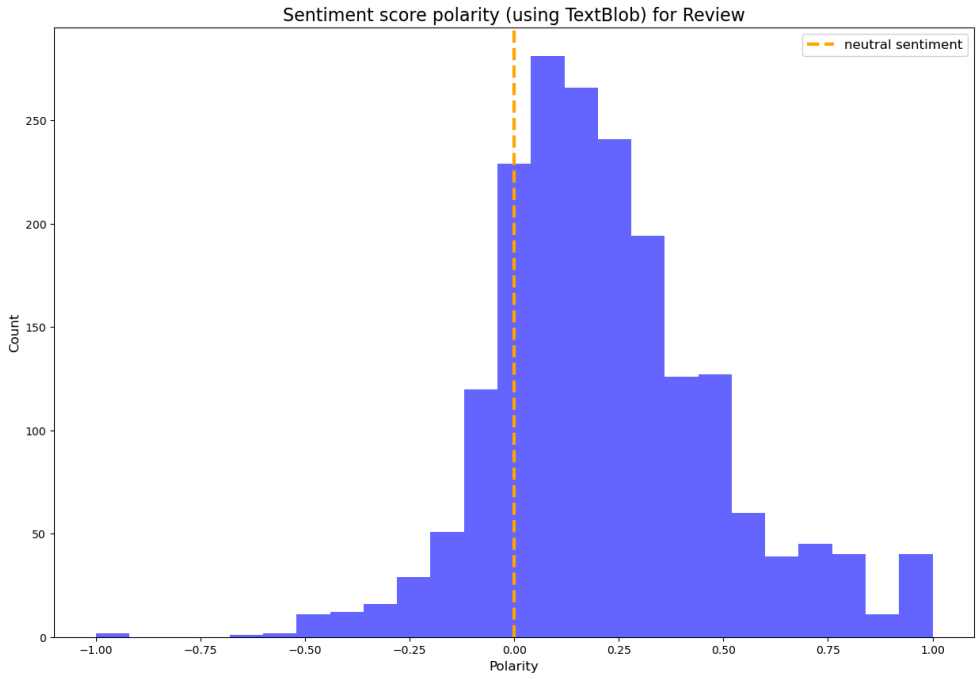
Looking at Review and Summary columns, **the overall sentiment is positive**. There are some errors in accuracy, however, I feel analysis Vader and TextBlob methods are useful for understanding how customers are feeling, and identifying issues and opportunities.

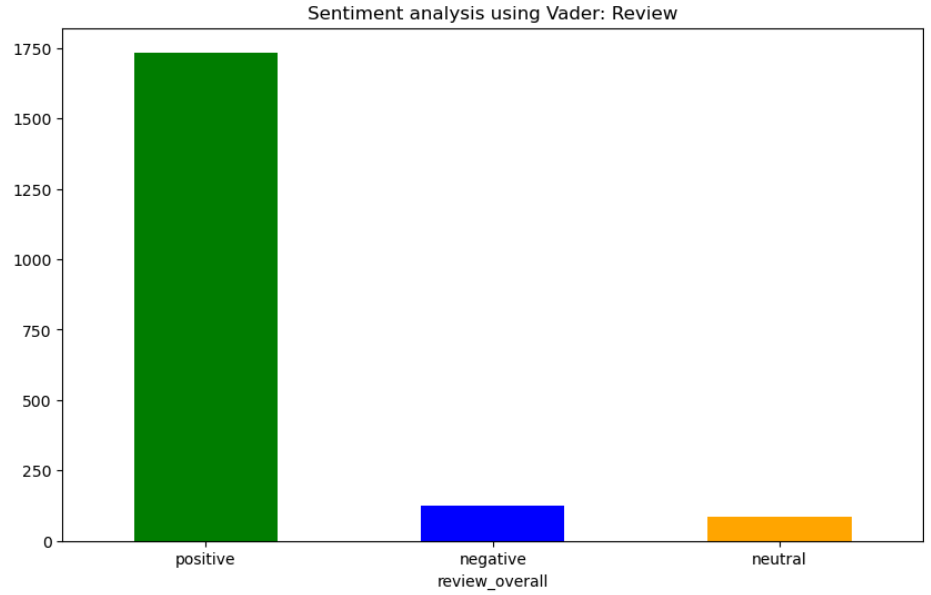
Recommendation

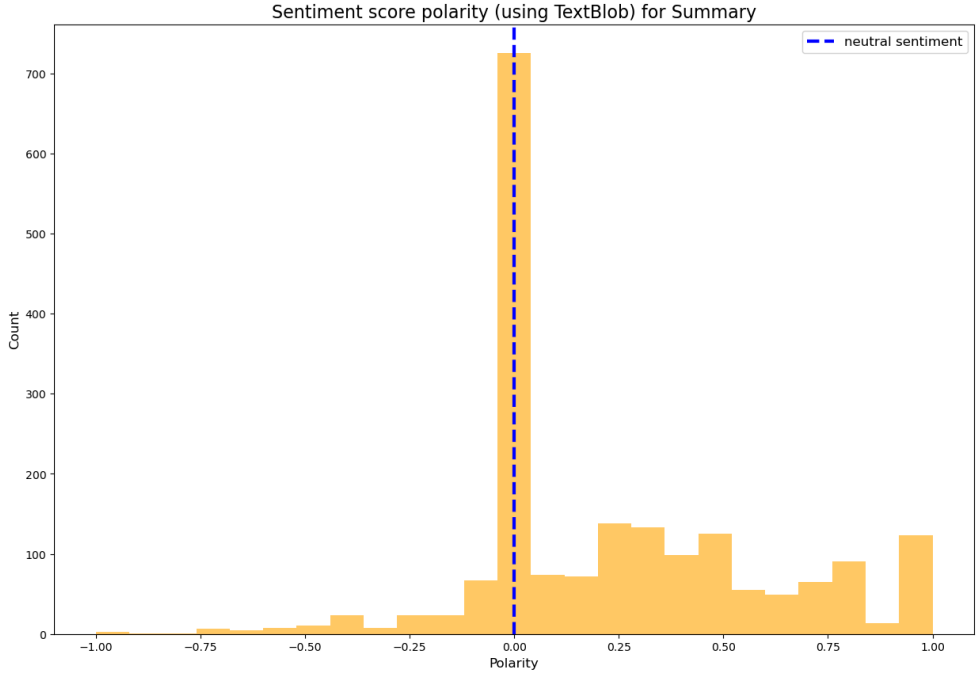
Since **negative reviews are the biggest risk to the business**, I recommend investigating them to determine the issue and any common themes so appropriate control measures can be put in place (e.g. discontinuing the product, contacting customers to resolve the issue).

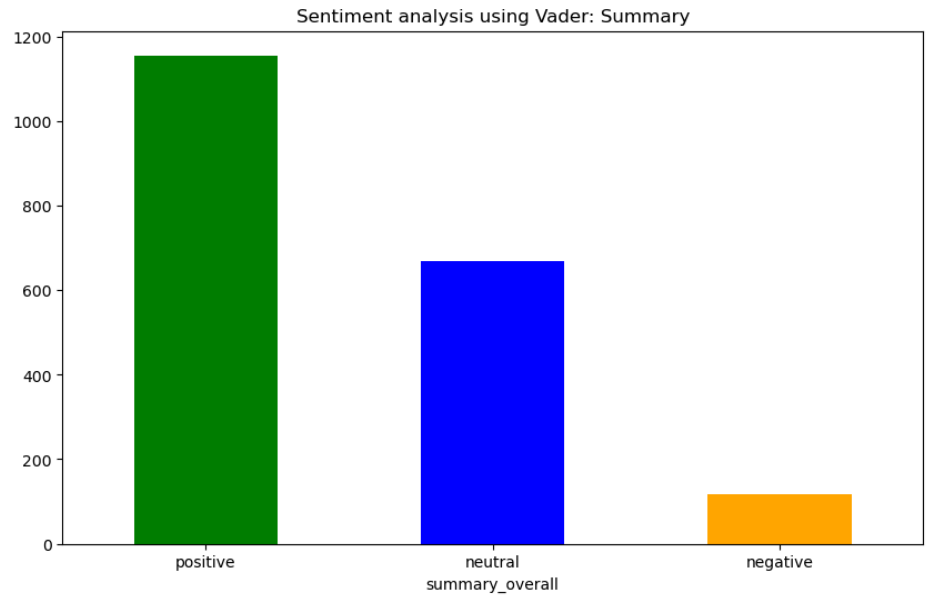
While this would involve manual input, Turtle Games will be able to identify areas for improvements and necessary corrective measures. Since the analysis has identified most of the sentiment is positive, it will be easier to filter through them.

