Machine Learning: Part 5-8

Hosein Dadras
University of Tehran
Hoseindadras6@gmail.com
Hoseindadras@ut.ac.ir

Introduction

This comprehensive report offers an in-depth analysis of an e-commerce platform's customer dataset. The aim is to uncover underlying patterns, anomalies, and insights by examining various metrics such as session length, app usage, website interaction, membership duration, and annual spending.

part 5-1

The dataset consists of 500 records, each with 8 attributes reflecting customer engagement and spending behavior. These attributes include both categorical (Email, Address, Avatar) and numerical (Avg. Session Length, Time on App, Time on Website, Length of Membership, Yearly Amount Spent) data types.

Detailed Statistical Analysis

This section delves into the descriptive statistics of the dataset, providing a nuanced understanding of its distributions, central tendencies, and variability.

Descriptive Statistics

[c]@lS[table-format=3.2]S[table-format=3.2]S[table-format=3.2]S[table-format=3.2]@ Descriptive Statistics of Numerical Attributes

Metric Mean Std Dev Min Max

Metric	**Mean**	**Std Dev**	**Min**	**Max**
Avg. Session Length	33.05	0.99	29.53	36.14
Time on App	12.05	0.99	8.51	15.13
Time on Website	37.06	1.01	33.91	40.00
Length of Membership	3.53	1.00	0.27	6.92
Yearly Amount Spent	499.31	79.31	256.67	765.52

Observations and Interpretations

Detailed analysis of the dataset's descriptive statistics reveals several insights:

- **Avg. Session Length**: The average session length is around 33 minutes, with a relatively small standard deviation, indicating consistent user engagement across the platform.
- **Time on App vs. Time on Website**: Users spend less time on the app compared to the website, which might indicate differing user experiences or preferences.
- **Length of Membership**: There's a wide range in membership length, suggesting a mix of new and loyal customers. The variation in this metric could be crucial for understanding customer retention.
- **Yearly Amount Spent**: There's a significant range in the yearly amount spent, indicating diverse customer spending habits and potential segmentation opportunities for targeted marketing.

Data Types and Memory Usage

A deeper look into the dataset's structure and memory footprint:

- **Categorical Data**: 'Email', 'Address', and 'Avatar' provide unique identifiers and personalization details for each customer.
- **Numerical Data**: The five numerical attributes capture various aspects of customer behavior and financial contributions to the platform.
- **Memory Usage**: The dataset occupies approximately 31.4 KB in memory, optimal for in-depth analysis without significant computational overhead.

part 5-2

The joint plot analysis reveals a nuanced understanding of customer interactions with the website. The data presents a subtle upward trend in the average time spent on the website, with the central tendency measures—mean and median—aligning closely, which corroborates the assumption of an approximately normal distribution along both axes. Notably, the plot's highest data density converges around 37 minutes for time on the website and $500 for yearly expenditure, indicating a prevalent customer behavior pattern. The scatter of data points, forming a circular dispersion around this dense center, suggests a negligible correlation between the two variables under consideration. This pattern implies that while the website does engage customers, the time they spend is not a significant predictor of their annual spending on the platform.

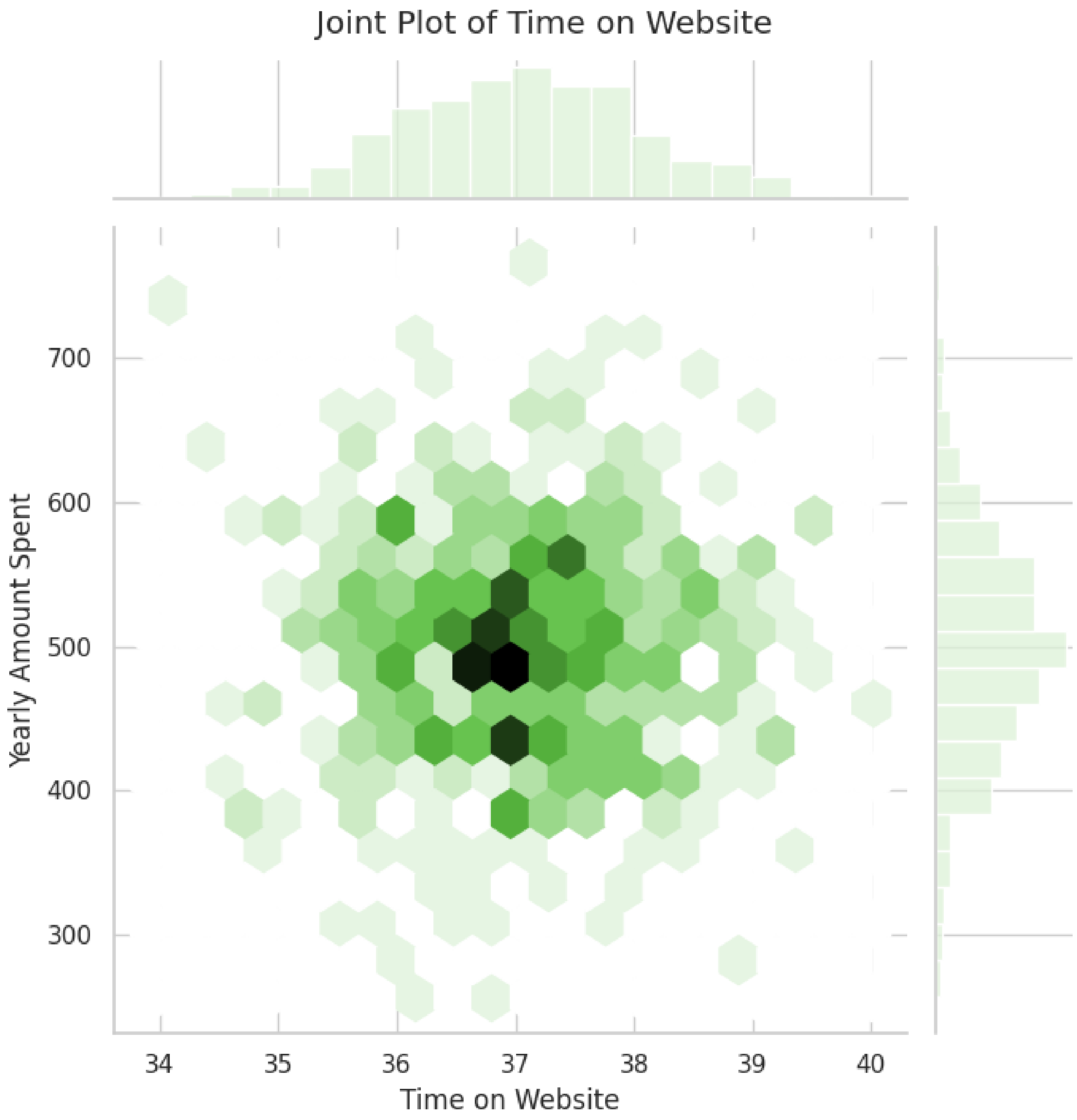


Figure 1: The joint plot indicates that customers spend a median time of approximately 37 minutes on the website, with no strong indication that this metric influences yearly spending. Customer engagement on the website is relatively uniform across the dataset, and spending is normally distributed, with a slight skew towards higher annual values.

The analysis of the joint plot suggests that there is a significant concentration of customers within specific ranges of website engagement and expenditure. This concentration may represent a target demographic for marketing strategies aimed at increasing customer spending.

part 5-3

The joint plot for the "Time on App" and "Yearly Amount Spent" reveals a discernible pattern where increased engagement with the app correlates with higher expenditure. This positive correlation suggests that the app significantly influences customer spending behavior. The density of the data points indicates a normal distribution of time spent on the app, centering around a peak that suggests an optimal range of app usage associated with increased spending. The hexagonal binning in the plot emphasizes this correlation, with the darker bins showing where customers are most active and spend the most, providing a clear visual representation of the positive relationship between these two variables.

