Ecommerce Clients Machine Learning Project with Linear Regression

PREPARED BY
SHAKIL AHAMMED

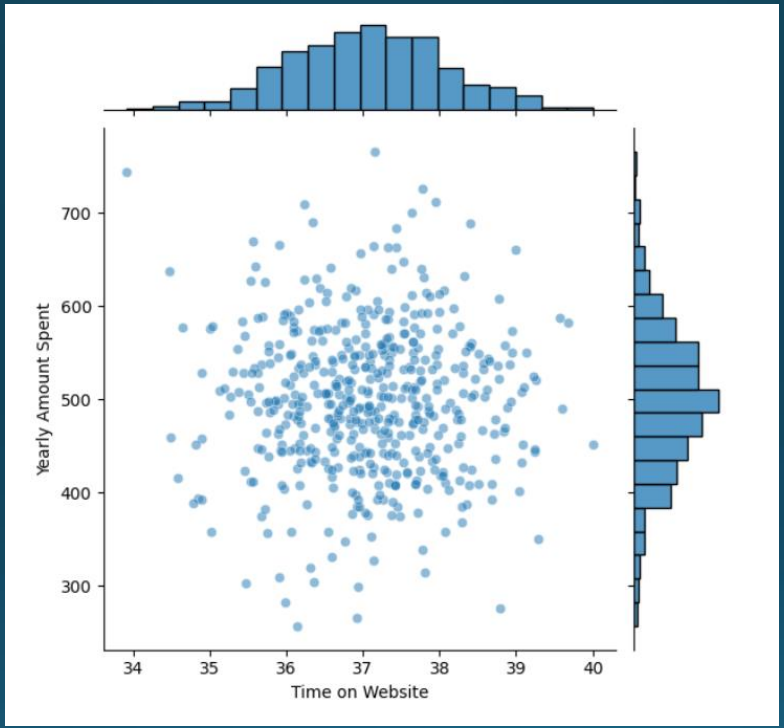
Objective

In this project we work with a dataset which includes information about customers of an e-commerce website, including the following:

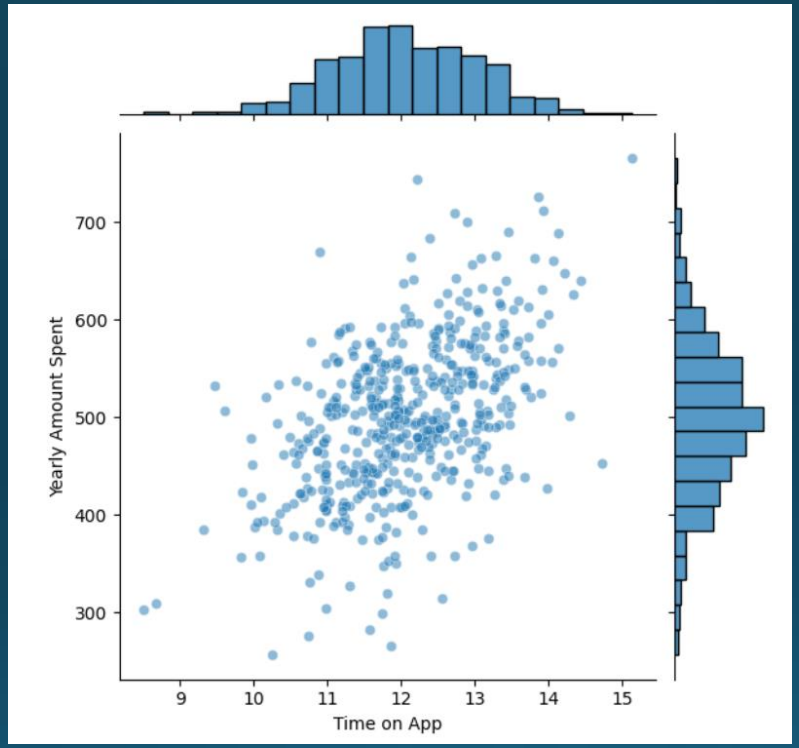
- Avg. Session Length: Average session of in-store style advice sessions.
- Time on App: Average time spent on App in minutes.
- Time on Website: Average time spent on Website in minutes.
- Length of Membership: How many years the customer has been a member.