Negative sentiment can also be turned into marketing opportunities. For example: communicate improvements and/or corrective measures to build trust, and promote alternative products that have high sentiment to unhappy customers.




Sentiment analysis using Vader: Review
1750
1500
1250
1000
750
500
250
0
positive
negative
neutral
review_overall


Sentiment score polarity (using TextBlob) for Summary
neutral sentiment
700
600
500
400
300
200
100
0
Count
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
Polarity



Review column: Most negative content based on polarity score (errors in green)

	review	polarity_review
206	BOOO UNLES YOU ARE PATIENT KNOW HOW TO MEASURE I DIDN'T HAVE THE PATIENCE NEITHER DID MY DAUGHTER. BORING UNLESS YOU ARE A CRAFT PERSON WHICH I AM NOT.	-1.000000
180	Incomplete kit! Very disappointing!	-0.975000
972	If you, like me, used to play D&D, but now you and your friends "growed up" and can't be together because all the responsibilities and bla bla bla... this game is for you! Come to the Dungeon!	-0.625000
1761	I'm sorry. I just find this product to be boring and, to be frank, juvenile.	-0.583333
360	One of my staff will be using this game soon, so I don't know how well it works as yet, but after looking at the cards, I believe it will be helpful in getting a conversation started regarding anger and what to do to control it.	-0.550000
115	I bought this as a Christmas gift for my grandson. Its a sticker book. So how can I go wrong with this gift.	-0.500000
225	this was a gift for my daughter. I found it difficult to use	-0.500000
228	I found the directions difficult	-0.500000
288	Instructions are complicated to follow	-0.500000
299	Difficult	-0.500000
790	This game is a blast!	-0.500000
1492	Expensive for what you get.	-0.500000
1786	Scrabble in a card game!	-0.500000
172	I sent this product to my granddaughter. The pom-pom maker comes in two parts and is supposed to snap together to create the pom-poms. However, both parts were the same making it unusable. If you can't make the pom-poms the kit is useless. Since this was sent as a gift, I do not have it to return. Very disappointed.	-0.491667
343	My 8 year-old granddaughter and I were very frustrated and discouraged attempting this craft. It is definitely not for a young child. I too had difficulty understanding the directions. We were very disappointed!	-0.452500
529	I purchased this on the recommendation of two therapists working with my adopted children. The children found it boring and put it down half way through.	-0.440741
304	Very hard complicated to make these.	-0.439583
419	Kids I work with like this game.	-0.400000
428	This game although it appears to be like Uno and have an easier play method it was still too time consuming and wordy for my children with learning disabilities.	-0.400000
488	My son loves playing this game. It was recommended by a counselor at school that works with him.	-0.400000
793	I bought this for my son. He loves this game.	-0.400000
809	Was a gift for my son. He loves the game.	-0.400000
1331	Excelent expansion, the game is nothing witout it.	-0.400000

Summary column: Most negative content based on polarity score

	summary	polarity_summary
21	The worst value I've ever seen	-1.000000
206	BORING UNLESS YOU ARE A CRAFT PERSON WHICH I AM ...	-1.000000
814	Boring	-1.000000
1143	before this I hated running any RPG campaign dealing with towns because it ...	-0.900000
1	Another worthless Dungeon Master's screen from GaleForce9	-0.800000
142	Disappointed	-0.750000
621	Disappointed.	-0.750000
780	Disappointed	-0.750000
1579	Disappointed	-0.750000
359	Promotes anger instead of teaching calming methods	-0.700000
870	Too bad, this is not what I was expecting.	-0.700000
875	Bad Quality-All made of paper	-0.700000
176	At age 31 I found these very difficult to make ...	-0.650000
99	Small and boring	-0.625000
495	It's UNO for the angry!	-0.625000
509	Mad dragon	-0.625000
1747	Ball of weird!	-0.625000
792	Disappointing	-0.600000
996	Disappointing.	-0.600000
1094	Disappointing	-0.600000
1761	Disappointing	-0.600000
74	Really small disappointed!	-0.593750
986	Then you will find this board game to be dumb and boring	-0.591667
360	Anger Control game	-0.550000
521	Anger Control Game	-0.550000
799	A Game of Luck and Strategy!	-0.500000
816	5 Star Game! 1 Star Version.	-0.500000

*See **Appendix** for details on word frequency and Word Clouds.*

Appendix

Python libraries and package

```
# Import libraries and packages

# Warnings
import warnings
warnings.filterwarnings('ignore')

# Data analysis
import numpy as np
import pandas as pd

# Visualisation
import matplotlib.pyplot as plt
import seaborn as sns

# Statistics and math
import scipy
from scipy import stats
import math

import statsmodels.api as sm
import statsmodels.stats.api as sms
from statsmodels.stats.outliers_influence import variance_inflation_factor
from statsmodels.formula.api import ols

# Regression
import sklearn
from sklearn.model_selection import train_test_split
from sklearn import linear_model
from sklearn import metrics
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error as mae

# Decision trees
from sklearn.tree import DecisionTreeRegressor, plot_tree
from sklearn.metrics import accuracy_score, confusion_matrix, ConfusionMatrixDisplay
from sklearn.metrics import classification_report
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import r2_score
```

```
# Clustering
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
from sklearn.metrics import accuracy_score
from scipy.spatial.distance import cdist

# Sentiment analysis
import nltk
import contractions
import os
from wordcloud import WordCloud
from nltk.tokenize import word_tokenize
from nltk import pos_tag
from nltk.probability import FreqDist
from nltk.corpus import stopwords
from nltk.corpus import wordnet as wn
from nltk.stem import WordNetLemmatizer, PorterStemmer
from textblob import TextBlob
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')
nltk.download('averaged_perceptron_tagger')

from collections import Counter
from collections import defaultdict
```

R libraries and packages

Imported ggokabeito colour package because it is a more accessible colour palette.

```r
# Import libraries
library(tidyverse)
library(skimr)
library(moments)
library(corrplot)
library(plotly)
library(GGally)

# Import colour package
library(ggokabeito)
```

Python functions

```
# Function to review and validate the data

def review_data(df):
    """
    Review the data in a dataframe returning:
    datatypes, missing values, descriptive statistics,
    duplicates and unique values.
    """
```

```
# Function to preprocess data

def preprocess(df, column, col_clean):
    """
    Preprocessing data so can perform sentiment analysis and,
    storing the cleaned up data in a new column (col_clean).
    """
```

```
# Function to fit linear model using polyfit and create scatterplot for simple linear regression

def slr_correlation_plot(df, dependent, independent):
    """
    Print scatterplot with a trendline and correlation calculation
    for simple linear regresssion using Numpy polyfit model
    with dependent/result variable (y co-ordinate) and
    one independent/predictor variable (x co-ordinate).
    """
```

```
# Function to create OLS regression summary table

def ols_summary(df, dependent, independent):
    """
    Print regression summary table using OLS (ordinary least squares) method,
    and predicted values,
    first adding a constant to the model.
    """
```

```
# Function to create residual plot

def residual_plot(df, dependent, independent):
    """
    Print residual plot which shows differences between
    observed values and predicted values.
    """
```

```
# Function to create simple linear regression and create plot with regression line

def slr(df, dependent, independent):
    """
    Create simple linear regression model,
    split data into test and train (50/50 split)
    and output scatterplot on training data with regression line.
    """
```

```
# Function to predict values for simple linear regression and determine accuracy, coefficient and intercept

def slr_predict(df, dependent, independent):
    """
    Create simple linear regression model and predict values,
    and print r-squared, intercept and coefficient values.
    """
```

