Project Statistical Methods for Decision Making: Austo Automobile Customers Data Analysis

Context

Analysts are required to explore data and reflect on the insights. Clear writing skill is an integral part of a good report. Note that the explanations must be such that readers with minimum knowledge of analytics is able to grasp the insight.

Austo Motor Company is a leading car manufacturer specializing in SUV, Sedan, and Hatchback models. In its recent board meeting, concerns were raised by the members on the efficiency of the marketing campaign currently being used. The board decides to rope in an analytics professional to improve the existing campaign.

Objective

They want to analyze the data to get a fair idea about the demand of customers which will help them in enhancing their customer experience. Suppose you are a Data Scientist at the company and the Data Science team has shared some of the key questions that need to be answered. Perform the data analysis to find answers to these questions that will help the company to improve the business.

Data Description

The data contains the different data related to Austo Automobile customers. The detailed data dictionary is given below.

Data Dictionary

- age: The age of the individual in years.
- gender: The gender of the individual, categorized as male or female.
- profession: The occupation or profession of the individual.
- marital_status: The marital status of the individual, such as married &, single
- education: The educational qualification of the individual Graduate and Post Graduate
- no_of_dependents: The number of dependents (e.g., children, elderly parents) that the individual supports financially.
- personal_loan: A binary variable indicating whether the individual has taken a personal loan "Yes" or "No"
- house_loan: A binary variable indicating whether the individual has taken a housing loan "Yes" or "No"

- partner_working: A binary variable indicating whether the individual's partner is employed "Yes" or "No"
- salary: The individual's salary or income.
- partner_salary: The salary or income of the individual's partner, if applicable.
- Total_salary: The total combined salary of the individual and their partner (if applicable).
- price: The price of a product or service.
- make: The type of automobile

Importing required libraries

In [96]:
```
# Import libraries for data manipulation
import numpy as np
import pandas as pd

# Import libraries for data visualization
import matplotlib.pyplot as plt
import seaborn as sns
```

Understanding the structure of data

In [97]:
```
df = pd.read_csv('austo_automobile.csv')

df.head() # Returns first 5 rows
```

Out[97]:

	Age	Gender	Profession	Marital_status	Education	No_of_Dependents	Personal_loan
0	53	Male	Business	Married	Post Graduate	4	No
1	53	Femal	Salaried	Married	Post Graduate	4	Yes
2	53	Female	Salaried	Married	Post Graduate	3	No
3	53	Female	Salaried	Married	Graduate	2	Yes
4	53	Male	Salaried	Married	Post Graduate	3	No

Number of rows and columns in the dataset

In [98]:
```
df.shape # Shape of the dataset
```

Out[98]:
```
(1581, 14)
```

Datatypes of the different columns in the dataset

In [99]:
```
df.info() # Concise summary of dataset
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1581 entries, 0 to 1580
Data columns (total 14 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   Age               1581 non-null   int64  
 1   Gender            1528 non-null   object 
 2   Profession        1581 non-null   object 
 3   Marital_status    1581 non-null   object 
 4   Education         1581 non-null   object 
 5   No_of_Dependents  1581 non-null   int64  
 6   Personal_loan     1581 non-null   object 
 7   House_loan        1581 non-null   object 
 8   Partner_working   1581 non-null   object 
 9   Salary            1581 non-null   int64  
 10  Partner_salary    1475 non-null   float64
 11  Total_salary      1581 non-null   int64  
 12  Price             1581 non-null   int64  
 13  Make              1581 non-null   object 
dtypes: float64(1), int64(5), object(8)
memory usage: 173.0+ KB
```

It seems Salary, Total_salary and Price columns have datatype as int64 in the dataset. These columns can contain decimal values hence changing them to float64 datatype as similar to Partner_salary column datatype.

In [100…

```
df.Salary = df.Salary.astype(float) # Change datatype to float
df.Total_salary = df.Total_salary.astype(float) # Change datatype to float
df.Price = df.Price.astype(float) # Change datatype to float

df.info() # Concise summary of dataset
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1581 entries, 0 to 1580
Data columns (total 14 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   Age               1581 non-null   int64  
 1   Gender            1528 non-null   object 
 2   Profession        1581 non-null   object 
 3   Marital_status    1581 non-null   object 
 4   Education         1581 non-null   object 
 5   No_of_Dependents  1581 non-null   int64  
 6   Personal_loan     1581 non-null   object 
 7   House_loan        1581 non-null   object 
 8   Partner_working   1581 non-null   object 
 9   Salary            1581 non-null   float64
 10  Partner_salary    1475 non-null   float64
 11  Total_salary      1581 non-null   float64
 12  Price             1581 non-null   float64
 13  Make              1581 non-null   object 
dtypes: float64(4), int64(2), object(8)
memory usage: 173.0+ KB
```

Finding missing values in the dataset

In [101…
```
df.isna().sum()  # Count NaN values in all columns of dataset
```
Out[101…
```
Age                  0
Gender              53
Profession           0
Marital_status       0
Education            0
No_of_Dependents     0
Personal_loan        0
House_loan           0
Partner_working      0
Salary               0
Partner_salary     106
Total_salary         0
Price                0
Make                 0
dtype: int64
```
Gender column has 53 missing values and Partner_salary column has 106 missing values.

Treating missing values in the dataset

1. Gender (datatype - object):

   Filling missing values with Mode as distribution is skewed.

   Replaced Femal/Femle value with correct value - Female.

In [102…
```
df['Gender'].value_counts() # Frequency of each distinct value in the Gender column
```
Out[102…
```
Gender
Male      1199
Female     327
Femal        1
Femle        1
Name: count, dtype: int64
```
In [103…
```
df['Gender'].fillna(df['Gender'].mode()[0], inplace = True) # Replace NaN values wi
df['Gender'].replace('Femal', 'Female', inplace=True) # Replace Femal value with Fe
df['Gender'].replace('Femle', 'Female', inplace=True) # Replace Femle value with Fe
```
In [104…
```
df['Gender'].value_counts() # Frequency of each distinct value in the Gender column
```
Out[104…
```
Gender
Male      1252
Female     329
Name: count, dtype: int64
```
In [105…
```
df.isna().sum()  # Count NaN values in all columns of dataset
```

```
Out[105…  Age                   0
          Gender                0
          Profession            0
          Marital_status        0
          Education             0
          No_of_Dependents      0
          Personal_loan         0
          House_loan            0
          Partner_working       0
          Salary                0
          Partner_salary      106
          Total_salary          0
          Price                 0
          Make                  0
          dtype: int64
```

2. Partner_salary (datatype - float64):

Filling missing values using formula: Partner_salary = Total_salary - Salary

```
In [106…  df['Partner_salary'].fillna(df['Total_salary'] - df['Salary'], inplace = True) # Re
```

```
In [107…  df.isna().sum()  # Count NaN values in all columns of dataset
```

```
Out[107…  Age                 0
          Gender              0
          Profession          0
          Marital_status      0
          Education           0
          No_of_Dependents    0
          Personal_loan       0
          House_loan          0
          Partner_working     0
          Salary              0
          Partner_salary      0
          Total_salary        0
          Price               0
          Make                0
          dtype: int64
```

We can see from above list that there is no NaN value in the Gender and Partner_salary columns

Statistical summary of the data

```
In [108…  df.describe(include='all').T # Summary statistics of the numerical and categorial d
```

Out[108…

	count	unique	top	freq	mean	std	min
Age	1581.0	NaN	NaN	NaN	31.922201	8.425978	22.0
Gender	1581	2	Male	1252	NaN	NaN	NaN
Profession	1581	2	Salaried	896	NaN	NaN	NaN
Marital_status	1581	2	Married	1443	NaN	NaN	NaN
Education	1581	2	Post Graduate	985	NaN	NaN	NaN
No_of_Dependents	1581.0	NaN	NaN	NaN	2.457938	0.943483	0.0
Personal_loan	1581	2	Yes	792	NaN	NaN	NaN
House_loan	1581	2	No	1054	NaN	NaN	NaN
Partner_working	1581	2	Yes	868	NaN	NaN	NaN
Salary	1581.0	NaN	NaN	NaN	60392.220114	14674.825044	30000.0 5
Partner_salary	1581.0	NaN	NaN	NaN	19233.776091	19670.391171	0.0
Total_salary	1581.0	NaN	NaN	NaN	79625.996205	25545.857768	30000.0 6
Price	1581.0	NaN	NaN	NaN	35597.72296	13633.636545	18000.0 2
Make	1581	3	Sedan	702	NaN	NaN	NaN

Observations and Insights:

1. Salary of customer increases when age increases.
2. Most customers are males.
3. Most customers are married.
4. Maximum number of dependents are 4 in a family.
5. Total Salary of customer increases when they have working partner.

Exploratory Data Analysis (EDA)

Univariate Analysis

Gender

In [109…

```
# Check unique Gender
df['Gender'].value_counts() # Frequency of each distinct value in the Gender column
```

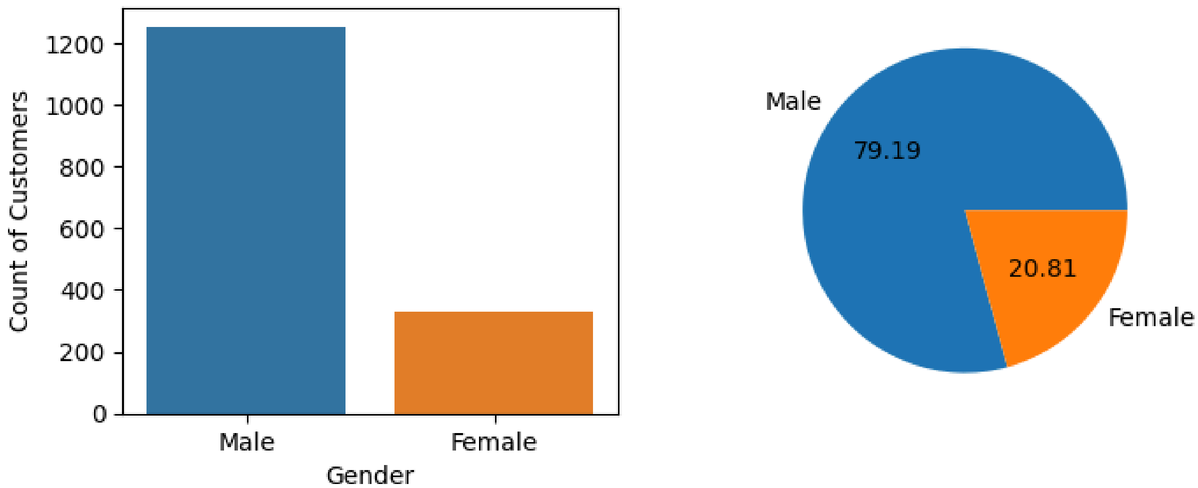
Out[109…

```
Gender
Male      1252
Female     329
Name: count, dtype: int64
```