In this project, we suppose that the company is trying to decide whether to focus their efforts on their mobile app experience or their website. We are here to help them make a data-driven decision.

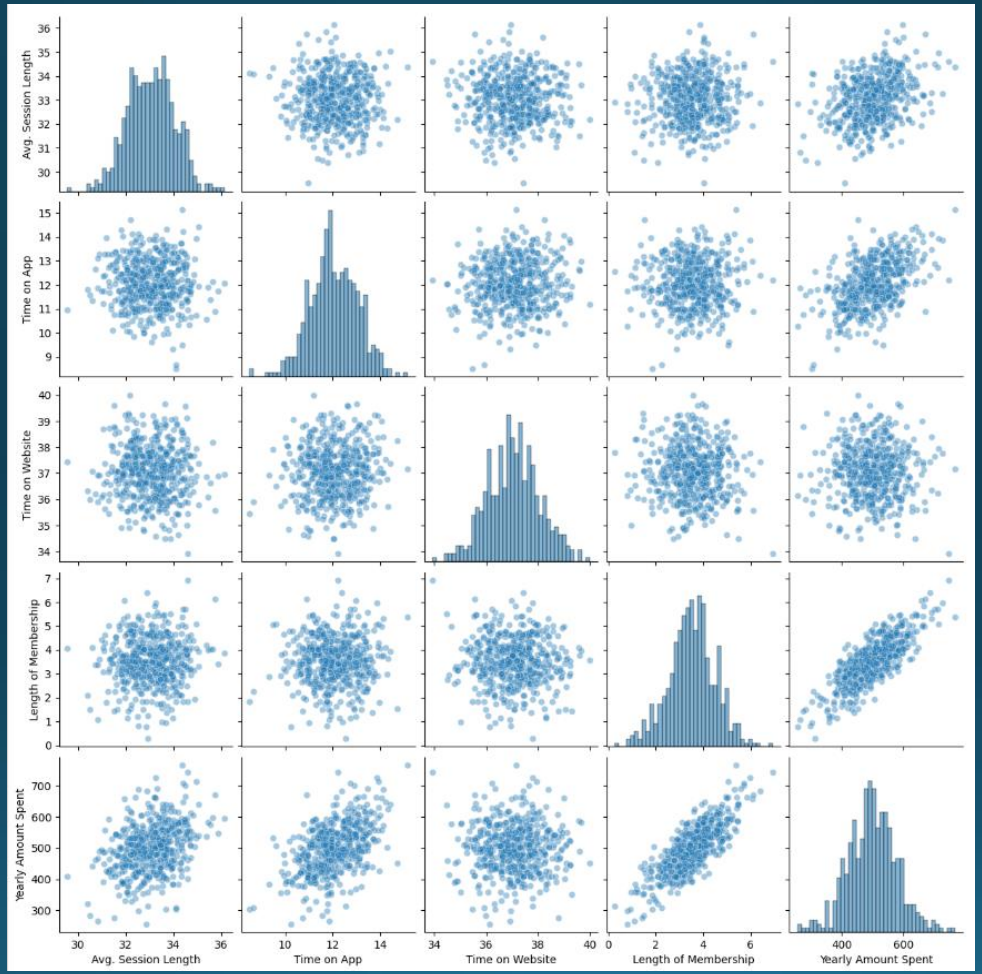
Insights



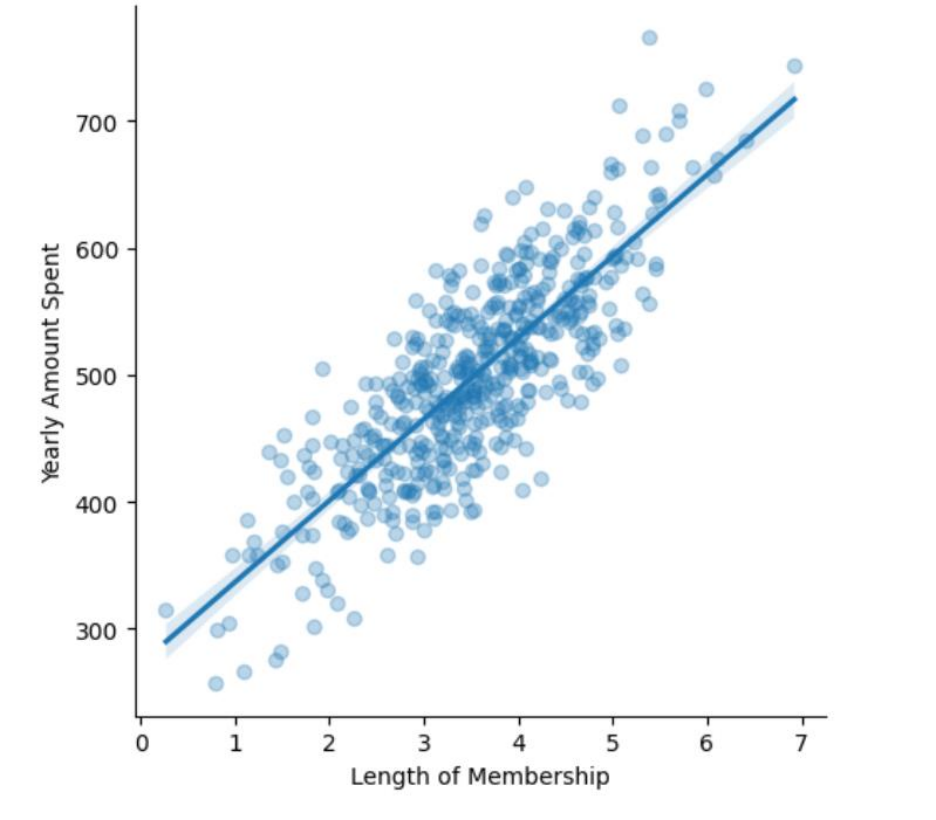
Time on website vs yearly amount spent



Time on app vs yearly amount spent



Correlation between all column



Length of membership vs yearly amount spent

Splitting the data

```
from sklearn.model_selection import train_test_split
```

```
X = customers[['Avg. Session Length', 'Time on App', 'Time on Website', 'Length of Membership']]
y = customers['Yearly Amount Spent']
```

```
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.3, random_state=42)
```

X are the predictors, and y is the output. What we want to do is create a model that will take in the values in the X variable and predict y with a linear regression algorithm. We will use the SciKit-Learn library to create the model.

```
from sklearn.linear_model import LinearRegression
```

```
lm = LinearRegression()
```

```
lm.fit(X_train, y_train)
```

```
▾ LinearRegression
LinearRegression()
```

```
# r squared
lm.score(X_test, y_test)
```

```
0.9808757641125855
```

```
lm.score(X_train, y_train)
```

```
0.9854085989105928
```

we create the model and feed the training data to it. This model will tell us which input has the biggest impact in the output (yearly expenditure). As the plots suggested, we find that the most important coefficient is that of the "Length of Membership" predictor, followed by the 'Time on App' and the 'Avg. Session Length'. The time on website does not seem to be an important factor to the amount a customer spends per year.