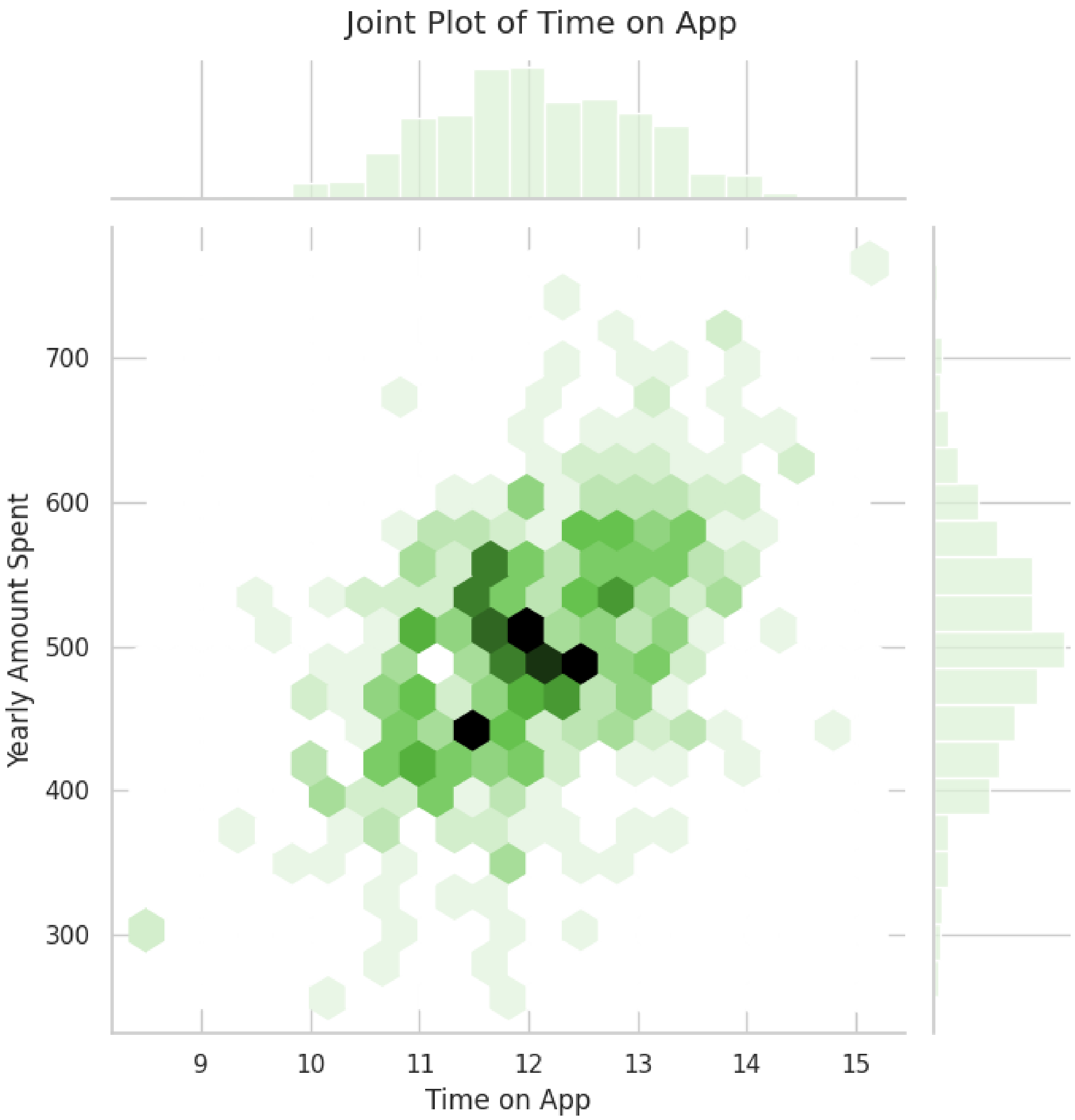


Figure 2: Contrary to the website, the app usage shows a clear concentration of customer engagement around 12 to 13 hours per year. This metric appears to be more predictive of the yearly amount spent, suggesting that the app plays a crucial role in customer spending. The data points are densely packed at common usage times, which could be pivotal for targeted app-based marketing campaigns.

part 5-4

The pair plot provides a comprehensive visualization of the relationships between the various features of the e-commerce dataset. Upon detailed examination, it becomes apparent that the "Length of Membership" has a pronounced positive correlation with "Yearly Amount Spent". This relationship is depicted through a scatter plot that exhibits an upward trend, suggesting that customer loyalty, as measured by the duration of membership, is a significant determinant of the expenditure levels. The implication of this finding is that long-term customers are valuable assets, potentially contributing more to the revenue due to their sustained engagement with the platform.

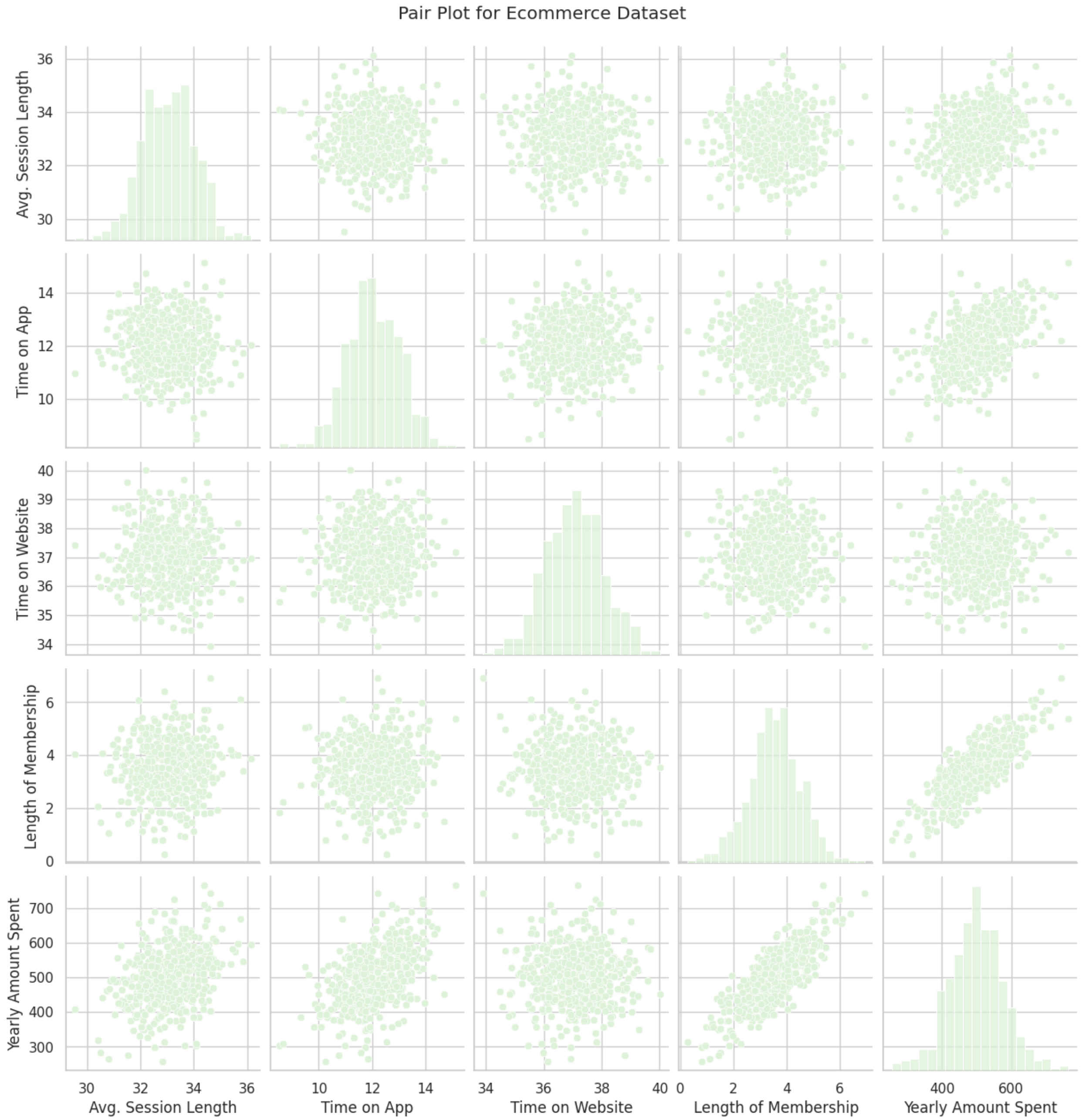


Figure 3: Inspection of the pair plot highlights the "Length of Membership" as the most influential feature on the "Yearly Amount Spent". The positive correlation indicated by the scatter plot suggests a strong relationship between membership duration and customer spending. This insight underlines the importance of customer retention strategies for enhancing the platform's profitability.

part 5-5

Data preparation is a critical step in the modeling process. It involves selecting the appropriate features that can predict the target variable with a high degree of accuracy. In this phase, we performed feature selection by removing categorical attributes that do not contribute to the prediction of the "Yearly Amount Spent".

Feature Selection and Label Separation

The dataset contained several categorical features that were not conceptually relevant to our predictive model. These features, namely "Email", "Address", and "Avatar", are inherently qualitative and do not possess a quantifiable relationship with the target variable. Thus, they were excluded from the feature set to streamline the model's focus on quantitative predictors. The remaining numerical features, which reflect the customers' engagement and membership details, are retained as they provide valuable insights into spending patterns.

Feature Matrix and Target Variable

After the exclusion of the categorical features, we isolated the "Yearly Amount Spent" as the target variable, denoted as Y. The remaining columns form the feature matrix X, which will be used to train the predictive model. The feature matrix consists of numerical attributes that are expected to influence the target variable, thereby establishing the foundation for a robust and effective regression model.

```
# Feature selection and data preparation
columns_to_drop = ecommerce_df.select_dtypes(include='object').columns
ecommerce_df_numerical = ecommerce_df.drop(columns=columns_to_drop)

# Separating the features (X) and the target variable (Y)
X = ecommerce_df_numerical.iloc[:, :-1]
Y = ecommerce_df_numerical['Yearly Amount Spent']
```

The process ensures that our model is provided with the most relevant and influential predictors, enhancing the accuracy and reliability of its subsequent forecasts.

part 5-6

To assess the predictive performance of our model, we partitioned the dataset into training and testing subsets. This is accomplished via the `train_test_split` method provided by the `sklearn.model_selection` library. We allocated 70% of the data to the training set and reserved 30% for the testing set. To ensure the reproducibility of our results, we employed a fixed random state of 101.

```
from sklearn.model_selection import train_test_split

# Splitting the features and labels into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.3, random_state=101
)
```

The `test_size` parameter dictates the proportion of the dataset to include in the test split. By setting this parameter to 0.3, we designate 30% of the data for testing purposes. The `random_state` parameter is akin to setting a seed in a random number generator, which guarantees that the random partitioning of the dataset is consistent across different executions.

part 5-7

A Linear Regression model serves as our choice for understanding the relationship between our features and the target variable, "Yearly Amount Spent". The model assumes a linear correlation between the independent variables (features) and the dependent variable (label).