In [110…

```
# Count Plot and Pie Chart - Distribution of Gender across customers

fig, ax = plt.subplots(1,2, figsize=(8,3))
sns.countplot(data=df, x='Gender', order = df['Gender'].value_counts().index, ax=ax
ax[0].set(xlabel = 'Gender', ylabel = 'Count of Customers')
ax[1]=plt.pie(df['Gender'].value_counts(), labels=['Male', 'Female'], autopct='%.2f
fig.suptitle('Fig 1: Distribution of Gender Across Customers')
plt.show()
```

Fig 1: Distribution of Gender Across Customers



Profession

In [111…

```
# check unique Profession
df['Profession'].value_counts() # Frequency of each distinct value in the Professio
```

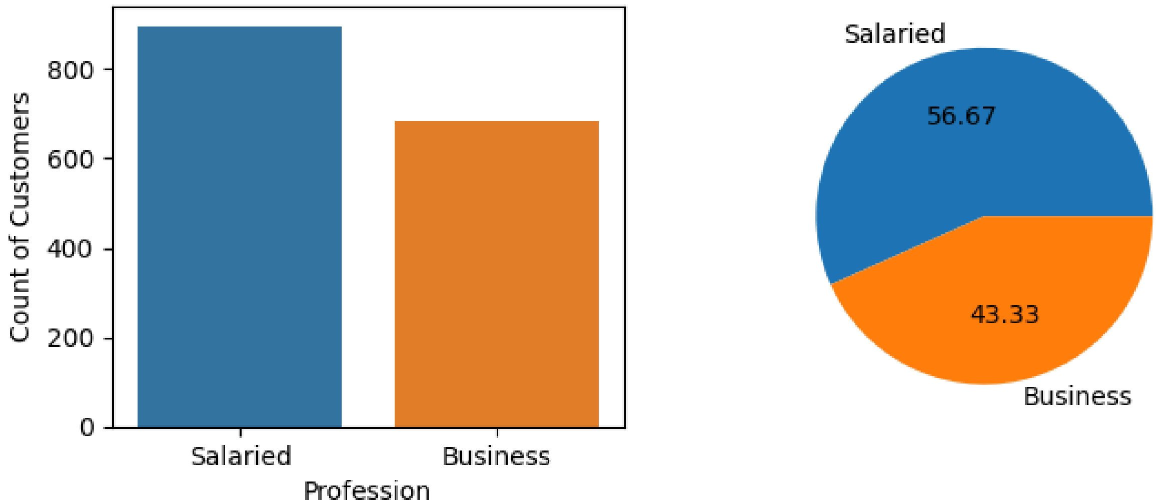
Out[111…

```
Profession
Salaried    896
Business    685
Name: count, dtype: int64
```

In [112…

```
# Count Plot and Pie Chart - Distribution of Profession across customers

fig, ax = plt.subplots(1,2, figsize=(8,3))
sns.countplot(data=df, x='Profession', order = df['Profession'].value_counts().inde
ax[0].set(xlabel = 'Profession', ylabel = 'Count of Customers')
ax[1]=plt.pie(df['Profession'].value_counts(), labels=['Salaried', 'Business'], aut
fig.suptitle('Fig 2: Distribution of Profession Across Customers')
plt.show()
```

Fig 2: Distribution of Profession Across Customers



Marital Status

In [113…

```
# check unique Marital Status
df['Marital_status'].value_counts() # Frequency of each distinct value in the Marit
```

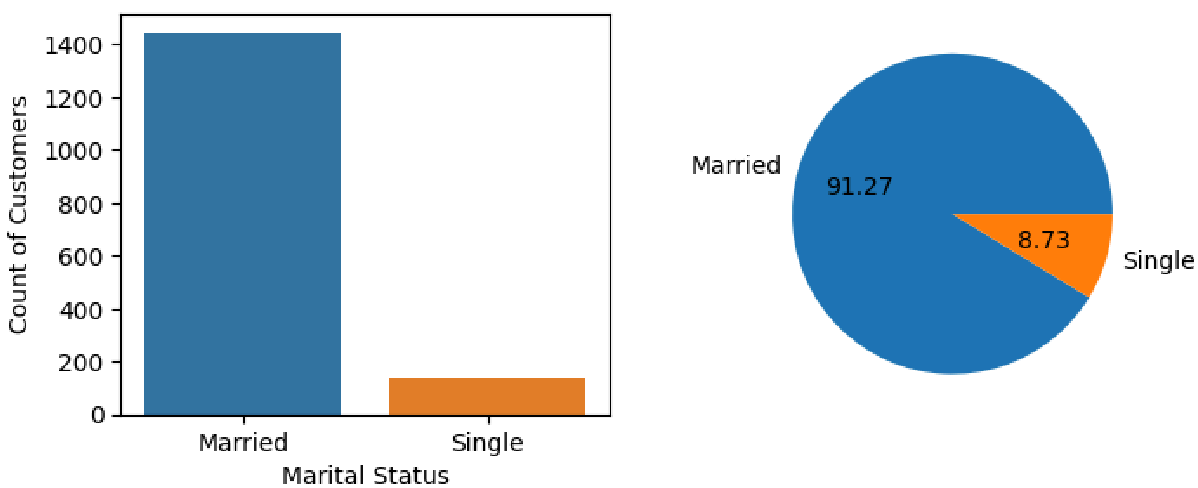
Out[113…

```
Marital_status
Married    1443
Single      138
Name: count, dtype: int64
```

In [114…

```
# Count Plot and Pie Chart - Distribution of Marital Status across customers

fig, ax = plt.subplots(1,2, figsize=(8,3))
sns.countplot(data=df, x='Marital_status', order = df['Marital_status'].value_count
ax[0].set(xlabel = 'Marital Status', ylabel = 'Count of Customers')
ax[1]=plt.pie(df['Marital_status'].value_counts(), labels=['Married', 'Single'], au
fig.suptitle('Fig 3: Distribution of Marital Status Across Customers')
plt.show()
```

Fig 3: Distribution of Marital Status Across Customers



Education

In [115…

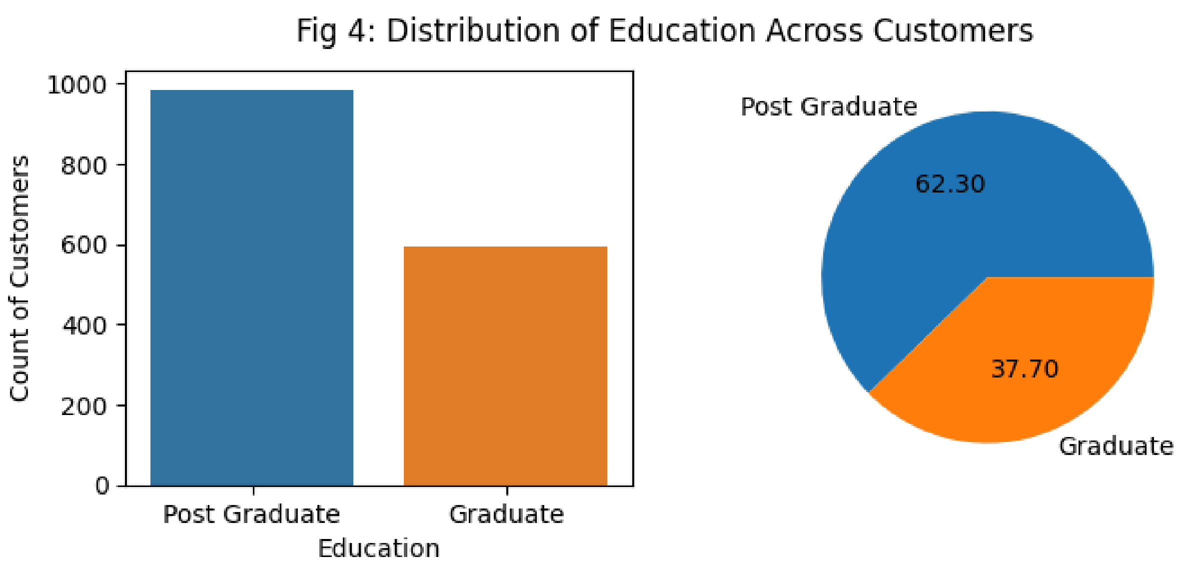
```
# Check unique Education
df['Education'].value_counts() # Frequency of each distinct value in the Education
```

Out[115…

```
Education
Post Graduate    985
Graduate         596
Name: count, dtype: int64
```

In [116…

```
# Count Plot and Pie Chart - Distribution of Education across customers

fig, ax = plt.subplots(1,2, figsize=(8,3))
sns.countplot(data=df, x='Education', order = df['Education'].value_counts().index,
ax[0].set(xlabel = 'Education', ylabel = 'Count of Customers')
ax[1]=plt.pie(df['Education'].value_counts(), labels=['Post Graduate', 'Graduate'],
fig.suptitle('Fig 4: Distribution of Education Across Customers')
plt.show()
```



Personal Loan

In [117…

```
# Check unique Personal Loan
df['Personal_loan'].value_counts() # Frequency of each distinct value in the Person
```

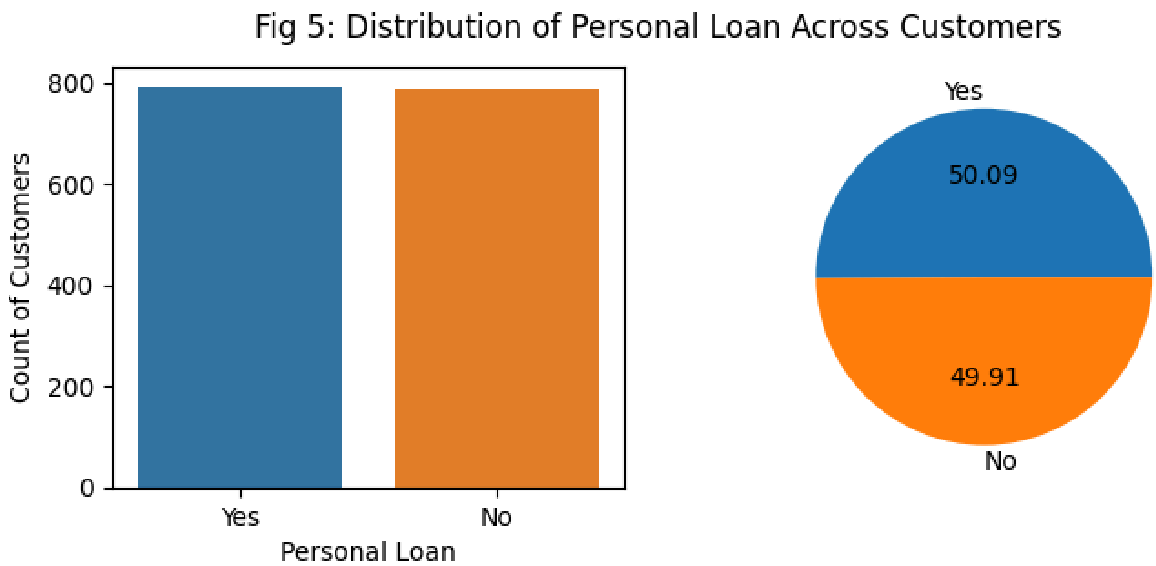
Out[117…

```
Personal_loan
Yes    792
No     789
Name: count, dtype: int64
```