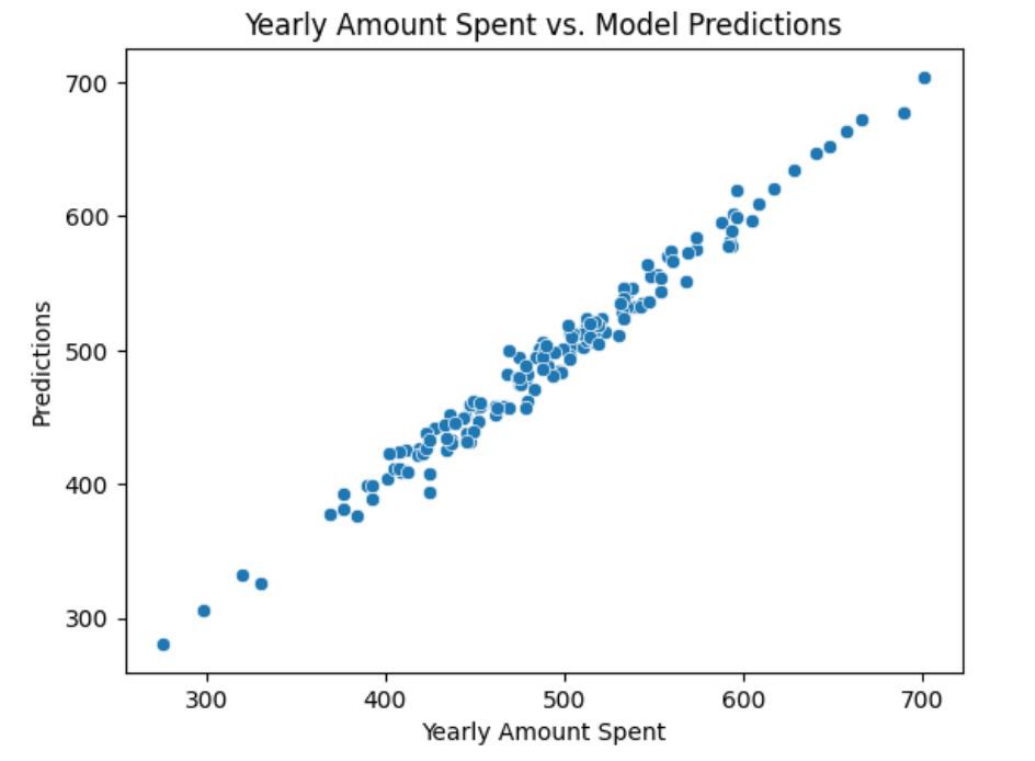
```
prediction= lm.predict(X_test)
prediction
```

```
array([403.66993069, 542.57756289, 427.06591658, 502.02460425,
       410.12143559, 569.93442508, 531.93431341, 506.29650969,
       408.71870658, 473.97737105, 441.46912726, 425.33703059,
       425.1297229 , 527.61676714, 431.45684016, 424.0769184 ,
       575.76543296, 484.89856554, 458.35936863, 481.96502182,
       502.32441491, 513.63783554, 507.58877002, 646.57464283,
       450.24372141, 496.27043415, 556.40457807, 554.95630839,
       399.64237199, 325.84623136, 532.89783259, 478.12238702,
       501.05701845, 305.97335848, 505.77244448, 483.79591969,
       518.8331528 , 438.18241857, 456.71094234, 471.04609461,
       494.44008972, 445.31155755, 508.78802753, 501.04594193,
       488.83499673, 535.38079541, 595.20129802, 514.04714872,
       280.76758312, 433.10112367, 421.70823427, 481.23640152,
```

```
y_test
```

```
361    401.033135
73     534.777188
374    418.602742
155    503.978379
104    410.069611
          ...
266    554.003093
23     519.340989
222    502.409785
261    514.009818
426    530.766719
```

Now that the model is trained, we should be able to use it to make our predictions and evaluate our model. The scatter plot below plots the actual y values to the model's predictions. The model seems to behave accurately almost 98%.