Simple linear regression results

Both income and spend score are strongly correlated to loyalty points (over 60%). Age and product have little to no correlation.

	age	income	spend_score	loyalty_points	product
age	1.00	-0.01	-0.22	-0.04	0.00
income	-0.01	1.00	0.01	0.62	0.31
spend_score	-0.22	0.01	1.00	0.67	0.00
loyalty_points	-0.04	0.62	0.67	1.00	0.18
product	0.00	0.31	0.00	0.18	1.00

Income

Income on its own is significant (p-value is 0) and a moderate predictor of loyalty points based on 0.38 R-squared value.

The relationship between loyalty points and income is positive. This correlates to the income effect, which predicts that as a person's income grows, they will begin to demand more[2].

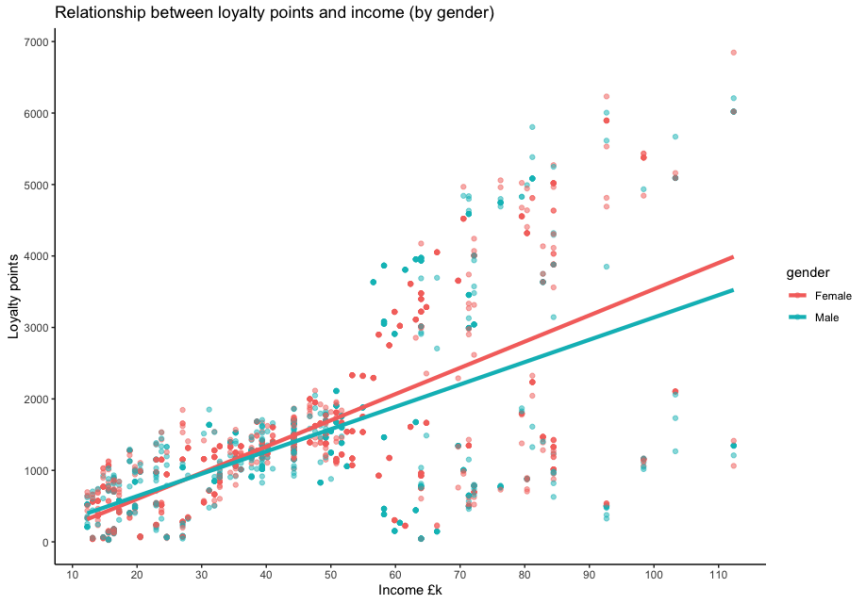
[2] What Is the Income Effect? Its Meaning and Example

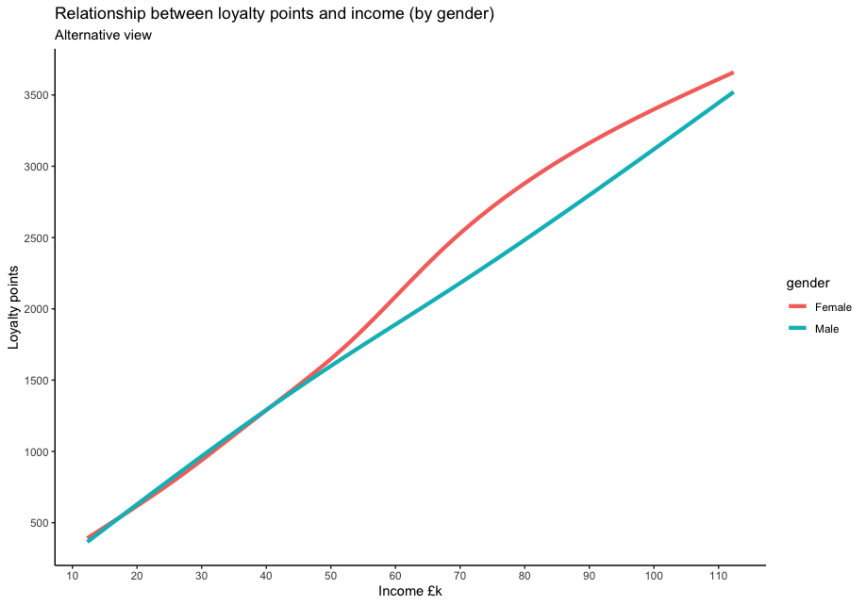
INDEPENDENT VARIABLE: income_£k — OLS Regression Results

Dep. Variable:	loyalty_points	R-squared:	0.380
Model:	OLS	Adj. R-squared:	0.379
Method:	Least Squares	F-statistic:	1222.
Date:	Fri, 28 Jun 2024	Prob (F-statistic):	2.43e-209
Time:	20:39:40	Log-Likelihood:	-16674.
No. Observations:	2000	AIC:	3.335e+04
Df Residuals:	1998	BIC:	3.336e+04
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-65.6865	52.171	-1.259	0.208	-168.001	36.628
income_£k	34.1878	0.978	34.960	0.000	32.270	36.106

Omnibus:	21.285	Durbin-Watson:	3.622
Prob(Omnibus):	0.000	Jarque-Bera (JB):	31.715
Skew:	0.089	Prob(JB):	1.30e-07
Kurtosis:	3.590	Cond. No.	123.