In [118…

```
# Count Plot and Pie Chart - Distribution of Personal Loan across customers

fig, ax = plt.subplots(1,2, figsize=(8,3))
sns.countplot(data=df, x='Personal_loan', order = df['Personal_loan'].value_counts(
ax[0].set(xlabel = 'Personal Loan', ylabel = 'Count of Customers')
ax[1]=plt.pie(df['Personal_loan'].value_counts(), labels=['Yes', 'No'], autopct='%.
```

```
fig.suptitle('Fig 5: Distribution of Personal Loan Across Customers')
plt.show()
```



House Loan

In [119…

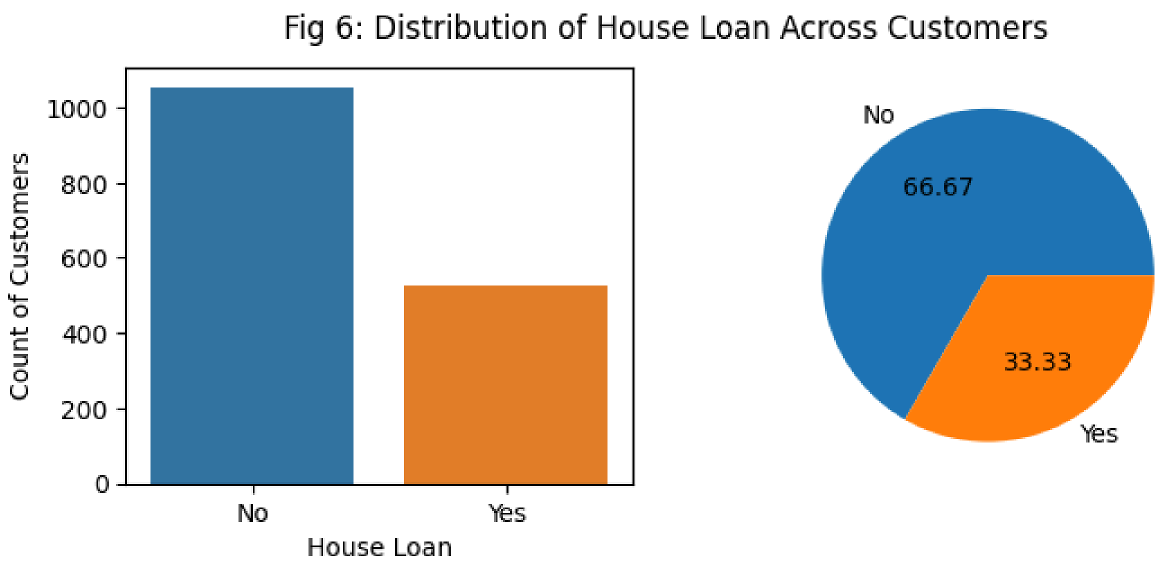
```
# Check unique House loan
df['House_loan'].value_counts() # Frequency of each distinct value in the House_loa
```

Out[119…

```
House_loan
No     1054
Yes     527
Name: count, dtype: int64
```

In [120…

```
# Count Plot and Pie Chart - Distribution of House Loan across customers

fig, ax = plt.subplots(1,2, figsize=(8,3))
sns.countplot(data=df, x='House_loan', order = df['House_loan'].value_counts().inde
ax[0].set(xlabel = 'House Loan', ylabel = 'Count of Customers')
ax[1]=plt.pie(df['House_loan'].value_counts(), labels=['No', 'Yes'], autopct='%.2f'
fig.suptitle('Fig 6: Distribution of House Loan Across Customers')
plt.show()
```



Partner Working

In [121…

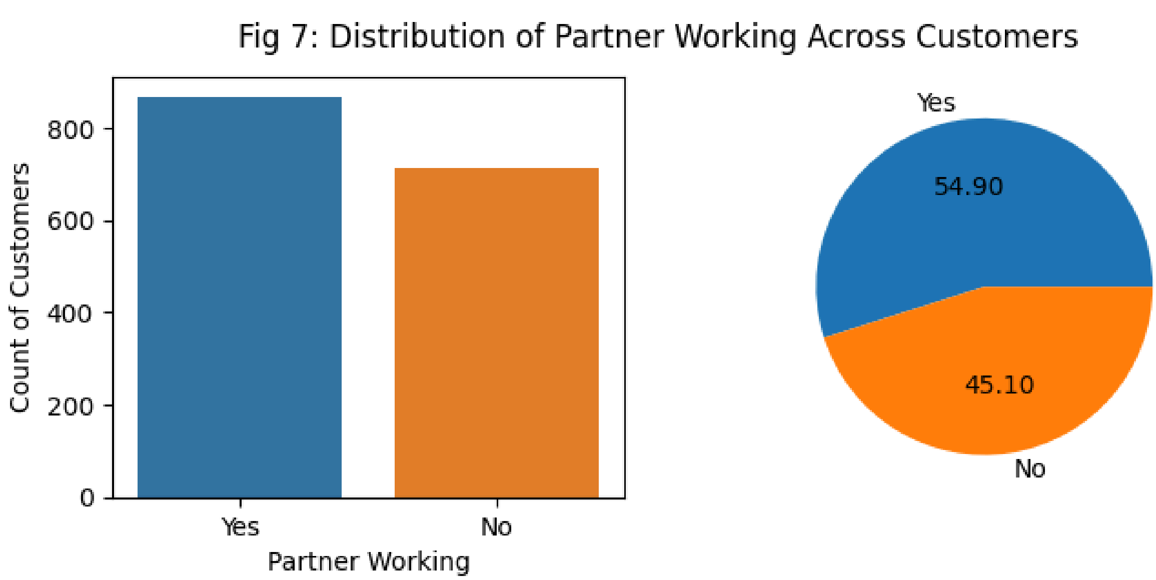
```
# Check unique Partner Working
df['Partner_working'].value_counts() # Frequency of each distinct value in the Part
```

Out[121…

```
Partner_working
Yes    868
No     713
Name: count, dtype: int64
```

In [122…

```
# Count Plot and Pie Chart - Distribution of Partner Working across customers

fig, ax = plt.subplots(1,2, figsize=(8,3))
sns.countplot(data=df, x='Partner_working', order = df['Partner_working'].value_cou
ax[0].set(xlabel = 'Partner Working', ylabel = 'Count of Customers')
ax[1]=plt.pie(df['Partner_working'].value_counts(), labels=['Yes', 'No'], autopct='
fig.suptitle('Fig 7: Distribution of Partner Working Across Customers')
plt.show()
```



Make

In [123…

```
# Check unique Make
df['Make'].value_counts() # Frequency of each distinct value in the Make column
```

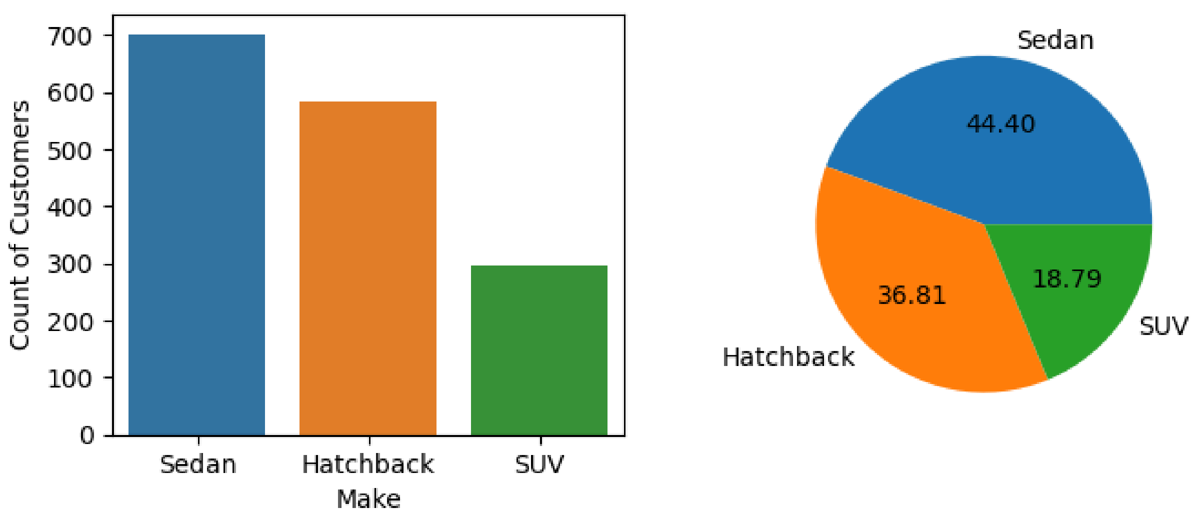
Out[123…

```
Make
Sedan        702
Hatchback    582
SUV          297
Name: count, dtype: int64
```

In [124…

```
# Count Plot and Pie Chart - Distribution of Make across customers

fig, ax = plt.subplots(1,2, figsize=(8,3))
sns.countplot(data=df, x='Make', order = df['Make'].value_counts().index, ax=ax[0])
ax[0].set(xlabel = 'Make', ylabel = 'Count of Customers')
ax[1]=plt.pie(df['Make'].value_counts(), labels=['Sedan', 'Hatchback', 'SUV'], autc
fig.suptitle('Fig 8: Distribution of Make Across Customers')
plt.show()
```