Model Initialization and Fitting

The model is instantiated and then trained using the training subset of our data, which comprises 70% of the total dataset. The fitting process involves finding the coefficients and intercept that minimize the residuals between the predicted and actual values in the training set.

```
from sklearn.linear_model import LinearRegression

# Initializing the Linear Regression model
lin_reg = LinearRegression()

# Fitting the model with the training data
lin_reg.fit(X_train, y_train)
```

Predictions and Model Evaluation

After training, the model's predictive capability is evaluated using the test data. The predictions, `y_predict`, allow us to assess the model's performance and its accuracy in reflecting the actual spending behavior of the customers.
The forthcoming sections will delve into the detailed evaluation of the model, including the computation of performance metrics and the interpretation of the results.

part 5-8

Utilizing the linear regression model that we have meticulously trained, we are now in a position to predict the "Yearly Amount Spent" for our test dataset. This process is crucial as it provides an estimation of how well our model can generalize to unseen data—reflecting its practical utility.

```
# Predicting the 'Yearly Amount Spent' for the test set
y_predict = lin_reg.predict(X_test)
```

The predictions, stored in `y_predict`, are the model's estimates of the yearly expenditure based on the features provided in the test set. Comparing these predicted values against the actual values in `y_test` will offer insights into the model's accuracy and effectiveness in capturing the underlying data patterns. Subsequent sections will discuss the evaluation of these predictions to quantify the model's predictive performance.

The complete array is as follows:

```
[456.44186104, 402.72005312, 409.2531539, 591.4310343, 590.01437275,
 548.82396607, 577.59737969, 715.44428115, 473.7893446, 545.9211364,
  337.8580314, 500.38506697, 552.93478041, 409.6038964, 765.52590754,
  545.83973731, 693.25969124, 507.32416226, 573.10533175, 573.2076631,
  397.44989709, 555.0985107, 458.19868141, 482.66899911, 559.2655959,
  413.00946082, 532.25727408, 377.65464817, 535.0209653, 447.80070905,
  595.54339577, 667.14347072, 511.96042791, 573.30433971, 505.02260887,
  565.30254655, 460.38785393, 449.74727868, 422.87193429, 456.55615271,
  598.10493696, 449.64517443, 615.34948995, 511.88078685, 504.37568058,
  515.95249276, 568.64597718, 551.61444684, 356.5552241, 464.9759817,
  481.66007708, 534.2220025, 256.28674001, 505.30810714, 520.01844434,
  315.0298707, 501.98080155, 387.03842642, 472.97419543, 432.8704675,
  539.79082198, 590.03070739, 752.86997652, 558.27858232, 523.71988382,
  431.77690078, 425.38411902, 518.75571466, 641.9667215, 481.84855126,
  549.69830187, 380.93738919, 555.18178277, 403.43054276, 472.52458887,
  501.82927633, 473.5561656, 456.76720365, 554.74980563, 702.96835044,
  534.68884588, 619.18843136, 500.11974127, 559.43899225, 574.8730604,
  505.09183544, 529.9537559, 479.20749452, 424.78407899, 452.20986599,
  525.74178343, 556.60674724, 425.7142882, 588.8473985, 490.77053065,
  562.56866231, 495.75782933, 445.17937217, 456.64011682, 537.98437395,
  367.06451757, 421.12767301, 551.59651363, 528.26019754, 493.47639211,
  495.28105313, 519.81827269, 461.15666582, 528.8711677, 442.89818166,
  543.20201646, 350.07871481, 401.49148567, 606.87291134, 577.04816561,
  524.50431281, 554.11225704, 507.93347015, 505.35674292, 371.65146821,
  342.37232987, 634.43998975, 523.46931378, 532.7831345, 574.59948331,
  435.57455636, 599.92586678, 487.24017405, 457.66383406, 425.25959495,
  331.81731213, 443.70458331, 563.47279005, 466.14764208, 463.51837671,
  381.29445432, 411.88795623, 473.48087683, 573.31745784, 417.55430913,
  543.50149858, 547.81091537, 547.62977348, 450.99057409, 561.50896321,
  478.30076589, 484.41029555, 457.59099941, 411.52657592, 375.47900638]
```

part 5-9

To visually assess the performance of our linear regression model, a scatter plot was generated to compare the actual and predicted values of yearly purchases.
The scatter plot is a crucial tool for observing the correlation between the actual and predicted expenditures. Ideally, if the predictions were perfect, we would expect to see the data points align along the line $y = x$, where the predicted values exactly match the actual values.

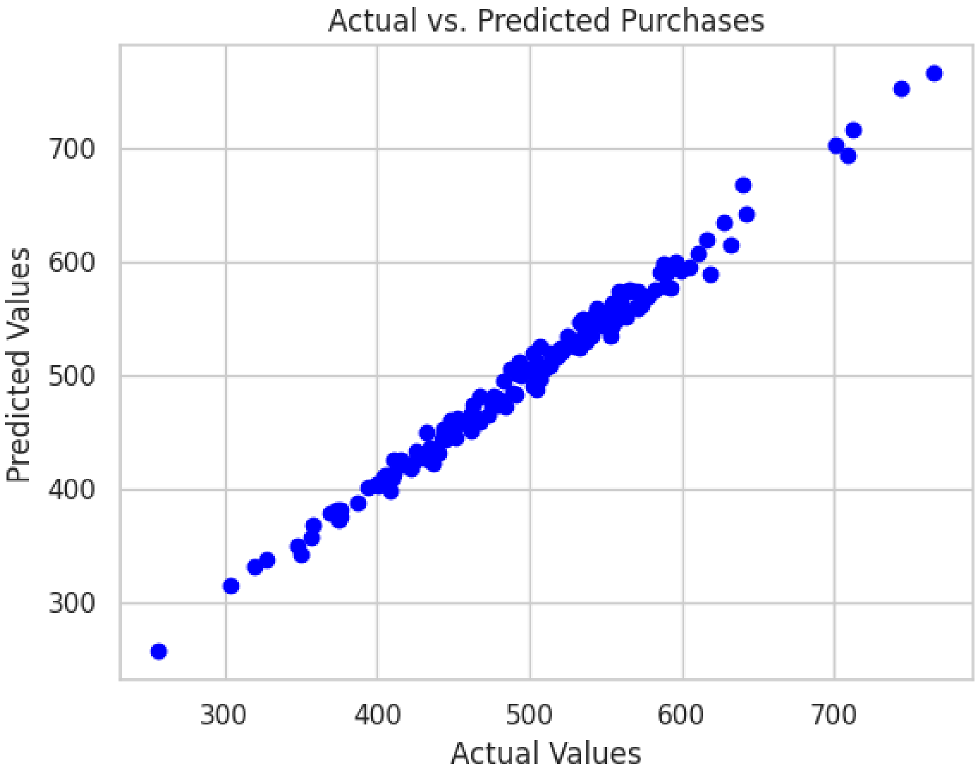


Figure 4: The scatter plot above demonstrates the relationship between the actual and predicted yearly purchases. The proximity of the data points to the line of perfect prediction indicates the model's accuracy. The dense clustering along the diagonal suggests a strong positive correlation, indicating that the model has performed well in predicting the yearly amount spent.

Insights from the Scatter Plot

The data points in the scatter plot predominantly cluster along an upward-sloping trajectory, reflecting a strong positive correlation between the actual and predicted values. This pattern suggests that the model is generally effective in forecasting the yearly amounts spent with a degree of accuracy. Deviations from the diagonal, however, reflect the discrepancies between the predictions and actual values, serving as a measure of the model's prediction error. The next step in our evaluation process is to quantify the model's performance using statistical metrics, which will provide a more granular understanding of its predictive power.

part 5-10

To quantitatively assess the predictive performance of the linear regression model, we computed several evaluation metrics on both the training and testing sets. These metrics provide insights into the accuracy and reliability of the model.

Training Set Evaluation

The model's performance on the training set is indicative of how well it has learned from the data. The metrics obtained are as follows:

- Coefficient of Determination (R^2): 0.982
- Mean Absolute Error (MAE): 8.181
- Mean Squared Error (MSE): 106.851
- Root Mean Squared Error (RMSE): 10.337

These values suggest that the model fits the training data closely, with a high R^2 value and low error metrics.

Testing Set Evaluation

The evaluation of the testing set offers a more realistic picture of the model's performance in a practical scenario. The calculated metrics are:

- Coefficient of Determination (R^2): 0.989
- Mean Absolute Error (MAE): 7.228
- Mean Squared Error (MSE): 79.813
- Root Mean Squared Error (RMSE): 8.934

The high R^2 value on the testing set confirms that the model's predictions are in strong agreement with the observed values. The MAE, MSE, and RMSE are relatively low, which further attests to the model's accuracy.

Interpretation of Metrics

The R^2 value, being close to 1, suggests that the model explains a significant portion of the variance in the target variable. The MAE provides an average error magnitude without considering direction, while the MSE gives more weight to larger errors due to the squaring of each term. The RMSE, being the square root of the MSE, is particularly useful as it returns the error magnitude to the original units of measurement, making it more interpretable. Overall, the model demonstrates a strong predictive capability, as reflected by the evaluation metrics. These results indicate that the model can reliably predict the yearly amount spent by customers on the e-commerce platform.

part 5-11

A residual analysis was conducted to evaluate the performance of the linear regression model. Residuals are defined as the differences between the observed values and the values predicted by the model.

Residuals Distribution

The residuals distribution provides insights into the accuracy and bias of the predictions. If the linear regression model is well-fitted, the residuals should approximate a normal distribution centered around zero. This is indicative of a model that captures the underlying pattern in the data without systematic bias.