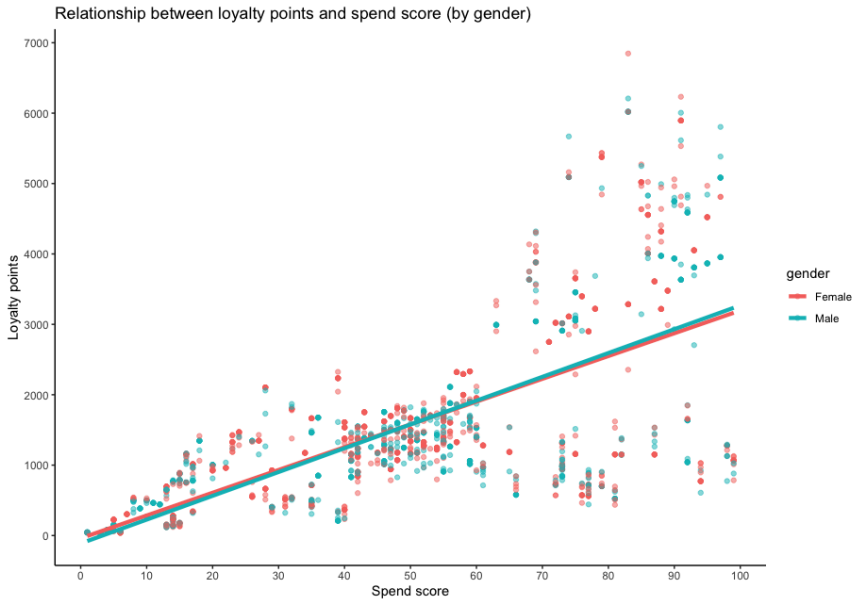


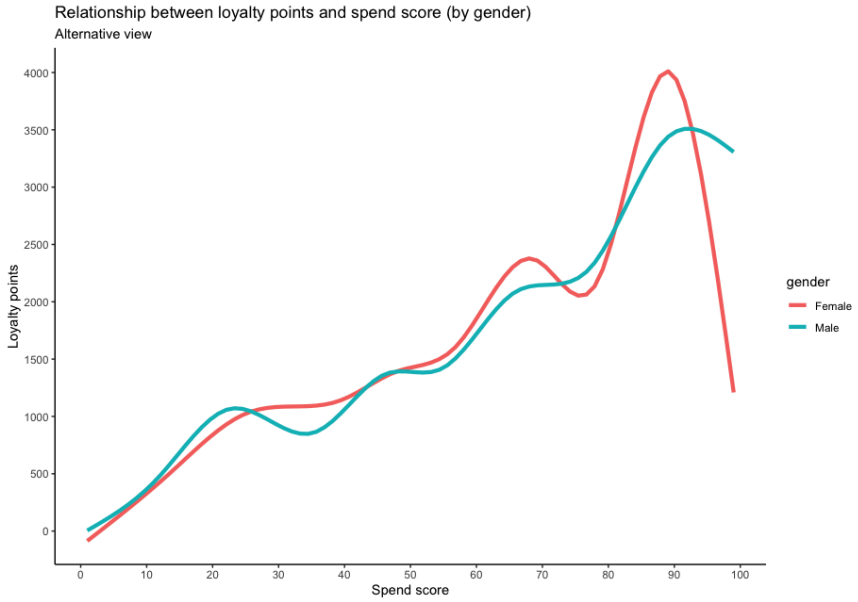


Spend score

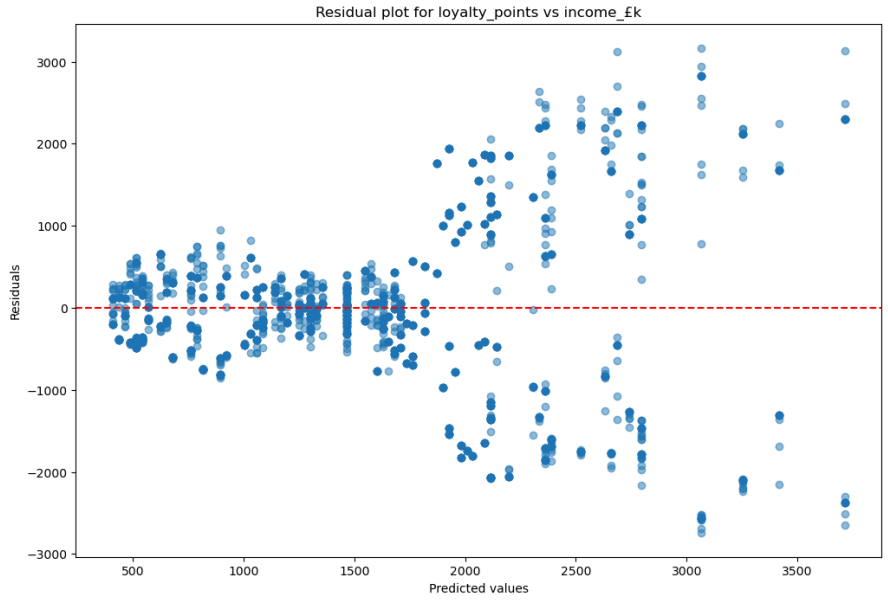
On its own, spend score is significant (p-value is 0) and a moderate predictor of loyalty points based (R-squared 0.452).

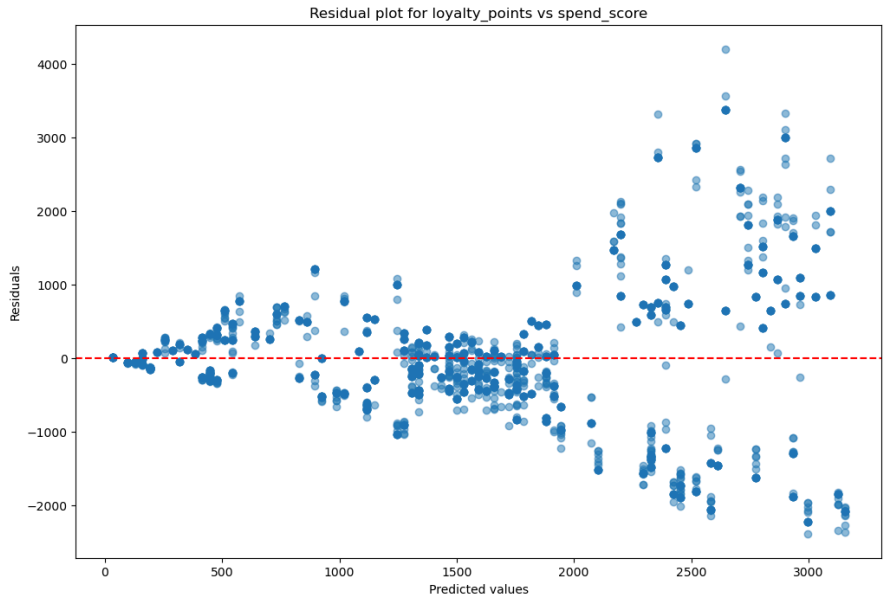
```
INDEPENDENT VARIABLE: spend_score                 OLS Regression Results
==============================================================================
Dep. Variable:         loyalty_points   R-squared:                       0.452
Model:                            OLS   Adj. R-squared:                  0.452
Method:                 Least Squares   F-statistic:                     1648.
Date:                Fri, 28 Jun 2024   Prob (F-statistic):          2.92e-263
Time:                        20:39:40   Log-Likelihood:                -16550.
No. Observations:                2000   AIC:                         3.310e+04
Df Residuals:                    1998   BIC:                         3.312e+04
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const        -75.0527     45.931     -1.634      0.102    -165.129      15.024
spend_score   33.0617      0.814     40.595      0.000      31.464      34.659
==============================================================================
Omnibus:                      126.554   Durbin-Watson:                   1.191
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              260.528
Skew:                           0.422   Prob(JB):                     2.67e-57
Kurtosis:                       4.554   Cond. No.                         122.
==============================================================================
```





The residual plots show that both income and spend score individually are ok at predicting loyalty points when values are lower, however, they display heteroscedasticity as loyalty points increase.





Age

Contrary to the exploratory analysis, age is **not** a predictor of loyalty points. R-squared is nearly zero and p-value is greater than 0.05, indicating it is not significant. The flat line on the plot confirms there is no relationship.

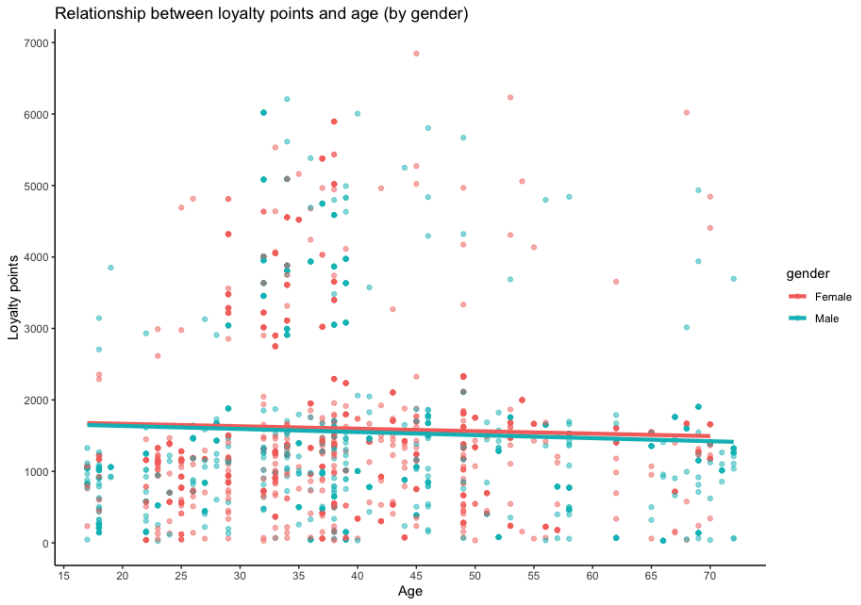
This could be because gaming is not just for young people: 56% of UK adults play games on or offline, and 23% of 55-64-year-olds play games online[3].