Fig 8: Distribution of Make Across Customers



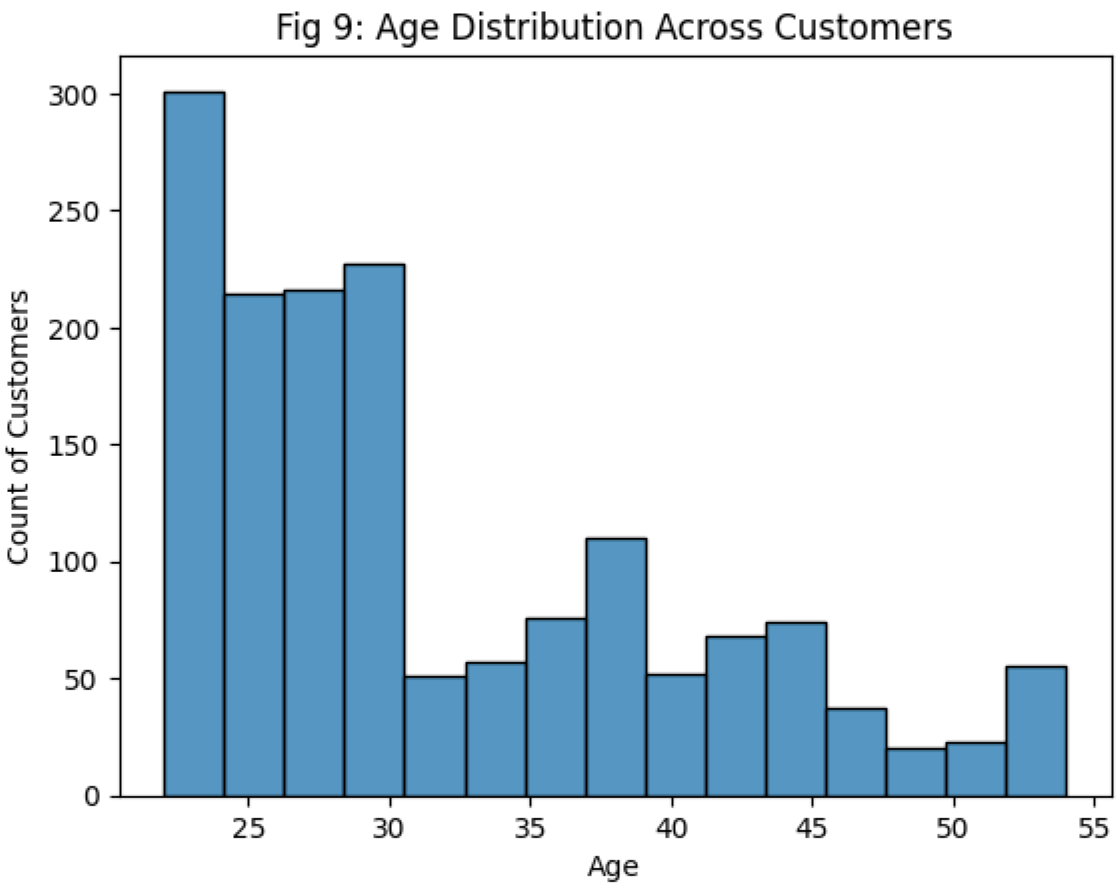
Observations and Insights:

1. There are more male customers than female customers.
2. There are more salaried class customers than business class customers.
3. There are more married customers than single customers.
4. There are more Post Graduate customers than Graduate customers.
5. Customers who have taken personal loan are almost equal to the customers who have not taken personal loan.
6. Customers who have taken housing loan are less than the customers who have not taken housing loan.
7. Customers with working partner are more than the customers who do not have a working partner.
8. Customers have a greater number of Sedan cars than Hatchback and SUV cars.

Age

In [125…

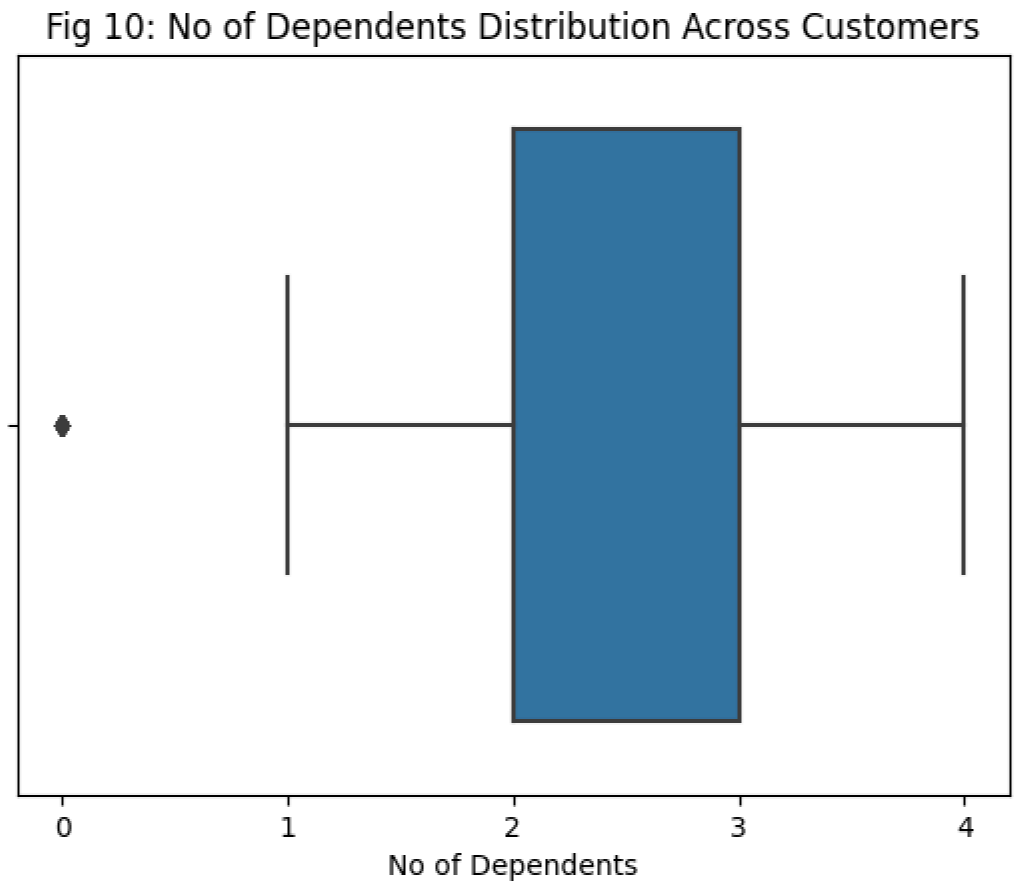
```
# Histogram for the Age distribution

sns.histplot(data=df,x='Age')
plt.title('Fig 9: Age Distribution Across Customers')
plt.ylabel('Count of Customers')
plt.show()
```



No of Dependents

In [126…

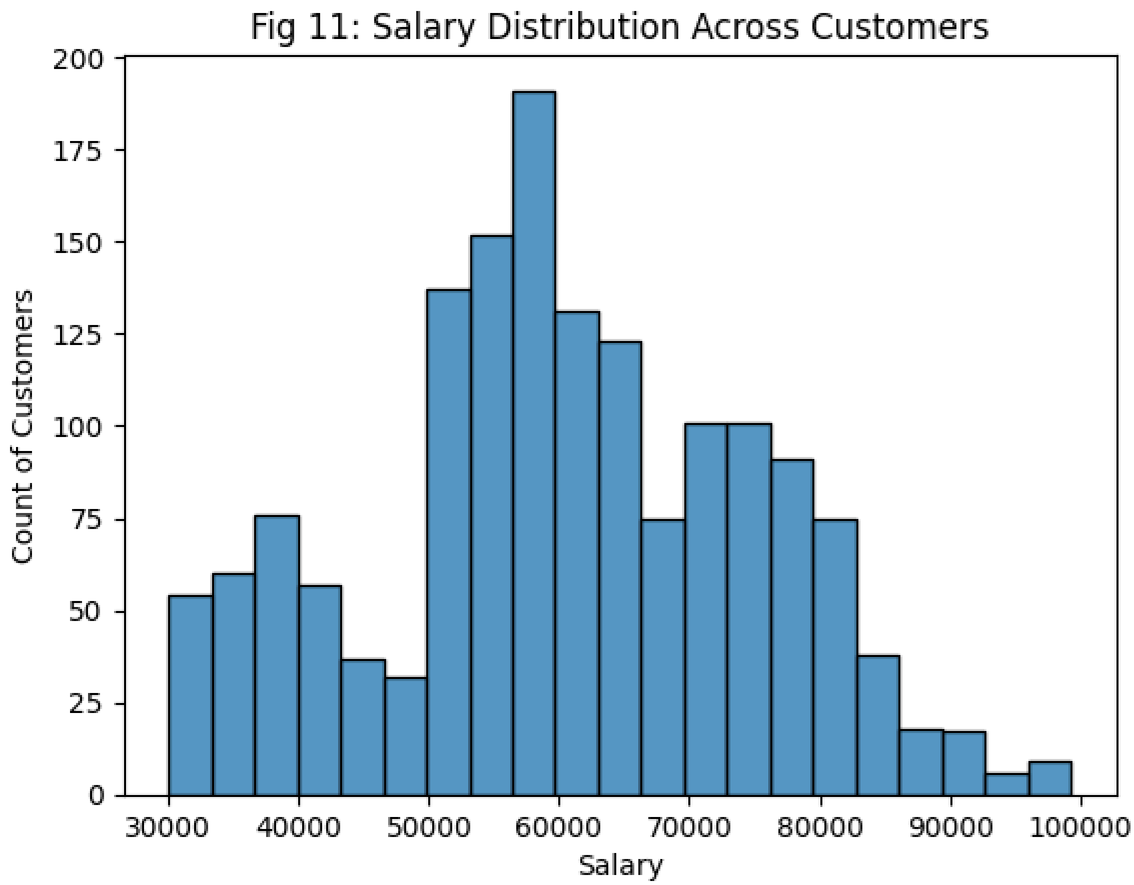
```
# Box Plot for the No of Dependents distribution

sns.boxplot(data=df,x='No_of_Dependents')
plt.title('Fig 10: No of Dependents Distribution Across Customers')
plt.xlabel('No of Dependents')
plt.xticks([0,1,2,3,4])
plt.show()
```



Salary

In [127…

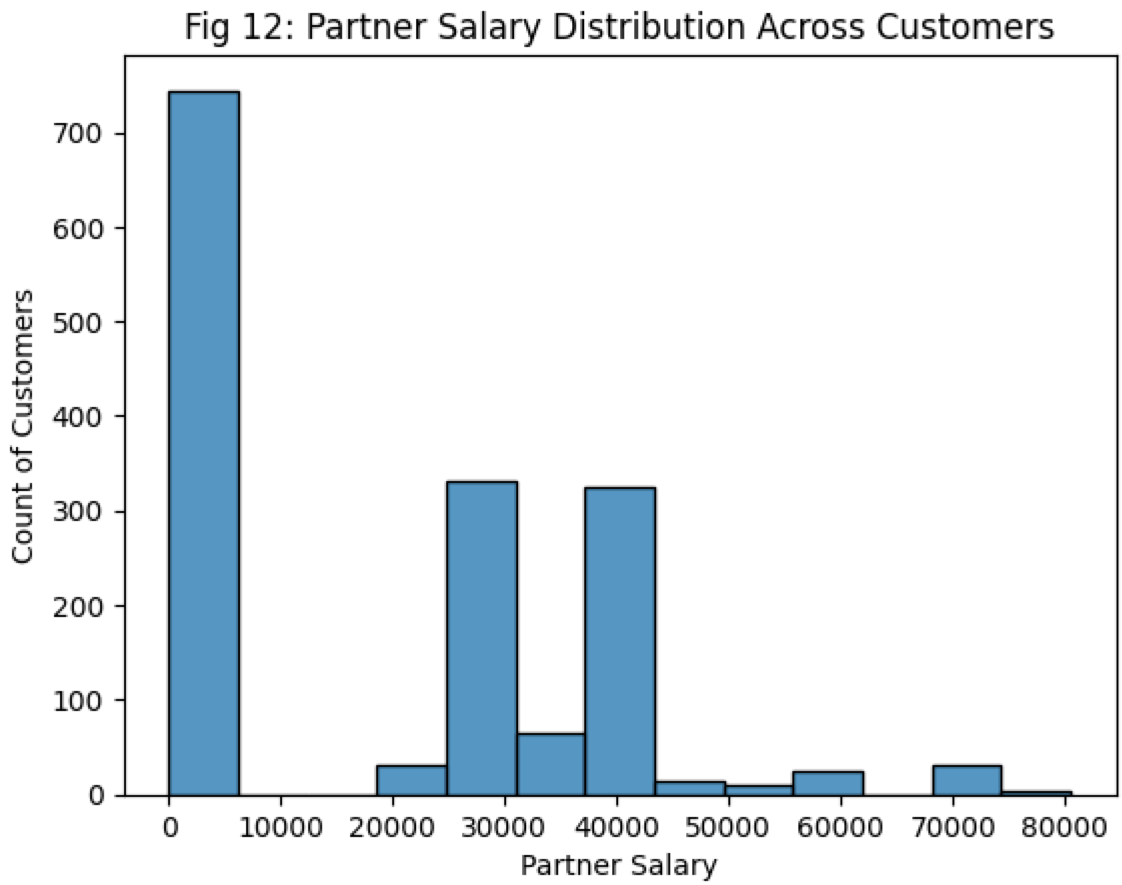
```
# Histogram for the Salary distribution

sns.histplot(data=df,x='Salary')
plt.title('Fig 11: Salary Distribution Across Customers')
plt.ylabel('Count of Customers')
plt.show()
```