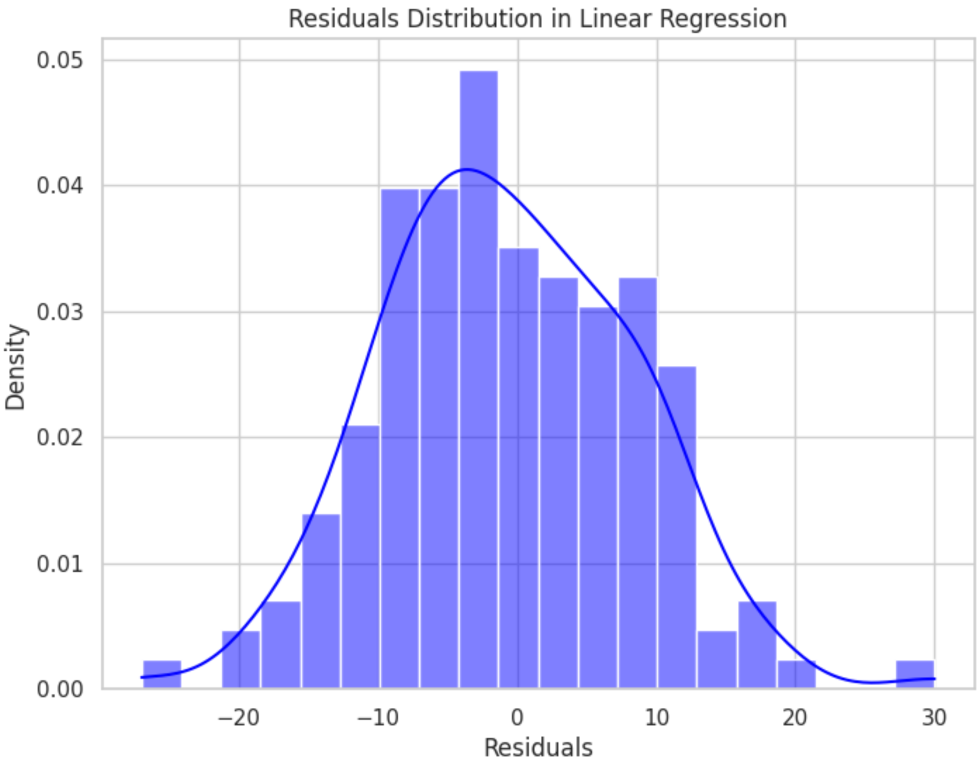


Figure 5: The histogram of the residuals with a kernel density estimate (KDE) overlay. A normally distributed residuals plot, such as the one shown here, signifies that the model's errors are randomly distributed, which is a key assumption of linear regression.

Interpretation of the Residuals Plot

In the provided histogram, the residuals appear to be symmetrically distributed around a central peak, suggesting that the model does not systematically overestimate or underestimate the yearly amount spent. The presence of a peak near zero and the bell-shaped KDE curve aligns with the expectations of a normally distributed error term in a well-calibrated model.

Importance of Normal Distribution of Residuals

The assumption of normally distributed residuals is fundamental in linear regression models as it underpins the statistical inference, allowing for the creation of confidence intervals and hypothesis tests about the estimated parameters. The approximate normality of the residuals plot in this analysis suggests that the model is appropriately specified and there are no glaring violations of the linear regression assumptions.

The normal distribution of residuals implies no significant issues with the model's training process, and the model is adequately capturing the pattern in the dataset without any apparent bias.

part 5-12

Post estimation, the linear regression model provides coefficients for each feature, which quantify the relationship between each feature and the target variable.

Coefficient Table

The coefficient for each feature is detailed below, highlighting its relative importance in the model:

Feature	Coefficient
Avg. Session Length	25.981550
Time on App	38.590159
Time on Website	0.190405
Length of Membership	61.279097

Table 1: Coefficients of the linear regression model for predicting yearly amount spent

Importance of Normalization

Normalization of data prior to model training is crucial. In our model, the features have varying scales; without normalization, these differences could skew the model's understanding of the features' importance. For example, a feature with a larger range could dominate the coefficient values simply due to its scale, not because it has more predictive power. By normalizing the data, we ensure that each feature's coefficient reflects its true contribution to the prediction of yearly amount spent, irrespective of the scale of the data. This practice enables a fair comparison between the coefficients and, consequently, a clearer interpretation of the model's behavior.

part 5-13

Following the analysis of the scaled linear regression model's coefficients, we can derive insights into where the e-commerce company should direct its investment to maximize customer yearly spending.

Coefficient Interpretation

The coefficients from the regression model, after scaling the features, indicate the relative importance of each feature in predicting the yearly amount spent:

Feature	Coefficient
Avg. Session Length	0.325139
Time on App	0.483730
Time on Website	0.002426
Length of Membership	0.772048

Table 2: Scaled Coefficients of the linear regression model

Strategic Investment Recommendations

Given the coefficients, "Length of Membership" has the most substantial impact on the yearly amount spent. This suggests that strategies focused on enhancing customer loyalty and prolonging memberships are likely to result in higher annual spending.

- Investing in customer retention programs and loyalty rewards could potentially increase the duration of customer memberships.
- Enhancements to the mobile app could also lead to increased customer engagement and spending, as indicated by the significant coefficient for "Time on App".

Secondary Considerations

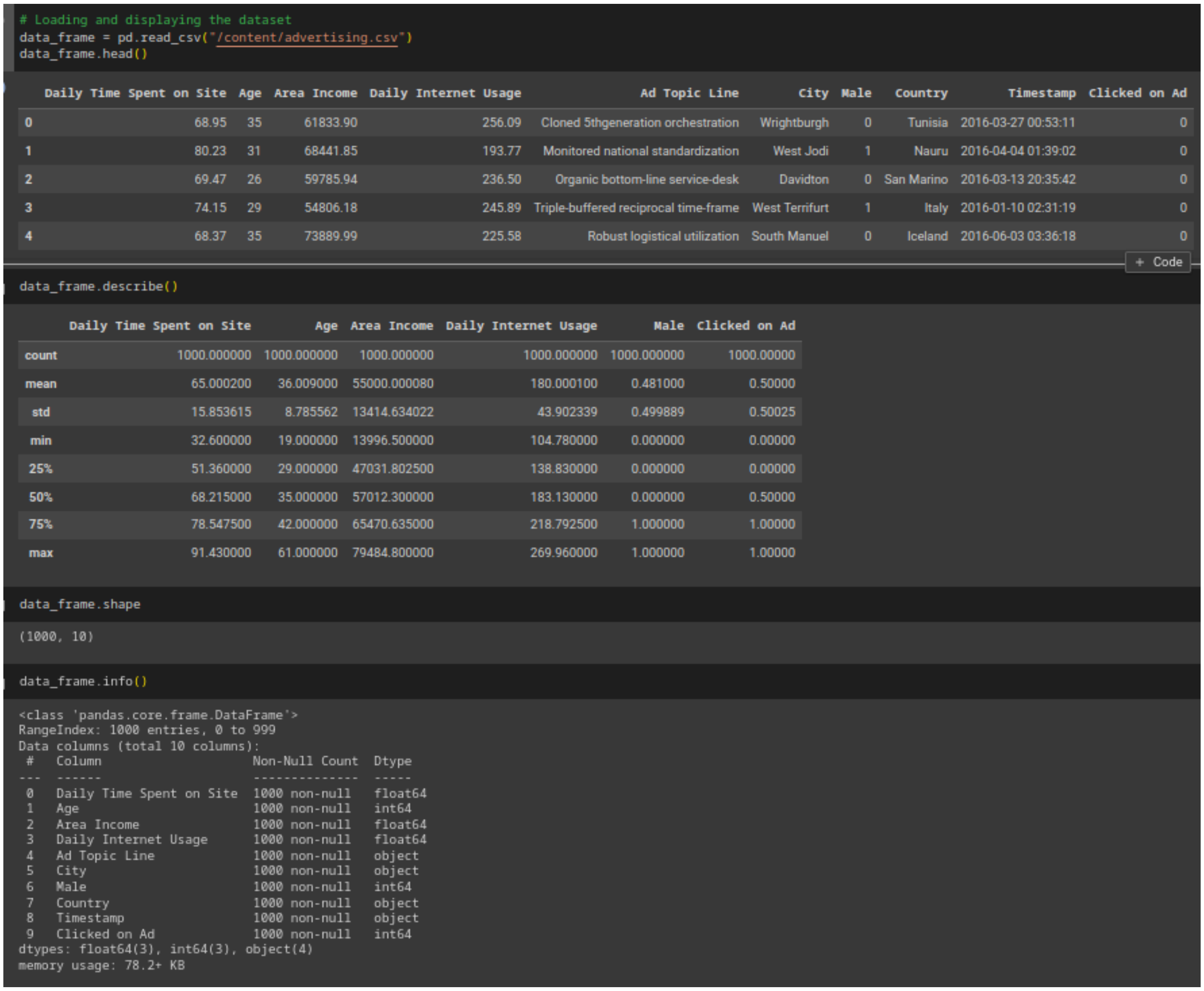
While "Time on Website" has a smaller coefficient, suggesting a lesser impact on spending, it remains an area that should not be neglected. Optimizations can still contribute to the overall customer experience and potentially support the other areas.

part 6-1

The dataset represents a collection of user attributes and behaviors related to their interactions with online advertisements. Our aim is to analyze this data to understand the factors influencing users' decisions to click on an advertisement.