INDEPENDENT VARIABLE: age — OLS Regression Results

Dep. Variable:	loyalty_points	R-squared:	0.002
Model:	OLS	Adj. R-squared:	0.001
Method:	Least Squares	F-statistic:	3.606
Date:	Fri, 05 Jul 2024	Prob (F-statistic):	0.0577
Time:	18:09:44	Log-Likelihood:	-17150.
No. Observations:	2000	AIC:	3.430e+04
Df Residuals:	1998	BIC:	3.431e+04
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	1736.5177	88.249	19.678	0.000	1563.449	1909.587
age	-4.0128	2.113	-1.899	0.058	-8.157	0.131

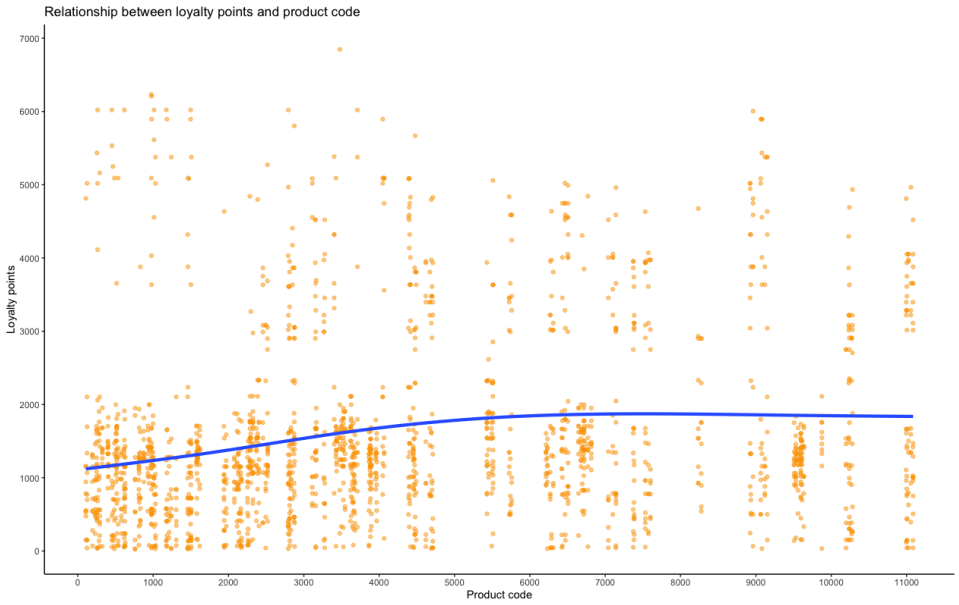
Omnibus:	481.477	Durbin-Watson:	2.277
Prob(Omnibus):	0.000	Jarque-Bera (JB):	937.734
Skew:	1.449	Prob(JB):	2.36e-204
Kurtosis:	4.688	Cond. No.	129.



[3] Ofcom Online Nation Report 2023

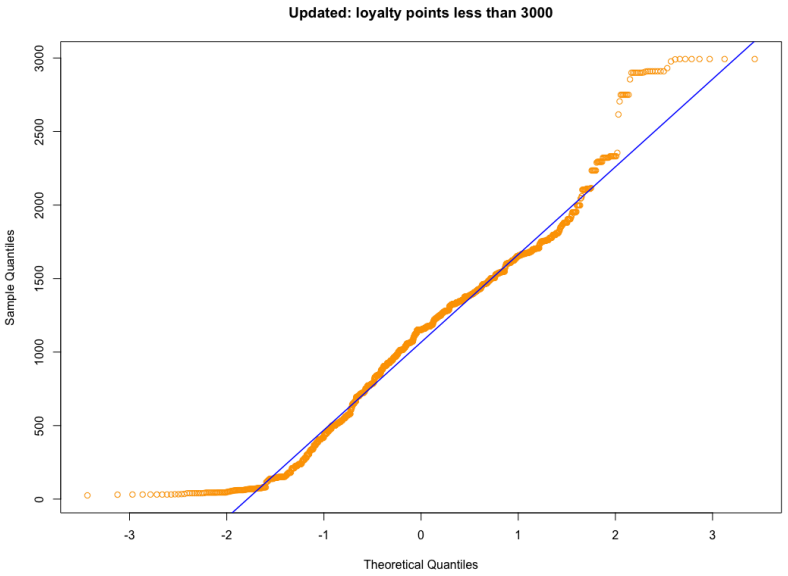
Product

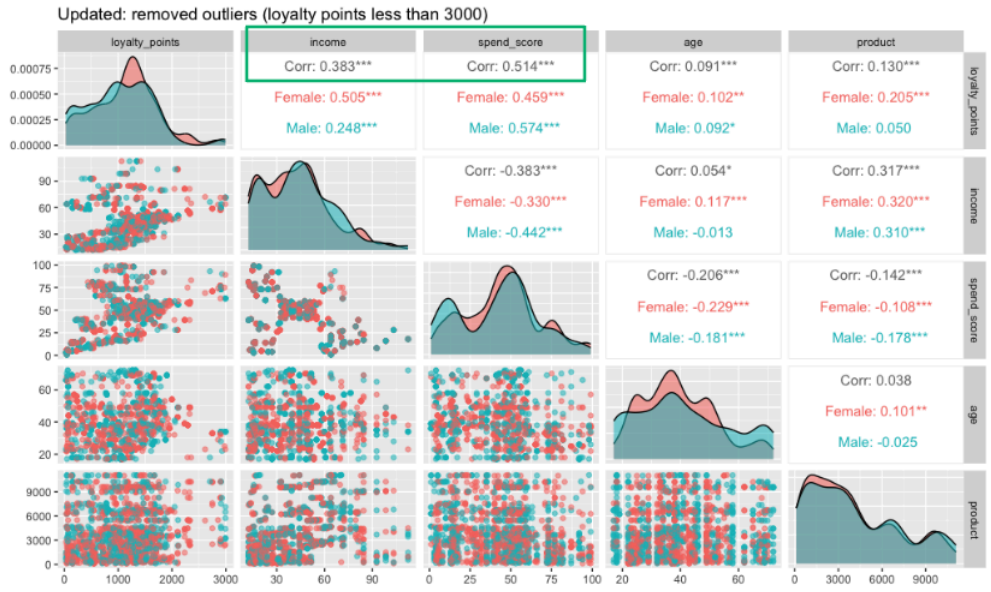
The relationship between product and loyalty points is nearly flat even though a linear line of best fit was not specified. This indicates that there is no correlation, which may not be true. This data is 200 product codes without any context of product categories.



Improving the multiple regression model

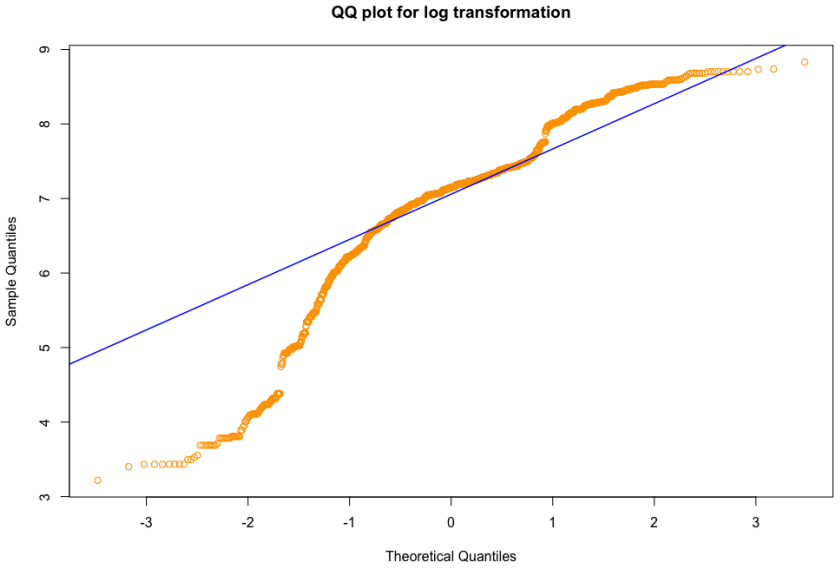
Removing 'outliers' resulted in a more normal distribution, however, the correlation for the two strongly related variables (spend score and income) was reduced.