Partner Salary

In [128…

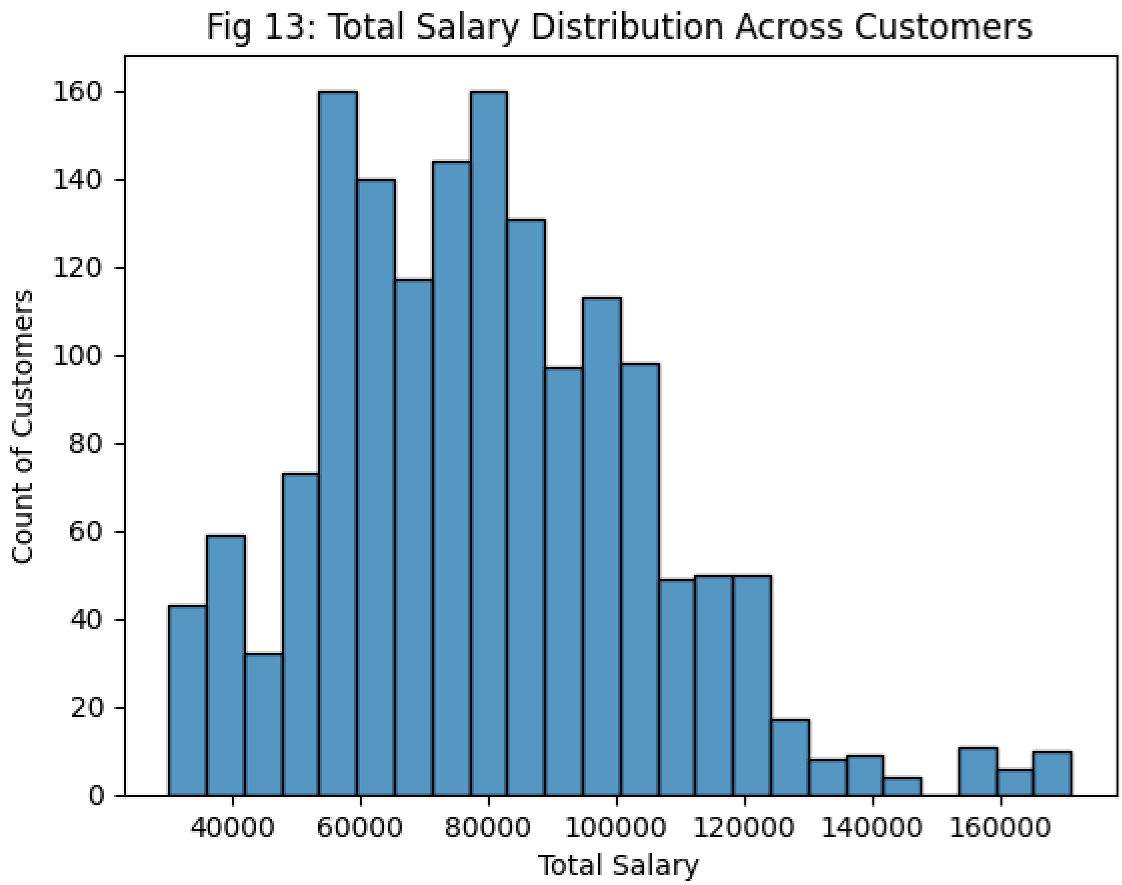
```
# Histogram for the Partner Salary distribution

sns.histplot(data=df,x='Partner_salary')
plt.title('Fig 12: Partner Salary Distribution Across Customers')
plt.xlabel('Partner Salary')
plt.ylabel('Count of Customers')
plt.show()
```



Total Salary

In [129…

```
# Histogram for the Total Salary distribution

sns.histplot(data=df,x='Total_salary')
plt.title('Fig 13: Total Salary Distribution Across Customers')
plt.xlabel('Total Salary')
plt.ylabel('Count of Customers')
plt.show()
```



In [130…

```
# Box Plot for the Total Salary distribution

sns.boxplot(data=df,x='Total_salary')
plt.title('Fig 14: Total Salary Distribution Across Customers')
plt.xlabel('Total Salary')
plt.show()
```

Price

In [131…

```
# Histogram for the Price distribution

sns.histplot(data=df,x='Price')
plt.title('Fig 15: Price Distribution Across Customers')
plt.ylabel('Count of Customers')
plt.show()
```

Observations and Insights:

1. Age distribution is Positively Skewed i.e. mean age of customer is more than the mode age of customer. Minimum age of customer is 22 years and maximum age of customer is 54 years.
2. Number of dependents in a family are evenly distributed (symmetric) - if count is more than 0. It has one outlier with count as 0 which is less than 0.5 (Q1 (2) - 1.5 * IQR (1)). Minimum number of dependents in a family is 0 and maximum number of dependents in a family are 4.
3. Salary distribution is Positively Skewed i.e. mean salary of customer is more than the mode salary of customer. Minimum salary of customer is 30000.0 USD and maximum salary of customer is 99300.0 USD.
4. Partner Salary distribution is Positively Skewed i.e. mean partner salary of customer is more than the mode partner salary of customer. Minimum partner salary of customer is 0.0 USD and maximum partner salary of customer is 80500.0 USD.
5. Total Salary distribution is Positively Skewed i.e. mean total salary of customer is more than the mode total salary of customer. It has outliers i.e. total salary of some customers is more than 149000.0 USD (Q3 (95900.0) + 1.5 * IQR (35400.0)). Minimum total salary of customer is 30000.0 USD and maximum total salary of customer is 171000.0 USD.
6. Price distribution is Positively Skewed i.e. mean price of car is more than the mode price of car. Minimum price of car is 18000.0 USD and maximum price of car is 70000.0 USD.

Bivariate Analysis

Correlation among variables

In [132…

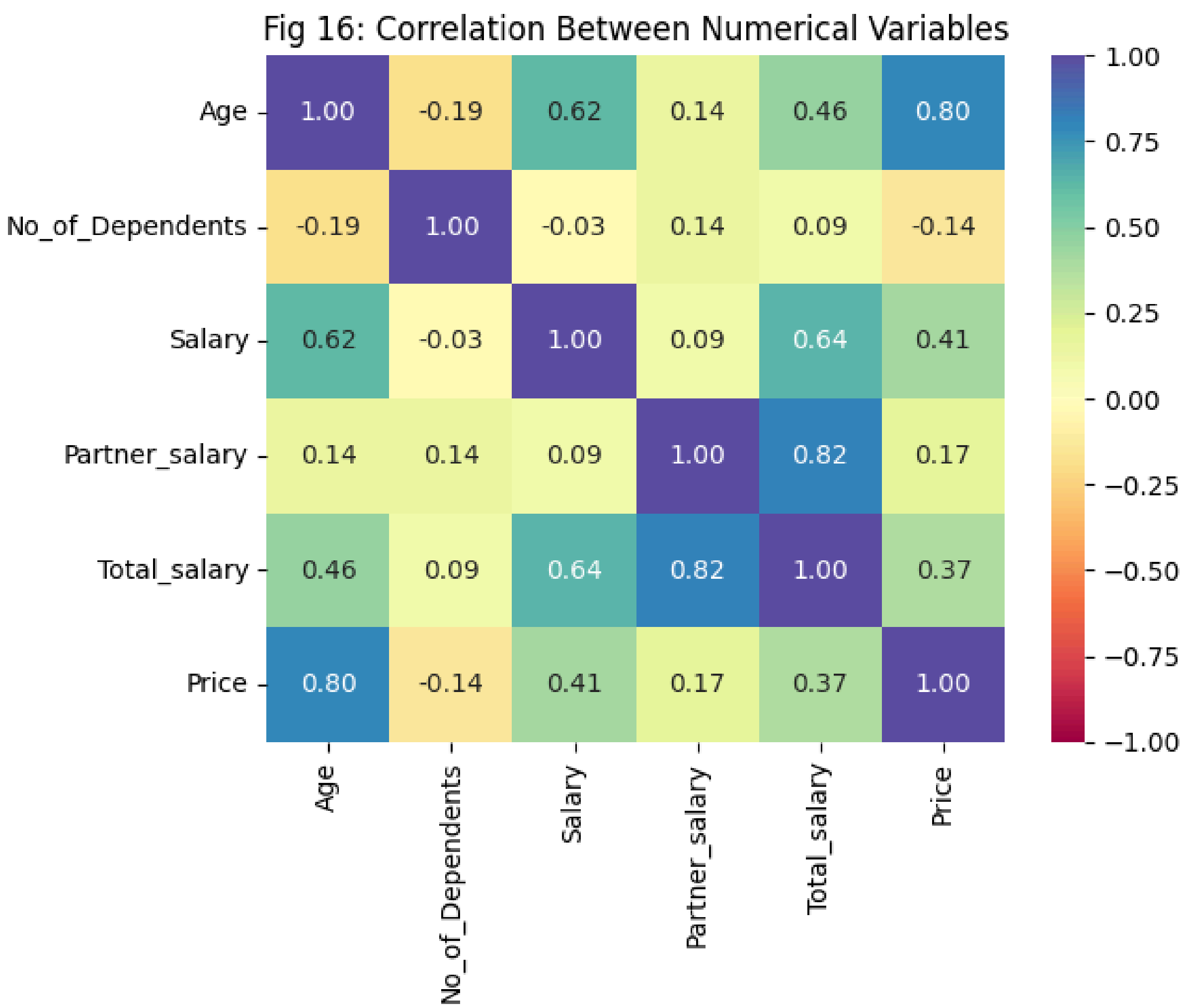
```
# orrelation between all numerical variables in the dataset

col_list = ['Age','No_of_Dependents','Salary','Partner_salary','Total_salary','Pric
corr = df[col_list].corr()
corr
```

Out[132…

	Age	No_of_Dependents	Salary	Partner_salary	Total_salary	
Age	1.000000	-0.189614	0.616899	0.135702	0.458869	0
No_of_Dependents	-0.189614	1.000000	-0.031746	0.144320	0.092890	-0
Salary	0.616899	-0.031746	1.000000	0.087155	0.641560	0
Partner_salary	0.135702	0.144320	0.087155	1.000000	0.820069	0
Total_salary	0.458869	0.092890	0.641560	0.820069	1.000000	0
Price	0.797831	-0.135839	0.409920	0.171875	0.367823	1

In [133…

```
# Heatmap to plot correlation between all numerical variables in the dataset

col_list = ['Age','No_of_Dependents','Salary','Partner_salary','Total_salary','Pric
sns.heatmap(df[col_list].corr(), annot=True, vmin=-1, vmax=1, fmt=".2f", cmap="Spec
plt.title('Fig 16: Correlation Between Numerical Variables')
plt.show()
```



Observations and Insights:

1. There is strong correation between age and salary.
2. There is strong correlation between age and price.
3. There is strong correlation between salary and total salary.
4. There is strong correlation between partner salary and total salary.
5. There is moderate correlation between age and total salary.
6. There is moderate correlation between salary and price.
7. There is moderate correlation between total salary and price.
8. There is weak correlation between age and partner salary.
9. There is weak correlation between partner salary and price.