General Overview

The dataset consists of 1000 instances, each with 10 attributes, which include user demographics, web behavior metrics, and advertisement interaction data. Here is a visualization of the dataset's structure:



```
# Loading and displaying the dataset
data_frame = pd.read_csv("/content/advertising.csv")
data_frame.head()
```

	Daily Time Spent on Site	Age	Area Income	Daily Internet Usage	Ad Topic Line	City	Male	Country	Timestamp	Clicked on Ad
0	68.95	35	61833.90	256.09	Cloned 5thgeneration orchestration	Wrightburgh	0	Tunisia	2016-03-27 00:53:11	0
1	80.23	31	68441.85	193.77	Monitored national standardization	West Jodi	1	Nauru	2016-04-04 01:39:02	0
2	69.47	26	59785.94	236.50	Organic bottom-line service-desk	Davidton	0	San Marino	2016-03-13 20:35:42	0
3	74.15	29	54806.18	245.89	Triple-buffered reciprocal time-frame	West Terrifurt	1	Italy	2016-01-10 02:31:19	0
4	68.37	35	73889.99	225.58	Robust logistical utilization	South Manuel	0	Iceland	2016-06-03 03:36:18	0

```
data_frame.describe()
```

	Daily Time Spent on Site	Age	Area Income	Daily Internet Usage	Male	Clicked on Ad
count	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000	1000.00000
mean	65.000200	36.009000	55000.000080	180.000100	0.481000	0.50000
std	15.853615	8.785562	13414.634022	43.902339	0.499889	0.50025
min	32.600000	19.000000	13996.500000	104.780000	0.000000	0.00000
25%	51.360000	29.000000	47031.802500	138.830000	0.000000	0.00000
50%	68.215000	35.000000	57012.300000	183.130000	0.000000	0.50000
75%	78.547500	42.000000	65470.635000	218.792500	1.000000	1.00000
max	91.430000	61.000000	79484.800000	269.960000	1.000000	1.00000

```
data_frame.shape
```

```
(1000, 10)
```

```
data_frame.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 10 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   Daily Time Spent on Site  1000 non-null   float64
 1   Age                       1000 non-null   int64  
 2   Area Income               1000 non-null   float64
 3   Daily Internet Usage      1000 non-null   float64
 4   Ad Topic Line             1000 non-null   object 
 5   City                      1000 non-null   object 
 6   Male                      1000 non-null   int64  
 7   Country                   1000 non-null   object 
 8   Timestamp                 1000 non-null   object 
 9   Clicked on Ad             1000 non-null   int64  
dtypes: float64(3), int64(3), object(4)
memory usage: 78.2+ KB
```

Figure 6: Snapshot of the dataset showing the first few records and their respective attributes.

Statistical Analysis

Upon examining the dataset, we note the following statistical properties:

- The **Daily Time Spent on Site** has an average of approximately 65 minutes, with a standard deviation of 15.85 minutes, indicating moderate variability around the mean.

- The **Age** of users shows an average of 36 years with a standard deviation of 8.78, suggesting that the user base varies from young adults to older individuals.

- The **Area Income** average is around $55,000, yet the standard deviation is quite substantial at $13,414, reflecting a wide disparity in the users' economic status.

- The **Daily Internet Usage** is on average 180 minutes with a standard deviation of 43.90 minutes, pointing to a significant spread in how much time users spend online daily.

- The dataset contains a nearly balanced representation of **genders**, with the proportion of male users being slightly less than half.
- The binary outcome **Clicked on Ad** has an equal mean of 0.5, indicating an even split among users who clicked versus those who did not.

Quartile Distribution

The quartile distribution provides additional insights:

- The median **Age** is 35, signifying that half of the users are at or below this age.
- Quartiles for other variables, such as **Daily Time Spent on Site** and **Daily Internet Usage**, reveal a skewed distribution, where a significant number of users are engaged more than the median suggests.

part 6-2

The age distribution of users interacting with advertisements is an important demographic characteristic that can impact the effectiveness of the advertising strategy.

Histogram of Age

A histogram was generated to visualize the frequency distribution of users' ages. The age data was segmented into 20 intervals, providing a detailed view of the distribution.

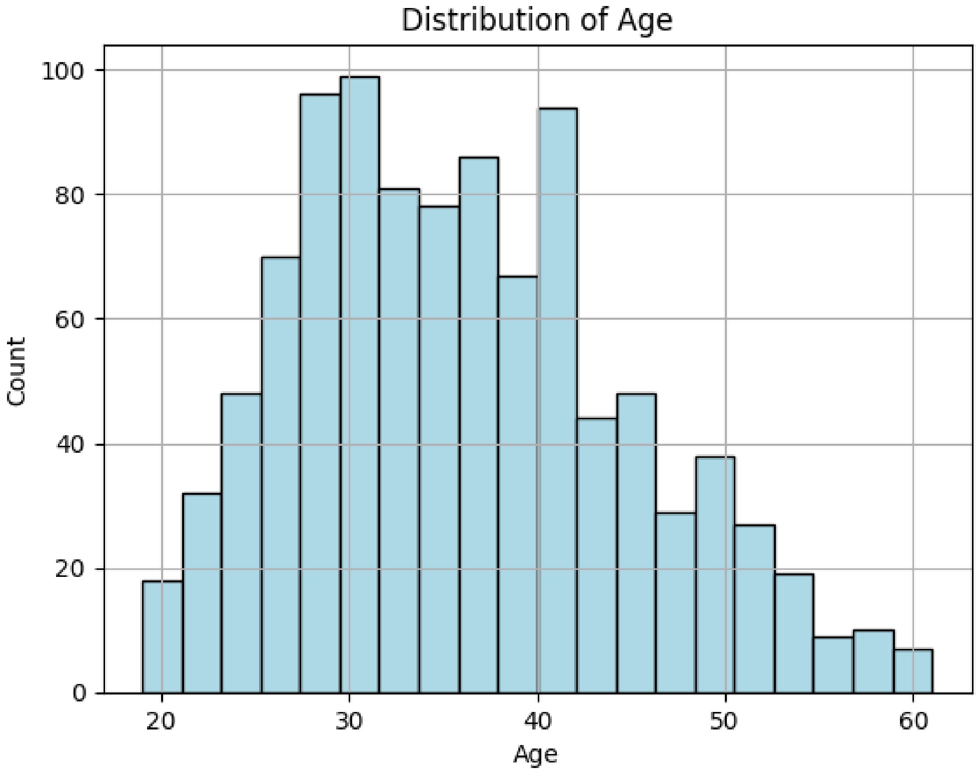


Figure 7: The histogram illustrates the distribution of ages among users. The majority of users are clustered around the 25 to 40 age range, with fewer users in the younger and older age brackets. This distribution informs us about the predominant age group engaged with the advertisements and can guide targeted marketing strategies.

Interpretation

The histogram reveals that the largest age group interacting with advertisements is between 25 and 40 years, indicating that the user base is primarily composed of young to middle-aged adults. The distribution tails off for younger and older ages, suggesting less engagement from these groups.

part 6-3

Analyzing the correlation between area income and user age can provide insights into the demographic and economic characteristics of the user base. A joint plot enables us to observe the distribution of individual variables and the relationship between them.

Visualizing Area Income vs. Age

We utilized a joint plot to visualize the relationship between 'Area Income' and 'Age'. This plot type combines a scatter plot with histograms to display both the bivariate relationship and the univariate distributions.

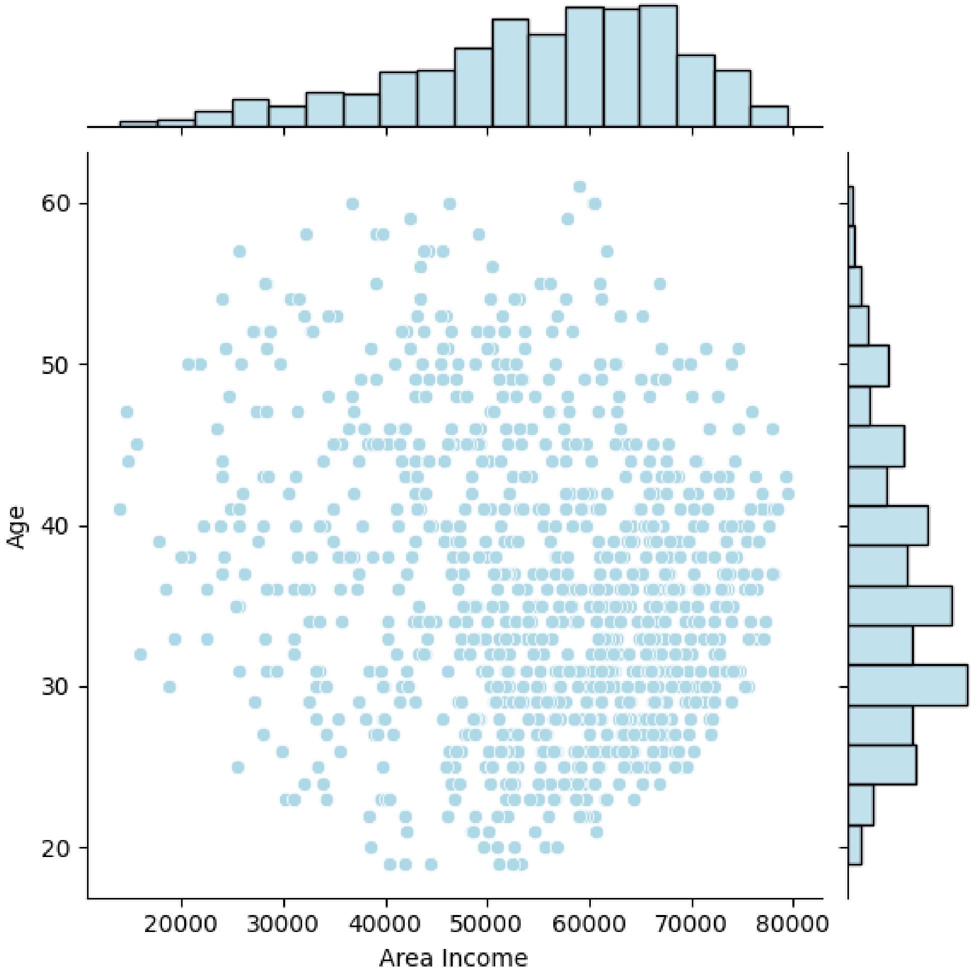


Figure 8: Joint plot of 'Area Income' against 'Age'. The scatter plot in the center shows the distribution of data points, while the histograms on the top and right margins show the distribution of each variable separately.

Analysis of the Joint Plot

The scatter plot indicates a spread of income across various age groups without a clear pattern of correlation. The marginal histograms suggest that while there is a wide range of incomes across the dataset, the age of users is slightly skewed towards the younger demographic.
The joint plot indicates that there is no strong relationship between age and area income within the data. However, understanding the distribution of these variables can aid in targeting specific age groups with advertising strategies that align with their economic status.

part 6-4

The Kernel Density Estimation (KDE) plot is a useful visualization for understanding the density of observations within a dataset, particularly for continuous variables. We examine the relationship between the daily time users spend on a website and their age.