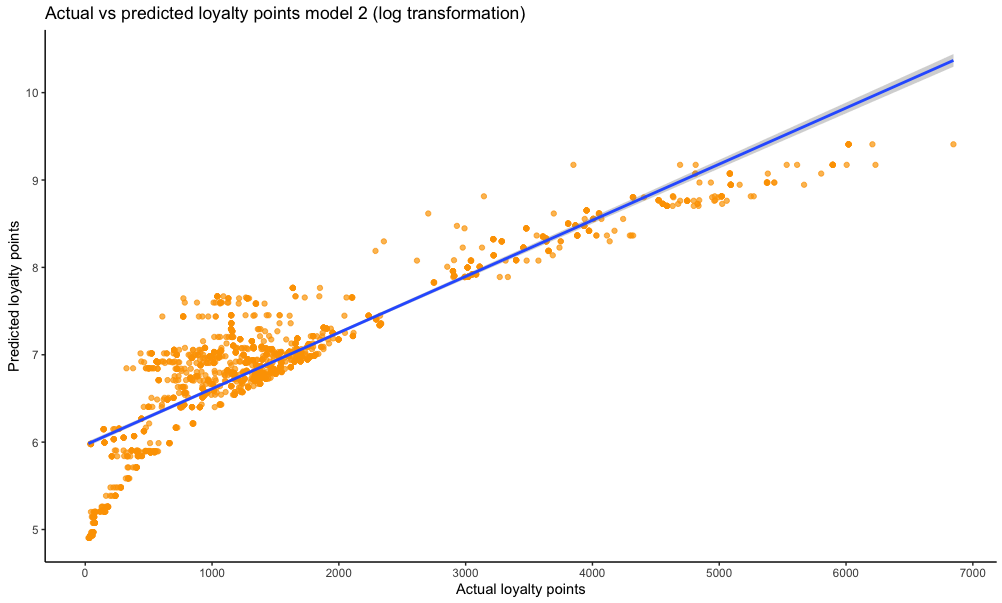




Log transformation gave mixed improvement. The variables are still significant, however, the adjusted R-square reduced from 82.67 to 79.86. The only improvement was the residual error significantly reduced from 534.1 to 0.4579.

When comparing the scatterplots with the original, there is not much difference. Since the original model has a higher R-squared, I used it to make my predictions.





Log transformation model

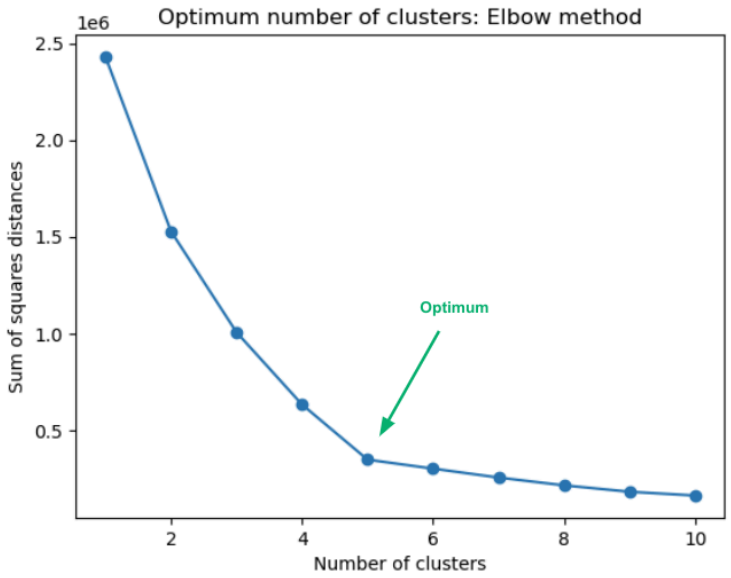
```
Residual standard error: 0.4579 on 1997 degrees of freedom
Multiple R-squared:  0.7989,    Adjusted R-squared:  0.7986
F-statistic:  3965 on 2 and 1997 DF,  p-value: < 0.00000000000000022
```

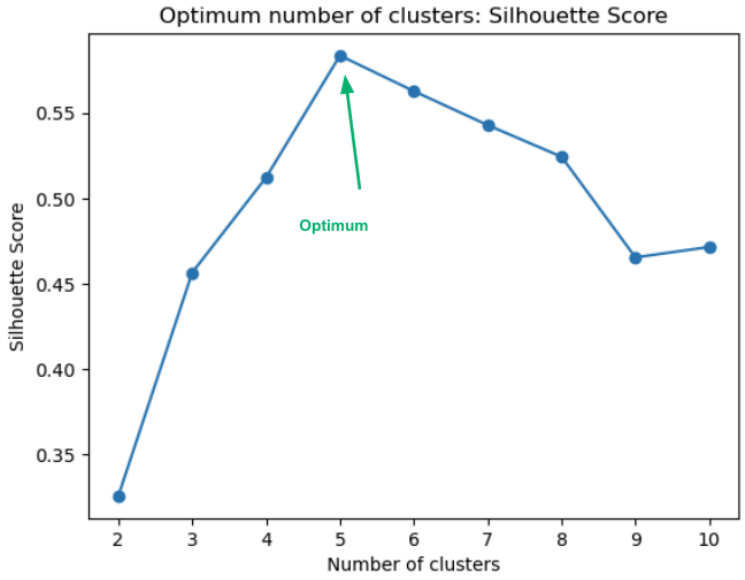
Original model

```
Residual standard error: 534.1 on 1997 degrees of freedom
Multiple R-squared:  0.8269,    Adjusted R-squared:  0.8267
F-statistic:  4770 on 2 and 1997 DF,  p-value: < 0.00000000000000022
```