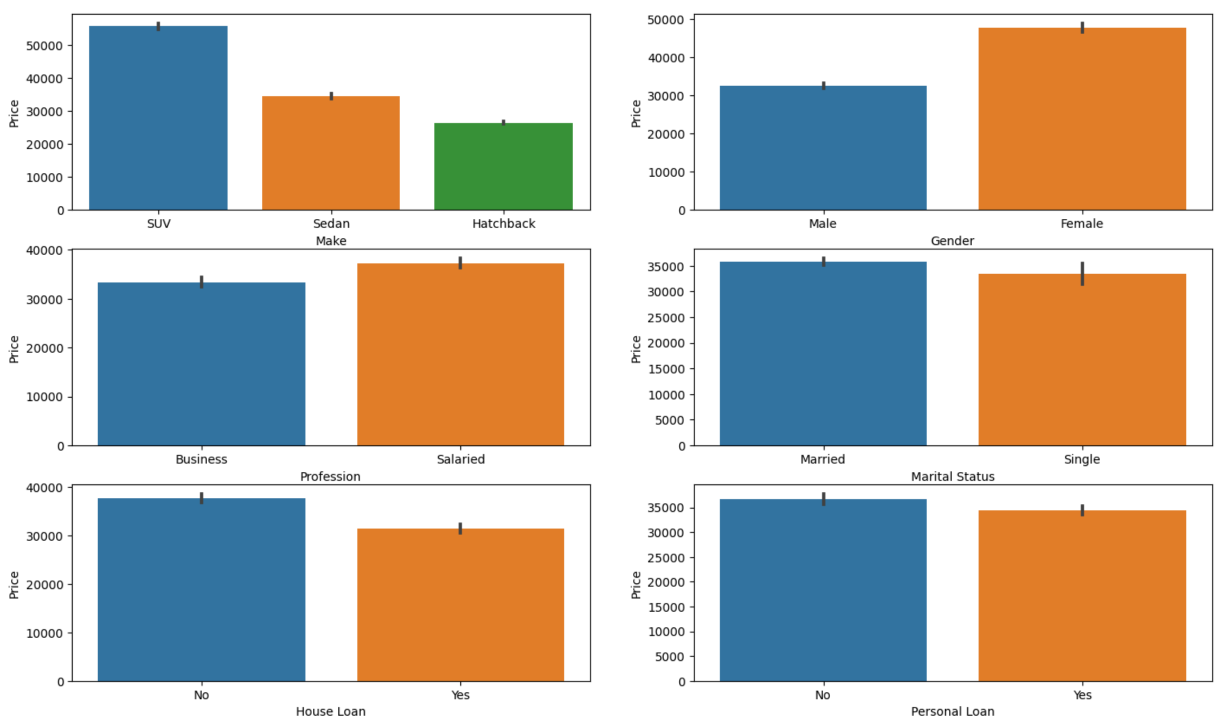
Price vs Make, Gender, Profession, Marital Status, House Loan, Personal Loan

In [134…

```
# Bar Plots for Price vs Make, Gender, Profession, Marital Status, House Loan, Pers

fig, axes = plt.subplots(3,2, figsize=(17, 10))

sns.barplot(ax=axes[0, 0], data=df, x='Make', y='Price')
sns.barplot(ax=axes[0, 1], data=df, x='Gender', y='Price')
```

```
sns.barplot(ax=axes[1, 0], data=df, x='Profession', y='Price')
sns.barplot(ax=axes[1, 1], data=df, x='Marital_status', y='Price')
sns.barplot(ax=axes[2, 0], data=df, x='House_loan', y='Price')
sns.barplot(ax=axes[2, 1], data=df, x='Personal_loan', y='Price')

axes[0,0].set(xlabel='Make', ylabel='Price')
axes[0,1].set(xlabel='Gender', ylabel='Price')
axes[1,0].set(xlabel='Profession', ylabel='Price')
axes[1,1].set(xlabel='Marital Status', ylabel='Price')
axes[2,0].set(xlabel='House Loan', ylabel='Price')
axes[2,1].set(xlabel='Personal Loan', ylabel= 'Price')

plt.suptitle('Fig 17: Price vs Make, Gender, Profession, Marital Status, House Loan

plt.show()
```

Fig 17: Price vs Make, Gender, Profession, Marital Status, House Loan, Personal Loan



Observations and Insights:

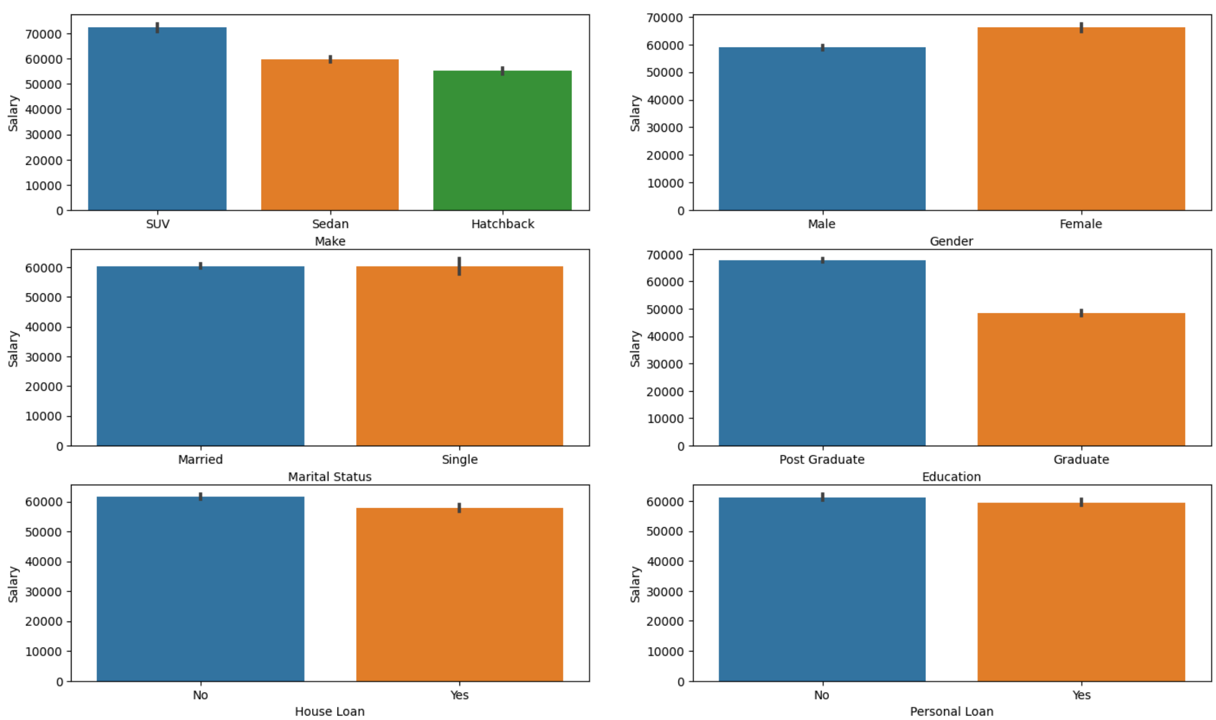
1. Price of SUV car is highest followed by Sedan and Hatchback cars.
2. Female customers are able to buy more expensive cars than male customers.
3. Salaried class customers are able to buy more expensive cars than business class customers.
4. Married customers are able to buy more expensive cars than single customers.
5. Customers without house loan are able to buy more expensive cars.
6. Customers without personal loan are able to buy more expensive cars.

Salary vs Make, Gender, Marital Status, Education, House Loan, Personal Loan

In [135…

```
# Bar Plots for Salary vs Make, Gender, Marital Status, Education, House Loan, Pers

fig, axes = plt.subplots(3, 2, figsize=(17, 10))

sns.barplot(ax=axes[0, 0], data=df, x='Make',y='Salary')
sns.barplot(ax=axes[0, 1], data=df, x='Gender',y='Salary')
sns.barplot(ax=axes[1, 0], data=df, x='Marital_status',y='Salary')
sns.barplot(ax=axes[1, 1], data=df, x='Education',y='Salary')
sns.barplot(ax=axes[2, 0], data=df, x='House_loan',y='Salary')
sns.barplot(ax=axes[2, 1], data=df, x='Personal_loan',y='Salary')

axes[0,0].set(xlabel = 'Make', ylabel = 'Salary')
axes[0,1].set(xlabel = 'Gender', ylabel = 'Salary')
axes[1,0].set(xlabel = 'Marital Status', ylabel = 'Salary')
axes[1,1].set(xlabel = 'Education', ylabel = 'Salary')
axes[2,0].set(xlabel = 'House Loan', ylabel = 'Salary')
axes[2,1].set(xlabel = 'Personal Loan', ylabel = 'Salary')

plt.suptitle('Fig 18: Salary vs Make, Gender, Marital Status, Education, House Loan

plt.show()
```

Fig 18: Salary vs Make, Gender, Marital Status, Education, House Loan, Personal Loan



Observations and Insights:

1. Customers with higher Salary are able to buy SUV cars and customers with lesser Salary are able to buy Sedan and Hatchback cars.
2. Female customers are having higher Salary than male customers.
3. Marital Status does not have much impact on salary of customers.
4. Post Graduate customers are having higher Salary than Graduate customers.