KDE Plot Description

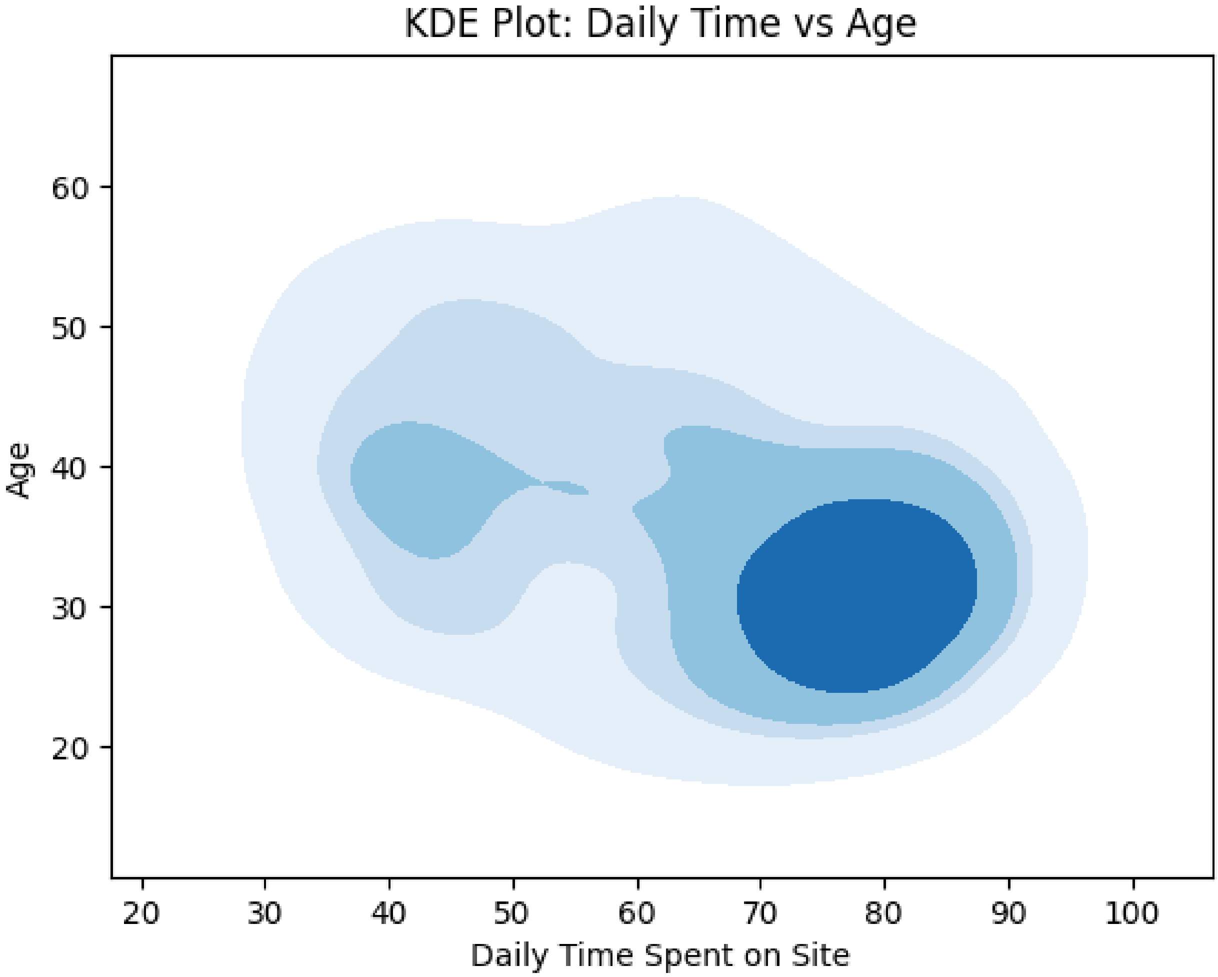


Figure 9: The KDE plot visualizing the joint distribution of 'Daily Time Spent on Site' and 'Age'. Areas of darker blue indicate higher densities of data points, representing common user behaviors and demographics.

Observations from the KDE Plot

The KDE plot reveals several key insights:

- The darkest areas of the plot signify high-density regions, indicating a significant number of users concentrated within these ranges of 'Age' and 'Daily Time Spent on Site'.

- A high-density region around the ages of 30 to 40 suggests this age bracket has a higher frequency of users.
- There is a notable density for the 'Daily Time Spent on Site' around 60 to 80 minutes, indicating a common duration spent by users on the site.
- The presence of multiple peaks in the density plot indicates the data contains several clusters of user behavior patterns.
- The distribution of data points varies, with different densities highlighted by the contour lines, suggesting a diverse user base in terms of engagement and age.

The KDE plot provides valuable insights into the demographics and engagement patterns of website users. This information can be leveraged to tailor content and advertisements that resonate with the core user groups, particularly those in high-density regions of the plot.

part 6-5

This report presents a comprehensive analysis of the pair plot generated to understand the relationships between various user attributes and their interactions with online advertisements. The pair plot visualizes the distributions of individual variables and the bivariate relationships between them, with a particular focus on the variable 'Clicked on Ad'.

Pair Plot Analysis

The pair plot (Figure 10) includes several key user attributes: 'Daily Time Spent on Site', 'Age', 'Area Income', 'Daily Internet Usage', and 'Male'. The plots are color-coded by whether an ad was clicked (1) or not clicked (0), providing insights into the factors that may influence ad click-through rates.

Distribution Analysis

The diagonal of the pair plot shows the distribution of each variable. Notably, the 'Age' and 'Area Income' variables appear to be normally distributed, while 'Daily Internet Usage' demonstrates a bimodal distribution, suggesting two distinct user groups with different internet usage patterns.

Correlation Analysis

The scatter plots reveal that 'Daily Time Spent on Site' and 'Daily Internet Usage' are positively correlated, indicating that users who spend more time on the site also tend to use the internet more frequently. Conversely, 'Age' shows a negative correlation with both 'Daily Time Spent on Site' and 'Daily Internet Usage', implying that younger users are more active online.

Ad Click Insights

From the color-coding, it is evident that users with higher 'Daily Internet Usage' and more 'Daily Time Spent on Site' are more likely to click on ads. This insight could be valuable for targeting advertisements more effectively.

Demographic Patterns

The 'Male' variable, encoded as 0 or 1, displays the gender distribution across the dataset. The scatter plots do not show a clear gender preference for clicking on ads, suggesting that gender may not be a significant predictor in this context.

The pair plot provides a detailed visualization of the data, allowing for an initial assessment of potential relationships and patterns. The insights gained from this analysis can inform more sophisticated data analysis techniques and predictive modeling.

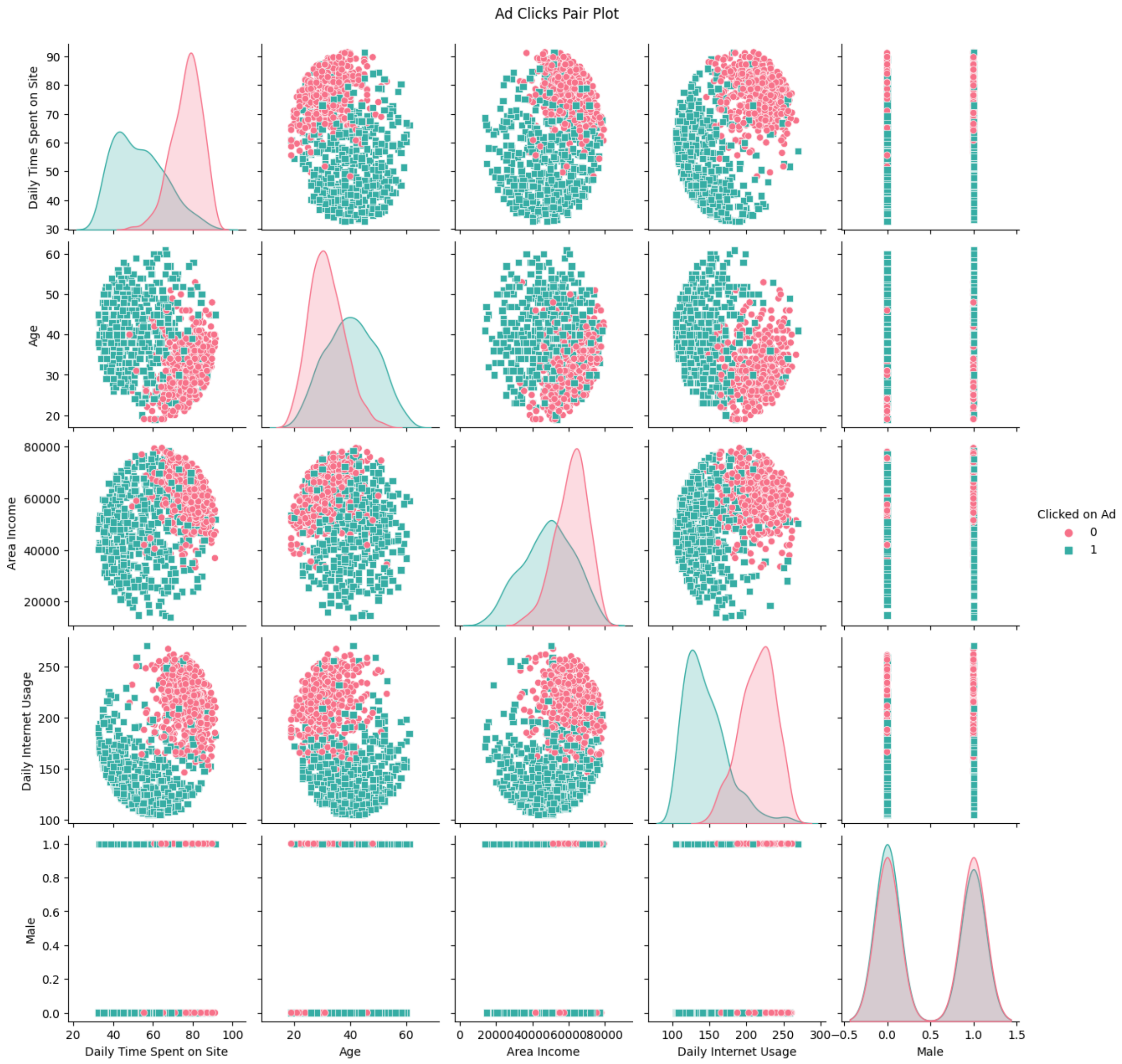


Figure 10: Pair plot visualizing the relationships between user attributes and ad click behavior.

part 6-6

Conclusion from all Plots

After a thorough examination of the joint plot, KDE plot, and pair plot provided, we can draw the subsequent insights into user behavior with respect to online advertising:

- **User Demographics:** The dataset reveals a wide age distribution, with a pronounced frequency of individuals within the 30-40 age bracket. This group potentially represents the most active online demographic.