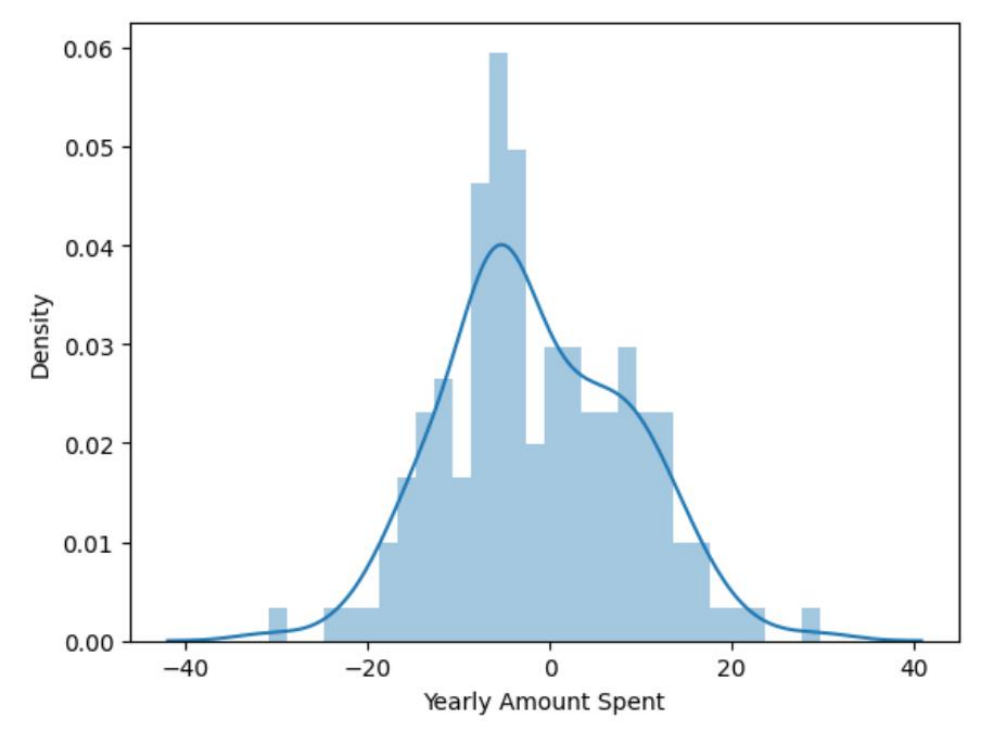
Scatter plot of actual values of y vs predicted values.

Evaluation of the model

```
from sklearn.metrics import mean_squared_error, mean_absolute_error
import math
```

```
print('Mean Absolute Error:',mean_absolute_error(y_test, predictions))
print('Mean Squared Error:',mean_squared_error(y_test, predictions))
print('Root Mean Squared Error:',math.sqrt(mean_squared_error(y_test, predictions)))
```

```
Mean Absolute Error: 8.426091641432116
Mean Squared Error: 103.91554136503333
Root Mean Squared Error: 10.193897260863155
```



Distribution plot of the residuals of the model's predictions. They should be normally distributed.

```
prediction= lm.predict([[33.0, 12.0, 39.0, 5]])
prediction
```

```
array([587.69283517])
```

If we input

Avg. Session Length= 33 min, Time on App= 12 min, Time on Website= 39 min, Length of Membership= 5.

Our prediction Output(Yearly Amount Spent) is= 587.69 minutes.

Conclusion

It can be tricky to interpret the information in this analysis. According to the model, the most significant factor for clients is not the time spent on the app or website, but their length of membership. However, of the two predictors (desktop vs app), the app has the strongest influence by far. In fact, the time spent on the desktop website does not seem to have any correlation at all! In other words, according to the data, the amount of time that the customer spends on the desktop website has almost nothing to do with the amount of money they will spend.

Suggestions

We could interpret this in two different ways. Firstly, this could mean that the desktop website needs more work to make its visitors buy more. Secondly, it could mean that people tend to be more influenced by mobile applications of online stores than by desktop websites. So maybe efforts should be directed towards taking advantage of this fact. Indeed, the interpretation of this information requires expertise in the online marketing sphere. Our analysis and our model, however, does a very good job in weighting the predictors importance.