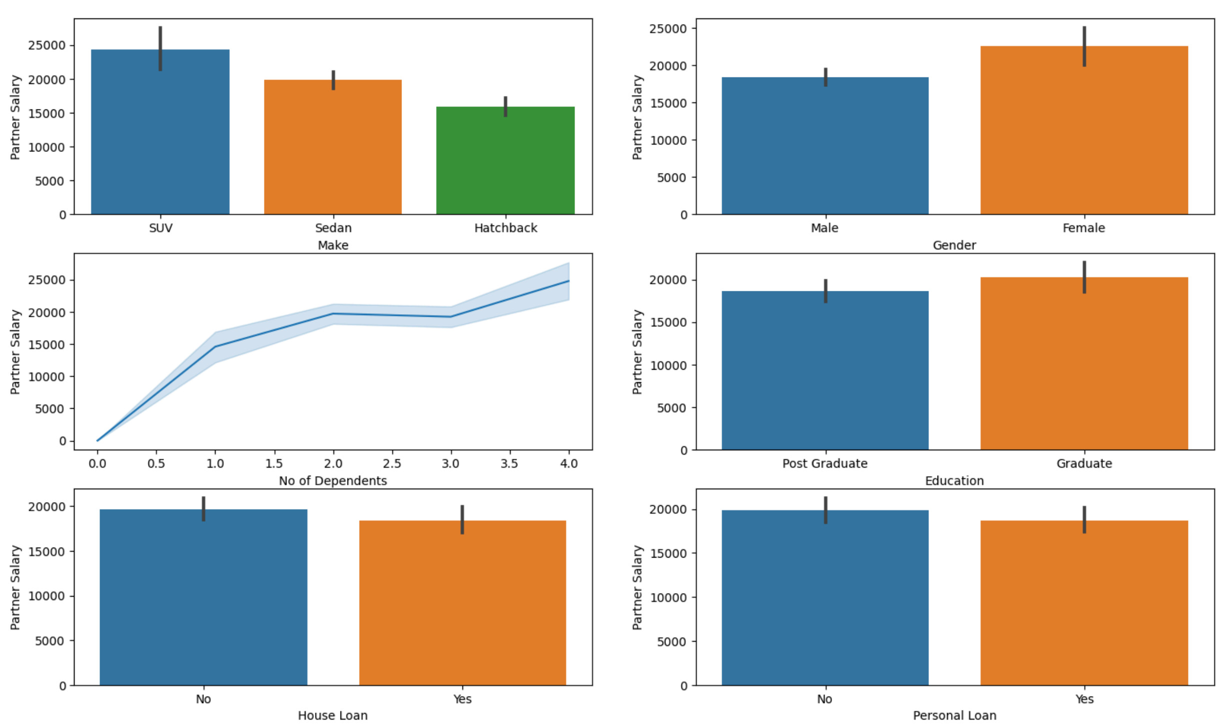
5. Customers without house loan are having higher Salary.
6. Customers without personal loan are having higher Salary.

Partner Salary vs Make, Gender, No of Dependents, Education, House Loan, Personal Loan

In [136…

```
# Bar Plots for Partner Salary vs Make, Gender, No of Dependents, Education, House

fig, axes = plt.subplots(3, 2, figsize=(17, 10))

sns.barplot(ax=axes[0, 0], data=df, x='Make',y='Partner_salary')
sns.barplot(ax=axes[0, 1], data=df, x='Gender',y='Partner_salary')
sns.lineplot(ax=axes[1, 0], data=df, x='No_of_Dependents',y='Partner_salary')
sns.barplot(ax=axes[1, 1], data=df, x='Education',y='Partner_salary')
sns.barplot(ax=axes[2, 0], data=df, x='House_loan',y='Partner_salary')
sns.barplot(ax=axes[2, 1], data=df, x='Personal_loan',y='Partner_salary')

axes[0,0].set(xlabel='Make', ylabel='Partner Salary')
axes[0,1].set(xlabel='Gender', ylabel='Partner Salary')
axes[1,0].set(xlabel='No of Dependents', ylabel='Partner Salary')
axes[1,1].set(xlabel='Education', ylabel='Partner Salary')
axes[2,0].set(xlabel='House Loan', ylabel='Partner Salary')
axes[2,1].set(xlabel='Personal Loan', ylabel='Partner Salary')

plt.suptitle('Fig 19: Partner Salary vs Make, Gender, No of Dependents, Education,

plt.show()
```

Fig 19: Partner Salary vs Make, Gender, No of Dependents, Education, House Loan, Personal Loan



Observations and Insights:

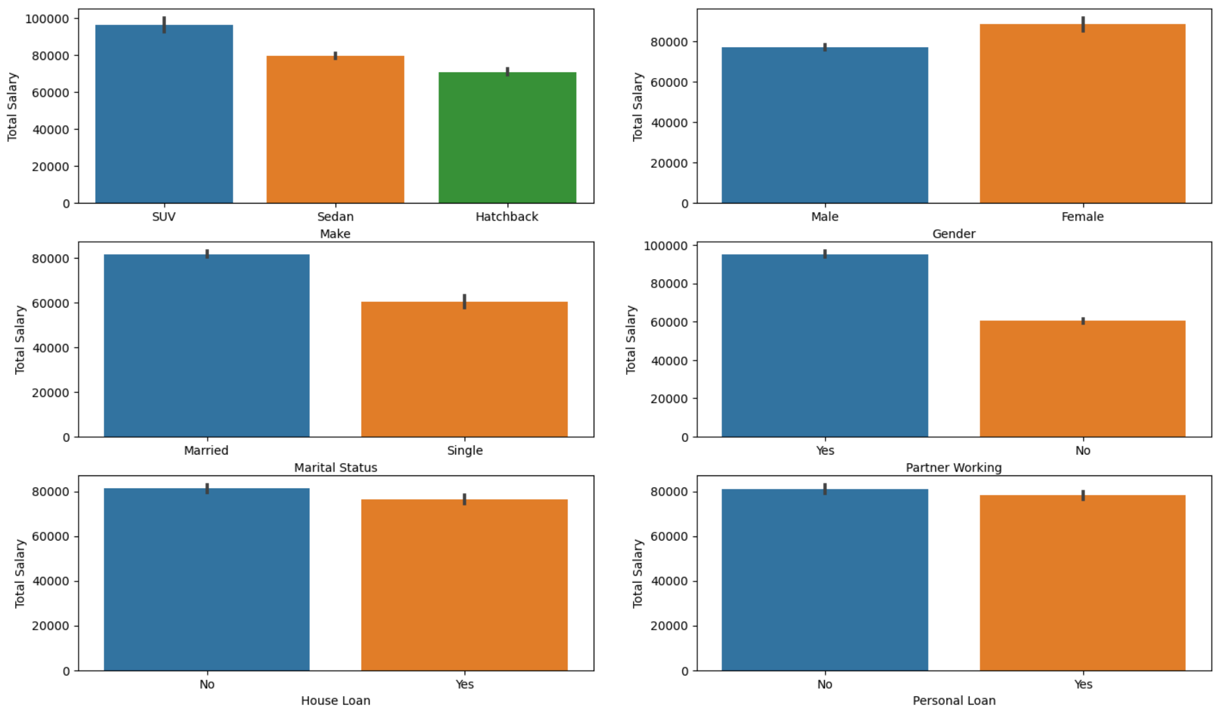
1. Customers with higher partner salary are able to buy SUV cars and customers with lesser partner salary are able to buy Sedan and Hatchback cars.
2. Female customers are having higher partner salary than male customers.
3. No of dependents increases when partner salary increases.
4. Graduate customers are having higher partner salary than Post Graduate customers.
5. Customers without house loan are having higher partner salary.
6. Customers without personal loan are having higher partner salary.

Total Salary vs Make, Gender, Marital Status, Partner Working, House Loan, Personal Loan

In [137…

```
# Bar Plots for Total Salary vs Make, Gender, Marital Status, Partner Working, Hous

fig, axes = plt.subplots(3, 2, figsize=(17, 10))

sns.barplot(ax=axes[0, 0], data=df, x='Make',y='Total_salary')
sns.barplot(ax=axes[0, 1], data=df, x='Gender',y='Total_salary')
sns.barplot(ax=axes[1, 0], data=df, x='Marital_status',y='Total_salary')
sns.barplot(ax=axes[1, 1], data=df, x='Partner_working',y='Total_salary')
sns.barplot(ax=axes[2, 0], data=df, x='House_loan',y='Total_salary')
sns.barplot(ax=axes[2, 1], data=df, x='Personal_loan',y='Total_salary')

axes[0,0].set(xlabel = 'Make', ylabel = 'Total Salary')
axes[0,1].set(xlabel = 'Gender', ylabel = 'Total Salary')
axes[1,0].set(xlabel = 'Marital Status', ylabel = 'Total Salary')
axes[1,1].set(xlabel = 'Partner Working', ylabel = 'Total Salary')
axes[2,0].set(xlabel = 'House Loan', ylabel = 'Total Salary')
axes[2,1].set(xlabel = 'Personal Loan', ylabel = 'Total Salary')

plt.suptitle('Fig 20: Total Salary vs Make, Gender, Marital Status, Partner Working

plt.show()
```

Fig 20: Total Salary vs Make, Gender, Marital Status, Partner Working, House Loan, Personal Loan



Observations and Insights:

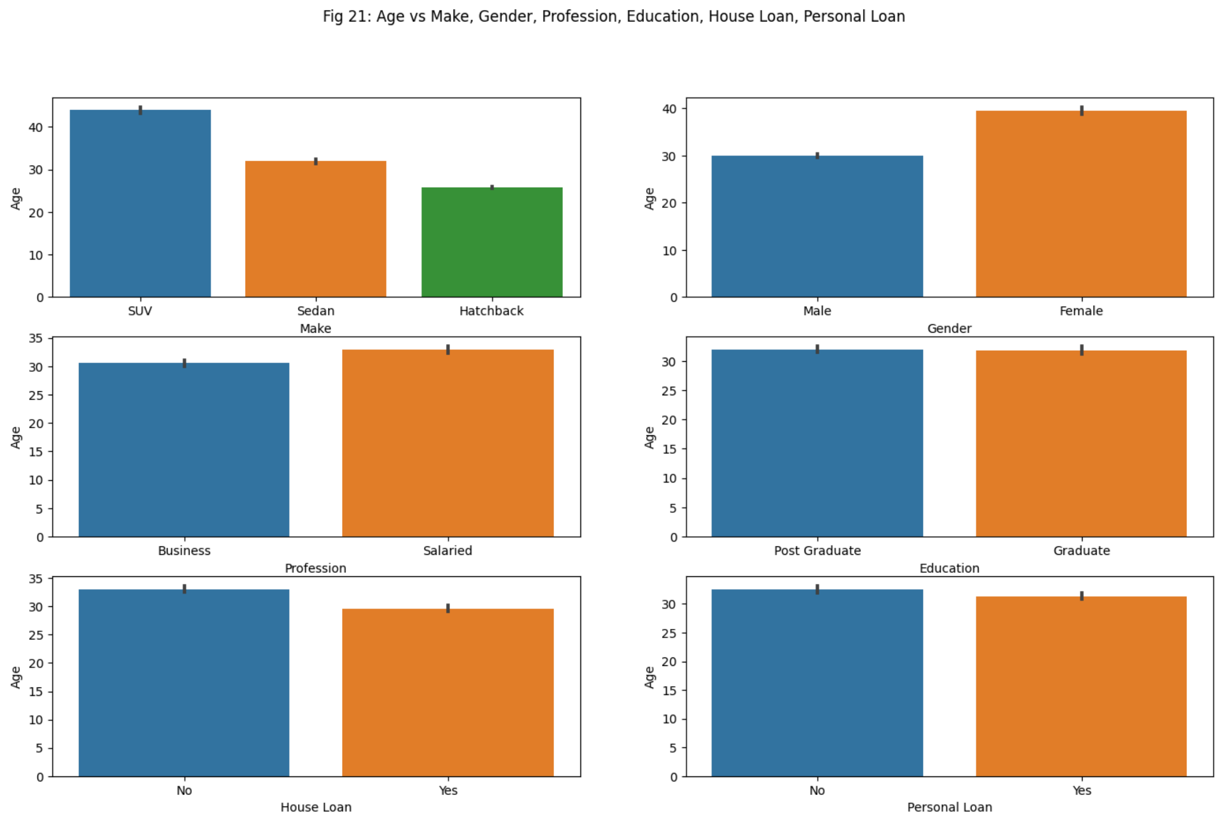
1. Customers with higher total salary are able to buy SUV cars and customers with lesser total salary are able to buy Sedan and Hatchback cars.
2. Female customers are having higher total salary than male customers.
3. Married customers are having higher total salary than single customers.
4. Customers with working partners are having higher total salary.
5. Customers without house loan are having higher total salary.
6. Customers without personal loan are having higher total salary.