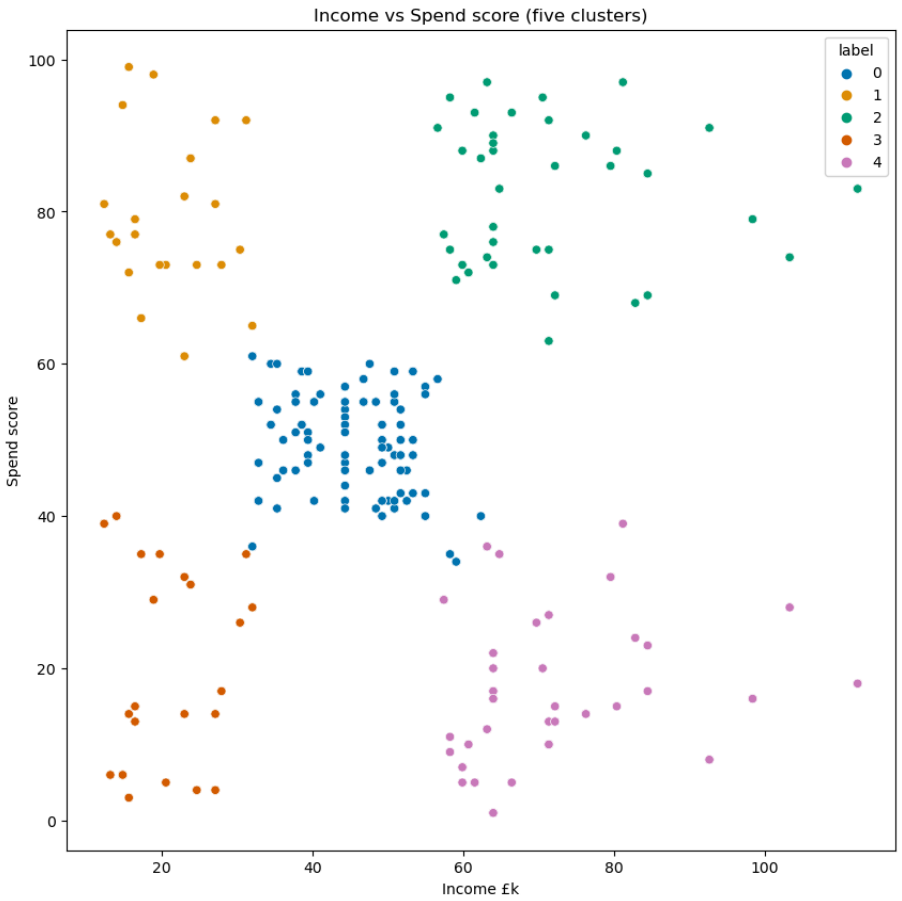
K-means clustering

Used the Elbow method and Silhouette Score to determine the ideal number of clusters, and both indicate that 5 clusters are optimum.