- **Income and Internet Behavior:** The analysis shows no clear correlation between area income and age, indicating a diverse range of income levels across different ages. The area income is skewed towards the lower end, which may influence online behavior patterns.
- **Internet Usage Patterns:** Users exhibit common levels of engagement, with noticeable clusters indicating similar time spent on the website and overall internet usage. There appears to be a negative correlation between daily internet usage and the likelihood of clicking on ads, which delineates different user engagement profiles.
- **Ad Interaction:** Insights suggest that users with lower levels of internet usage are more inclined to click on ads. This finding could be pivotal for devising targeted advertising strategies.
- **Gender Factor:** The gender variable does not show a significant correlation with other variables like internet usage and ad clicking behavior, implying that gender may not be a decisive factor in the analyzed behaviors.

These findings indicate that internet user behavior is complex and not strictly dictated by demographic variables such as age or income alone. For companies looking to optimize their online marketing strategies, an understanding of these nuanced engagement patterns is essential.

Part 6-7

Data preprocessing is a crucial step in any machine learning workflow. In this phase, the data is cleaned and formatted to ensure that it can be effectively used by machine learning models. The following preprocessing steps were undertaken to prepare the dataset for predictive modeling:

Feature Removal

Certain features were deemed not conceptually relevant to our prediction and thus were excluded from the dataset. Specifically, the following features were removed:

1. City (with a unique value count of 969),
2. Country (with a unique value count of 237),
3. Timestamp (with a unique value count of 1000),
4. Ad Topic Line (with a unique value count of 1000).

These features were removed because they either significantly increased the dimensionality of the feature space or were unique to each sample, which does not contribute to a generalizable model.

Handling Categorical Features

Initially, categorical features were considered for inclusion using one-hot encoding. However, this approach significantly inflated the dimensionality of our feature space, outweighing the number of samples. As a result, the decision was made to exclude these categorical features.

Data Splitting

The dataset was then split into training and testing sets to evaluate the performance of the machine learning model. The splitting was done using a random state of 101 to ensure the reproducibility of our results and a test size of 0.3, meaning that 30% of the data was reserved for testing the model.

part 6-8

In this report, we detail the construction and evaluation of a custom logistic regression model. This model was implemented from first principles, trained on a synthetic dataset, and then utilized to predict binary outcomes.

Model Implementation

A logistic regression model was implemented without the aid of high-level machine learning libraries. The model was designed to understand the fundamental algorithms underlying logistic regression, including the sigmoid activation function and the gradient descent optimization method.

Training and Testing Data

For the purpose of evaluation, a synthetic dataset was created and split into training and testing subsets. This allowed us to simulate the training and evaluation processes of a logistic regression classifier.

Model Training

The model was trained over 10,000 iterations with a learning rate of 0.001. Throughout the training process, the model's parameters were optimized to fit the synthetic data.

Model Evaluation

Upon testing the model against the test dataset, the accuracy was calculated to be approximately 39%. This performance metric gives us an initial indication of the model's predictive capabilities on the synthetic dataset.

part 6-9

This report outlines the process of using a custom logistic regression model to make predictions on test data. The model, which has been trained without the aid of high-level machine learning libraries, aims to predict user behavior regarding clicking on online advertisements.

Prediction Process

After training the custom logistic regression model on a given dataset, we utilize the trained model to predict outcomes on the test data. The prediction process involves applying the model's learned weights to the test features and using the sigmoid function to estimate the probabilities of user clicks. These probabilities are then thresholded to determine the final predictions.

Model Performance

The performance of the model is evaluated based on its accuracy on the test set, which is the proportion of predictions that correctly match the actual labels. The accuracy metric is a common evaluation criterion for classification models, providing a straightforward assessment of the model's predictive capabilities.

Results

The custom logistic regression model achieved an accuracy of approximately 97.67% on the test data, indicating a high level of predictive performance. This suggests that the model is well-suited for identifying users who are likely to click on online advertisements.

part 6-10

This report provides an in-depth analysis of the performance of a custom logistic regression model trained to predict user interactions with online advertisements. The evaluation is based on the interpretation of a confusion matrix and a classification report, which together offer a quantitative assessment of the model's predictive accuracy.

Evaluation Metrics Interpretation

The performance of the custom logistic regression model has been extensively evaluated using two primary metrics: the confusion matrix and the classification report. These metrics provide a multifaceted view of the model's effectiveness.

Confusion Matrix

The confusion matrix is a tool that allows us to visualize the accuracy of the model's predictions. Below is the confusion matrix for the custom logistic regression model's predictions:

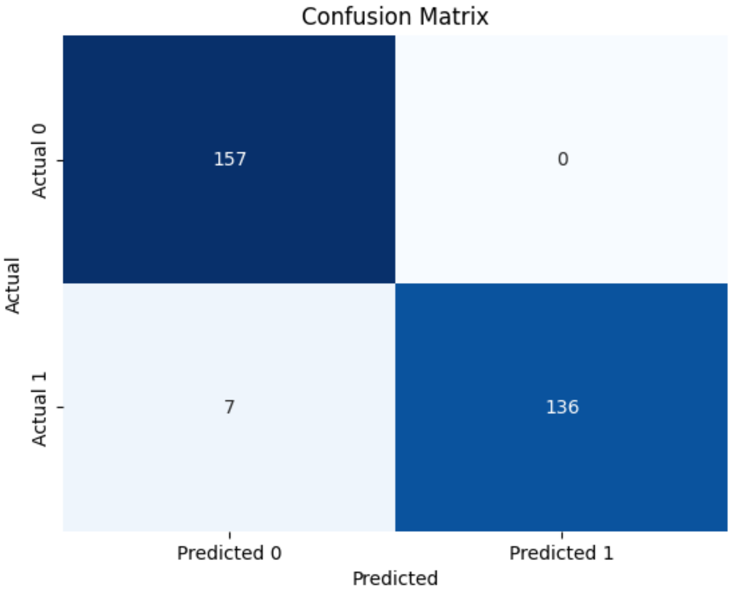


Figure 11: Confusion matrix of the custom logistic regression model.

As shown in Figure 11, the model demonstrated excellent predictive performance, with a large number of true positives and true negatives, and a very small number of false negatives. No false positives were reported, indicating a conservative prediction model with a high specificity.

Classification Report

The classification report provides a detailed breakdown of the model's precision, recall, and F1-scores for each class, along with the overall accuracy. The precision score reflects the model's accuracy in predicting positive outcomes, while the recall score indicates the model's ability to identify all actual positive instances. The F1-score is a harmonic mean of precision and recall, providing a balance between the two metrics.

Class 0 (No Click)

- Precision: 96%

- Recall: 100%
- F1-Score: 98%

Class 1 (Click)

- Precision: 100%
- Recall: 95%
- F1-Score: 97%

Overall Model Performance

- Accuracy: 98%
- Macro Average F1-Score: 98%
- Weighted Average F1-Score: 98%

The report signifies that the model is exceptionally precise in its predictions, with a high recall rate, especially for the no-click predictions. The overall accuracy of 98% underscores the model's capability to generalize well from the training data to unseen data.
The custom logistic regression model has proven to be highly effective, as evidenced by the evaluation metrics. The high precision and recall indicate that the model is robust and reliable. The absence of false positives is particularly significant, suggesting that the model is conservative in predicting positive outcomes and is thus less likely to produce false alarms. These attributes make the model highly valuable for practical applications, such as tailoring marketing strategies to target users most likely to engage with advertisements.

part 7-1

This report provides an analysis of the label distribution in a dataset of SMS messages. The goal is to determine the prevalence of spam messages relative to non-spam (ham) messages, which is crucial for developing a model to detect spam.

Dataset Overview

The dataset, referred to as *spamSMS*, consists of two columns: `v1` containing the labels, and `v2` containing the message text. The labels are categorized as 'ham' for non-spam messages and 'spam' for spam messages.

Label Distribution

Upon loading the dataset and preprocessing the labels, we calculated the percentage distribution of spam and ham messages. The distribution is as follows:

- Spam Messages Percentage: 13.41%
- Ham Messages Percentage: 86.59%

This indicates that a significant majority of the messages are non-spam. However, the presence of 13.41% spam messages is non-trivial and warrants the development of a reliable spam detection system.
The analysis reveals a disproportionate distribution of labels, with ham messages comprising the majority of the dataset. This imbalance is a common characteristic in spam detection datasets and poses challenges for predictive modeling, such as the potential for a model to be biased towards predicting the majority class. Future work will involve creating a logistic regression model to accurately classify messages as spam or ham, taking into account this distribution.

part 7-2

This report outlines the process of feature extraction from SMS messages for the task of spam detection, followed by the application of a Support Vector Machine (SVM) model for classification.