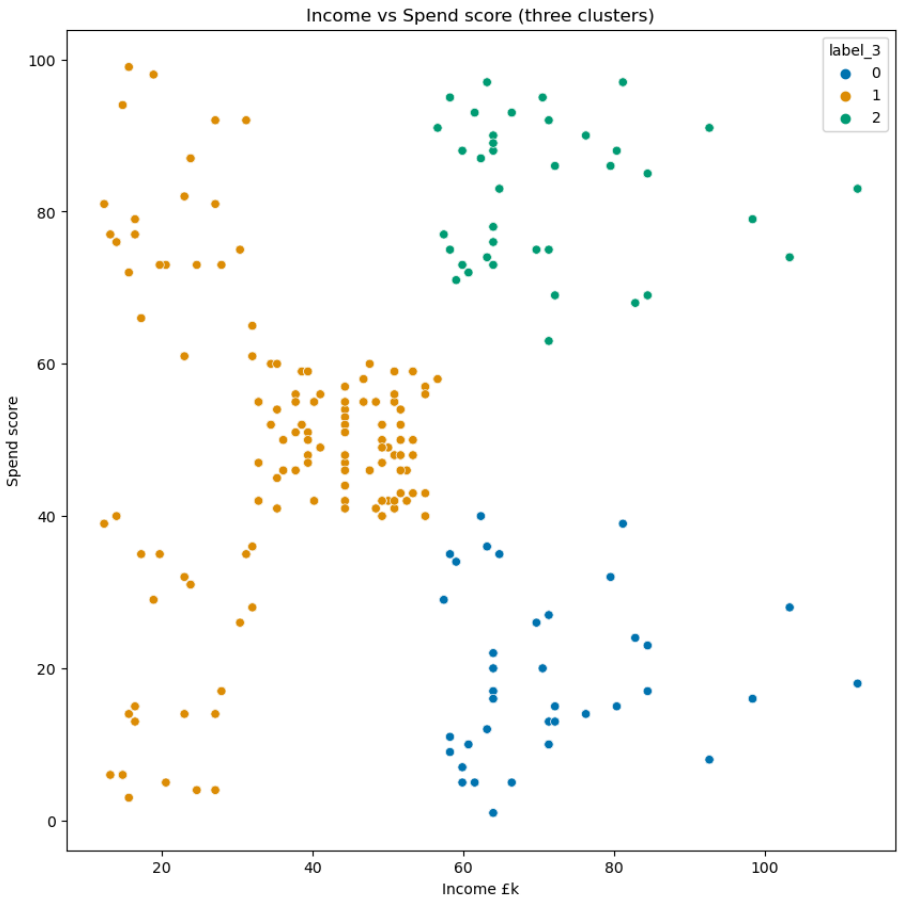


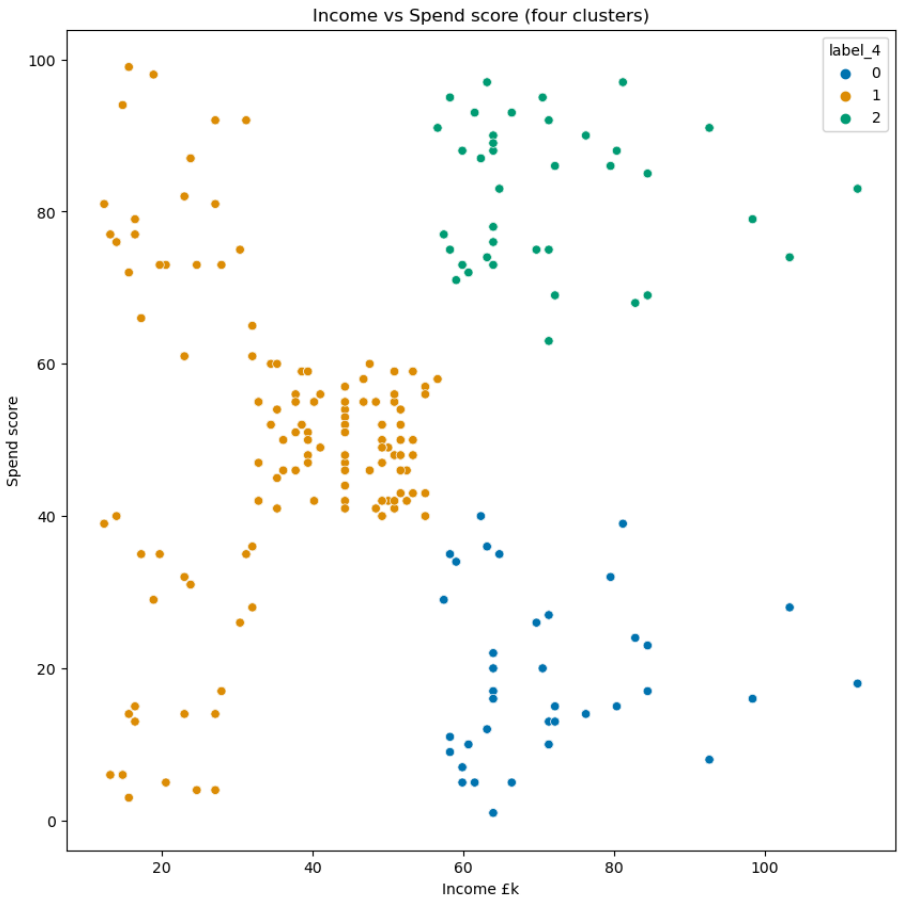


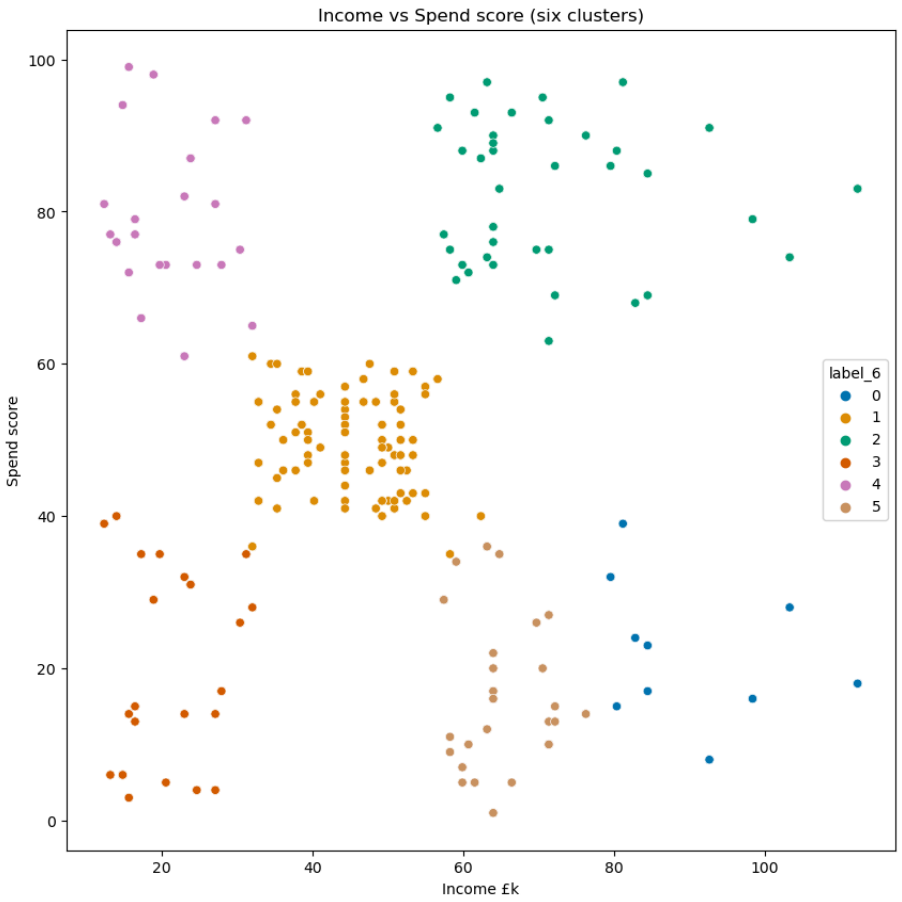
From the initial scatterplot, it's clear there are five distinct clusters. The four clusters on the outer edges are of similar size, and the central cluster (blue) has a bigger concentration of data points.



I fitted the model with 3, 4 and 6 clusters to compare and none of them performed as well.

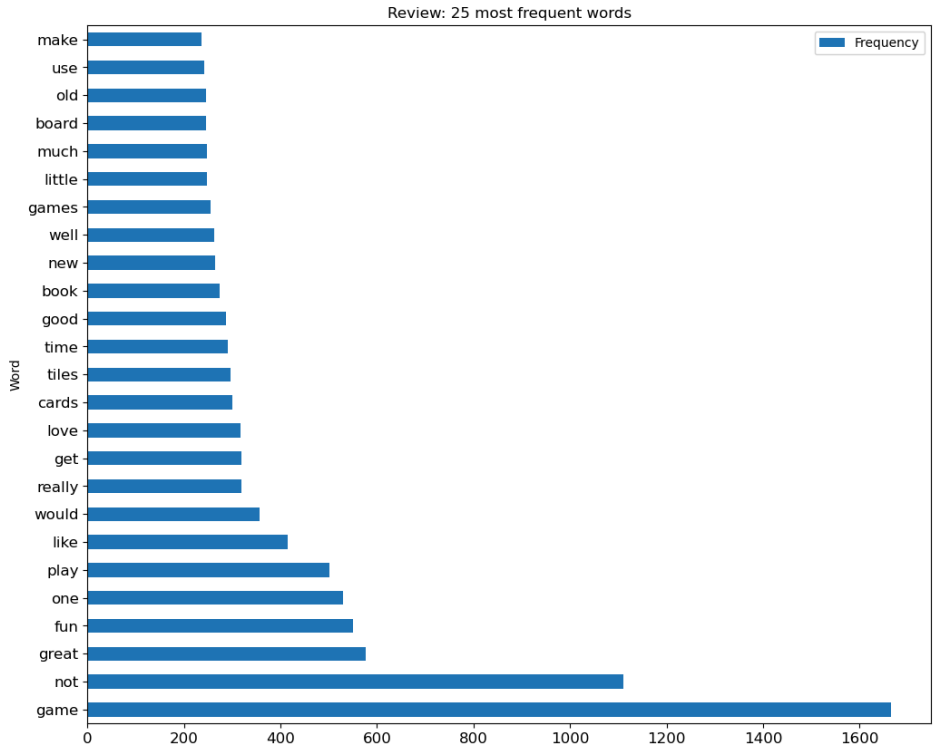

Income vs Spend score (three clusters)
label_3
0
1
2
Spend score
Income £k
0
20
40
60
80
100
20
40
60
80
100

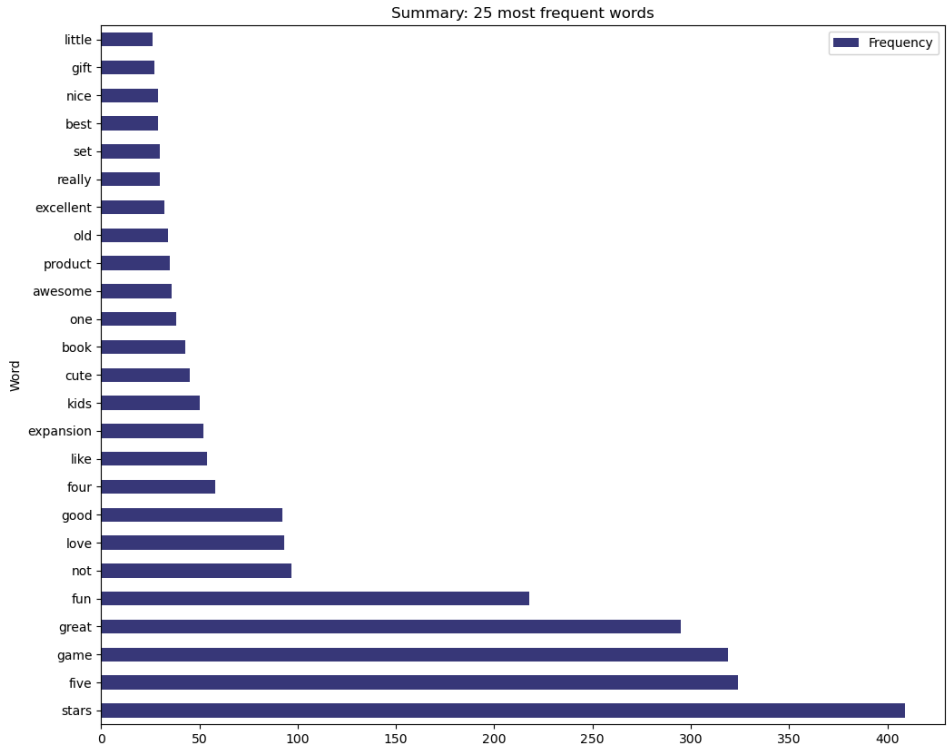

Income vs Spend score (four clusters)
label_4
0
1
2
Spend score
Income £k
0
20
40
60
80
100
20
40
60
80
100



Word frequency and Word Clouds

I also analysed word frequency and created Word Clouds. However, I feel these results do not give as much actionable insight as the Vader and TextBlob sentiment analysis, apart from indicating that star rating was prevalent in Summary column.


Review: 25 most frequent words
Frequency
Word
make
use
old
board
much
little
games
well
new
book
good
time
tiles
cards
love
get
really
would
like
play
one
fun
great
not
game
0
200
400
600
800
1000
1200
1400
1600


Summary: 25 most frequent words
Frequency
Word
little
gift
nice
best
set
really
excellent
old
product
awesome
one
book
cute
kids
expansion
like
four
good
love
not
fun
great
game
five
stars
0
50
100
150
200
250
300
350
400

Review WordCloud

Summary WordCloud

```
Number of occurences of star rating in Summary column:

Six stars:  0
6 stars:  1

Five stars:  322
5 stars:  1

Four stars:  57
4 stars:  0

Three stars:  14
3 stars:  1

Two stars:  12
2 stars:  0

One star:  8
1 star:  1

Zero stars:  1
0 stars:  0
```

Get in touch

lilliana.golob@gmail.com | 07501 450668