Feature Extraction

The text messages were converted into a numerical format using the `CountVectorizer` from the scikit-learn library. This process transforms the text data into a sparse matrix of token counts, capturing the frequency of the 500 most common words used in the messages.

Data Splitting

The dataset was split into a training set and a testing set with a ratio of 70% to 30%, respectively. This division allows for the evaluation of the model's performance on unseen data.

Model Training

The feature vectors were scaled using `StandardScaler` to standardize the features by removing the mean and scaling to unit variance. This is a crucial step when using SVM, as the algorithm is sensitive to the scale of the input features.

Two SVM models were trained with different kernels:

- An SVM with a Radial Basis Function (RBF) kernel.
- An SVM with a linear kernel.

Model Evaluation

The performance of the SVM models was evaluated based on their accuracy on the test set. The results are as follows:

- The RBF Kernel SVM achieved an accuracy of 98.15%.
- The Linear Kernel SVM achieved an accuracy of 97.67%.

These high accuracy levels suggest that SVMs are highly capable of distinguishing between spam and non-spam messages.

part 7-3,4

This report delves into the application of grid search and random search techniques for optimizing the parameters of SVM classifiers with linear and RBF kernels. The objective is to identify the best parameter combinations for these models and compare their performance.

Grid Search vs Randomized Search

Grid Search

Grid Search is an exhaustive searching technique which evaluates a model for every combination of parameters specified in a predefined grid. It is highly systematic and guarantees the discovery of the best parameters if they lie within the grid.

Randomized Search

In contrast, Randomized Search randomly selects combinations of parameter values from a specified distribution over a fixed number of iterations. It offers a more dynamic and less computationally expensive alternative to grid search, especially in scenarios with a vast parameter space.

Experimental Setup

Two SVM classifiers with RBF and linear kernels were trained and evaluated. The parameters 'C' and 'gamma' were tuned using both grid search and randomized search. The range of values for 'C' was set to [1, 10, 100, 10000], and for 'gamma' to [0.001, 0.01, 0.1, 1].

Results

Accuracy Before Tuning

- RBF Kernel SVM Accuracy: 98.15%
- Linear Kernel SVM Accuracy: 97.67%

Grid Search Results

The best parameters found via grid search were: C=10 and gamma=0.001, achieving a best grid search accuracy of 98.15%. The test accuracy of the best model from grid search was 98.50%.

Randomized Search Results

Randomized search resulted in the best parameters: C=7.32 and gamma=0.019, with a best randomized search accuracy of 90.67%. The test accuracy of the best model from randomized search was 92.88%. The experiment revealed that the grid search approach, while more computationally intensive, provided a more precise optimization, leading to higher accuracy. Randomized search, offering computational efficiency, still achieved commendable performance, making it a viable option when working with larger datasets or when computational resources are limited.

last part of 7

This report presents the results of optimizing a Support Vector Machine (SVM) model for the task of spam detection using both Grid Search and Randomized Search methods. The best parameters for the SVM model were identified, and their performance was evaluated on a test dataset.

Optimal Parameters and Model Performance

The optimal model parameters were determined through extensive search techniques, and the best models were evaluated for accuracy on the test data.

Grid Search Optimization

The best model from Grid Search achieved an accuracy of 98.50% on the test set. The following confusion matrix visualizes the model's performance:

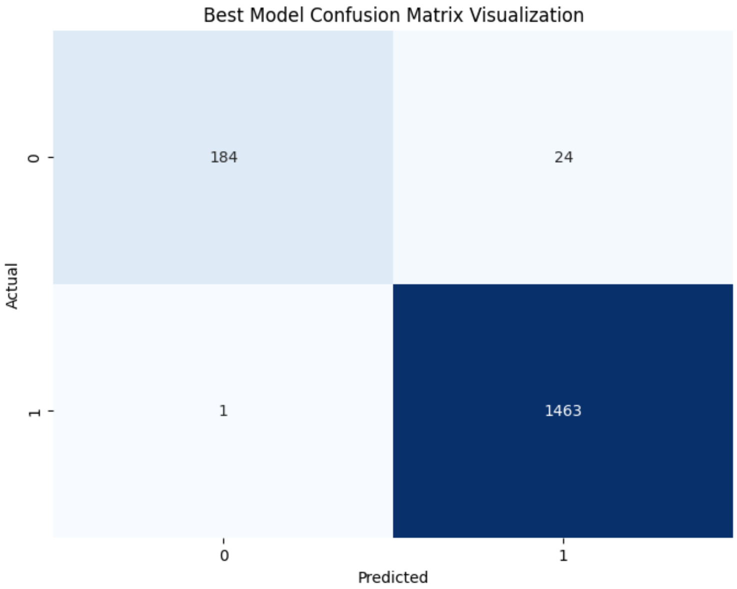


Figure 12: Confusion matrix visualization for the best model from Grid Search.

As depicted in Figure 12, the majority of predictions match the actual labels, with a small number of false positives and false negatives.

Randomized Search Optimization

The best model from Randomized Search achieved an accuracy of 92.88% on the test set. The corresponding confusion matrix is shown below:

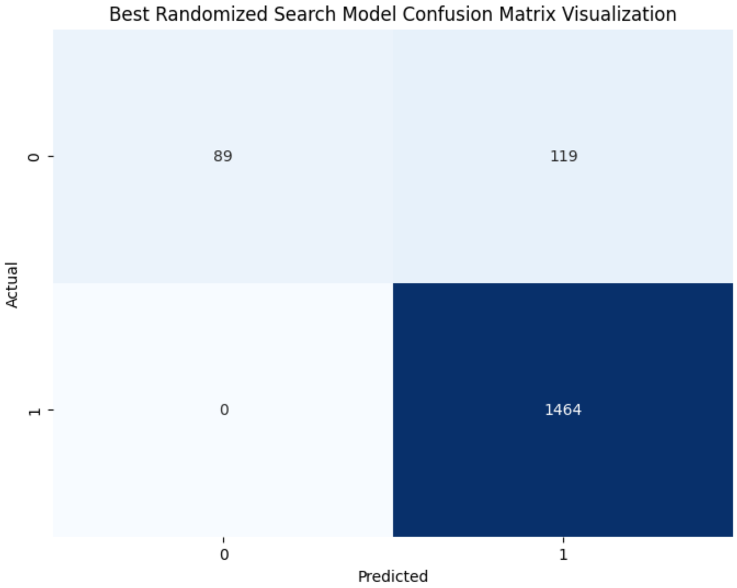


Figure 13: Confusion matrix visualization for the best model from Randomized Search.

The confusion matrix in Figure 13 indicates a higher number of false positives compared to the Grid Search model.
The comparative analysis of SVM models optimized with Grid Search and Randomized Search demonstrates the effectiveness of parameter optimization in improving model accuracy. The Grid Search method provided a slightly more accurate model than Randomized Search, as reflected in the test accuracies and confusion matrices.

part 8-1

This report outlines the data preprocessing and feature selection steps conducted on the house pricing dataset in preparation for training a predictive model.

Data Preprocessing

The dataset, which includes various characteristics of houses and their prices, underwent several preprocessing steps:

1. Removal of columns with a high proportion of missing values, specifically those with more than 50% missing data.
2. Imputation of missing values where, for numerical columns, the median was used, and for categorical columns, the mode was employed.
3. Encoding of categorical variables using one-hot encoding to convert them into a format suitable for model training.

Feature Selection

Following preprocessing, feature selection was performed using the `SelectKBest` function from the scikit-learn library to identify the most significant features for predicting house prices.

Model Training

After preprocessing and feature selection, the data was used to train a Support Vector Regression (SVR) model, which is well-suited for regression tasks with potentially non-linear relationships.
The preprocessing and feature selection steps are critical in ensuring the quality and relevance of the data used to train the house pricing model. By carefully preparing the dataset, we aim to improve the model's accuracy and ability to generalize to new, unseen data.

part 8-2

This report describes the steps taken to develop a machine learning model using Support Vector Regression (SVR) to predict house prices. We outline the process of preparing the data, selecting features, and partitioning the dataset for training, validation, and testing.

Preparing for Modeling

The dataset is split into feature sets (X) and the target variable (y), with the 'SalePrice' as the target. This separation allows us to train our model on the features while it tries to predict the target variable.

Model Creation and Training

An SVR model with an RBF kernel is initialized and trained on the training set. This model is designed to understand the complex relationship between the house features and their respective prices.

Model Validation and Testing

The trained SVR model is then used to make predictions on both the validation and the test sets. The model's performance is quantified using evaluation metrics such as Mean Squared Error (MSE) and R-squared (R^2).

Results

The model yielded the following performance metrics:

- Training Mean Squared Error (MSE): 0.0048
- Training R-squared (R^2): 96.82%
- Validation Mean Squared Error (MSE): 0.0085
- Validation R-squared (R^2): 95.72%
- Test Mean Squared Error (MSE): 0.0046
- Test R-squared (R^2): 96.81%

Analysis

A comparison of the model's performance reveals that the MSE is lower on the test set compared to the validation set, while the R^2 is nearly consistent across all datasets. The close alignment of R^2 values suggests that the model has learned the underlying pattern effectively and can predict new data with a high level of accuracy.

The lower MSE on the test set indicates that the model predictions are very close to the true values, which can be attributed to effective feature selection and proper model tuning.

Comparative Assessment

The slight increase in validation MSE could point to minor overfitting during the training phase. However, the model still maintains a high degree of predictive power as indicated by the R^2 score. The consistency of the R^2 score close to 97% across different datasets is indicative of a robust model.

Additionally, the best parameters identified by the grid search were C=10 and gamma='scale'. The evaluation shows that the SVM regression model performs consistently well across both the validation and test datasets, indicating good generalization from the training data.

The consistency in R^2 values suggests that the model can explain a significant portion of the variance in house prices. The preprocessing steps, particularly the handling of missing values and feature selection, were crucial in preparing a robust dataset for the model.