Age vs Make, Gender, Profession, Education, House Loan, Personal Loan

In [138…

```
# Bar Plots for Age vs Make, Gender, Profession, Education, House Loan, Personal Lo

fig, axes = plt.subplots(3, 2, figsize=(17, 10))

sns.barplot(ax=axes[0, 0], data=df, x='Make', y='Age')
sns.barplot(ax=axes[0, 1], data=df, x='Gender', y='Age')
sns.barplot(ax=axes[1, 0], data=df, x='Profession', y='Age')
sns.barplot(ax=axes[1, 1], data=df, x='Education', y='Age')
sns.barplot(ax=axes[2, 0], data=df, x='House_loan', y='Age')
sns.barplot(ax=axes[2, 1], data=df, x='Personal_loan', y='Age')

axes[0,0].set(xlabel='Make', ylabel='Age')
axes[0,1].set(xlabel='Gender', ylabel='Age')
axes[1,0].set(xlabel='Profession', ylabel='Age')
axes[1,1].set(xlabel='Education', ylabel='Age')
axes[2,0].set(xlabel='House Loan', ylabel='Age')
```

```
axes[2,1].set(xlabel='Personal Loan', ylabel='Age')

plt.suptitle('Fig 21: Age vs Make, Gender, Profession, Education, House Loan, Perso

plt.show()
```



Observations and Insights:

1. Customers with higher age bought more SUV cars than Sedan and Hatchback cars.
2. Female customers have higher age than male customers.
3. Salaried class customers are having higher age than business class customers.
4. Education does not have much impact on age of customers.
5. Customers without house loan have higher age.
6. Customers without personal loan have higher age.

Multivariate Analysis

Price vs Make and Marital Status

In [139…

```
# Point Plot for Price vs Make and Marital Status

sns.pointplot(data=df, x='Make', y='Price', hue='Marital_status')
plt.title('Fig 22: Price vs Make and Marital Status')
plt.xlabel('Make')
plt.ylabel('Price')
plt.show()
```

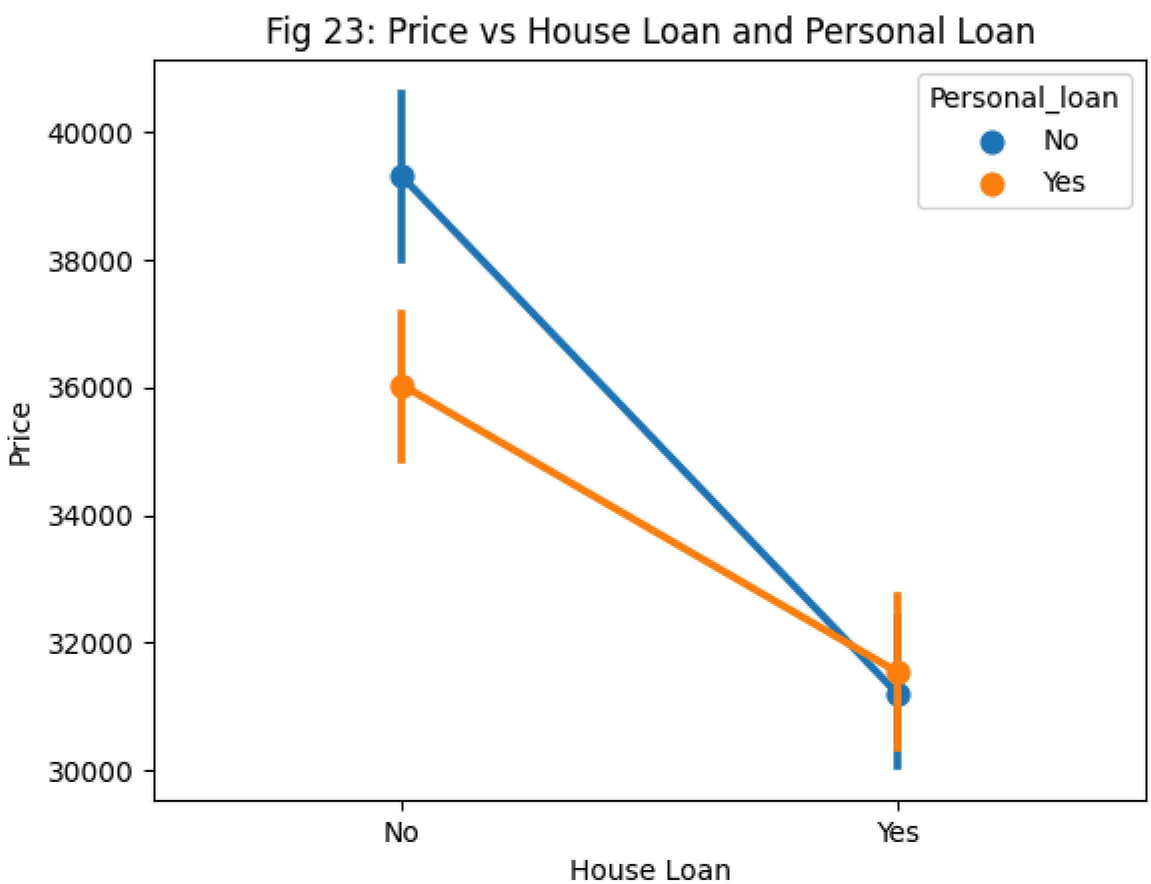
Observations and Insights:

Married and Single customers are buying more SUV cars than Sedan and Hatchback cars.

Price vs House Loan and Personal Loan

In [140…

```
# Point Plot for Price vs House Loan and Personal Loan

sns.pointplot(data=df, x='House_loan', y='Price', hue='Personal_loan')
plt.title('Fig 23: Price vs House Loan and Personal Loan')
plt.xlabel('House Loan')
plt.ylabel('Price')
plt.show()
```



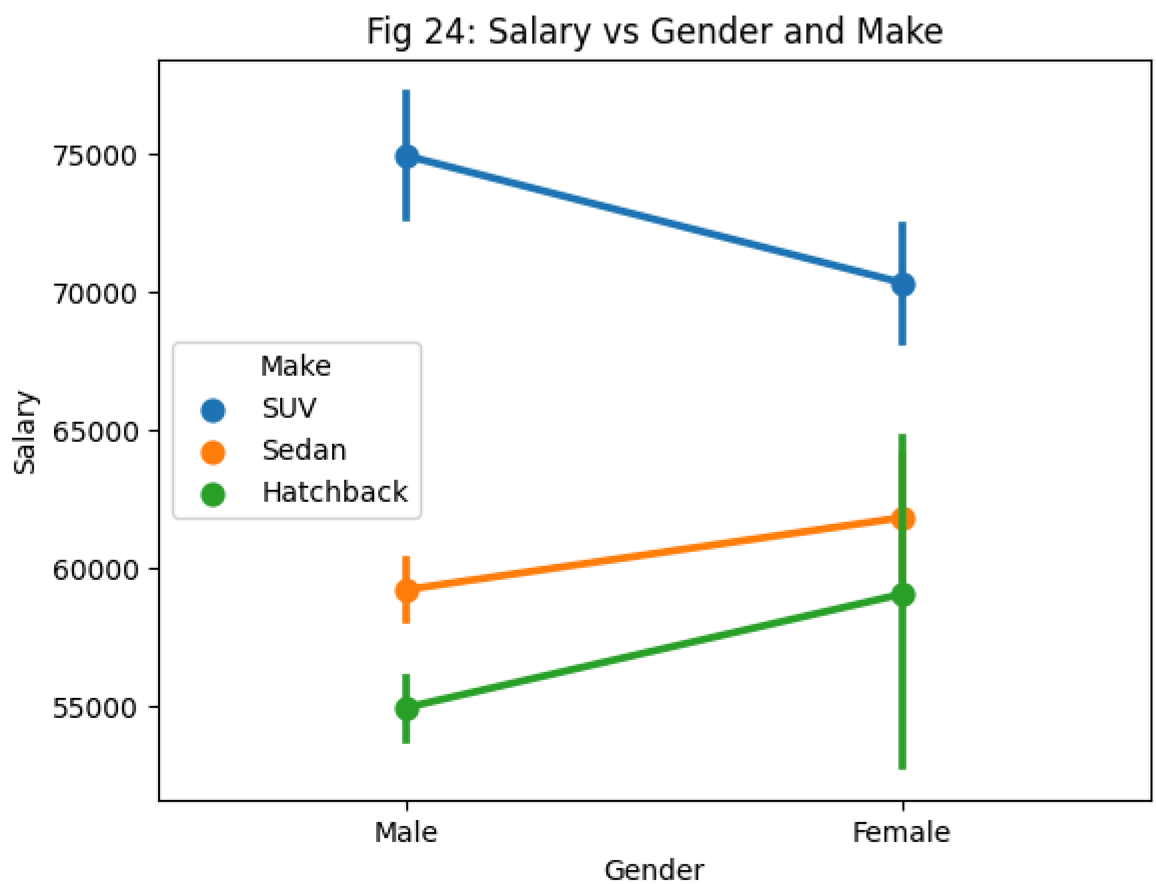
Observations and Insights:

Customers without house and personal loan are able to buy more expensive cars.

Salary vs Gender and Make

In [141…

```
# Point Plot for Salary vs Gender and Make

sns.pointplot(data=df, x='Gender', y='Salary', hue='Make')
plt.title('Fig 24: Salary vs Gender and Make')
plt.xlabel('Gender')
plt.ylabel('Salary')
plt.show()
```



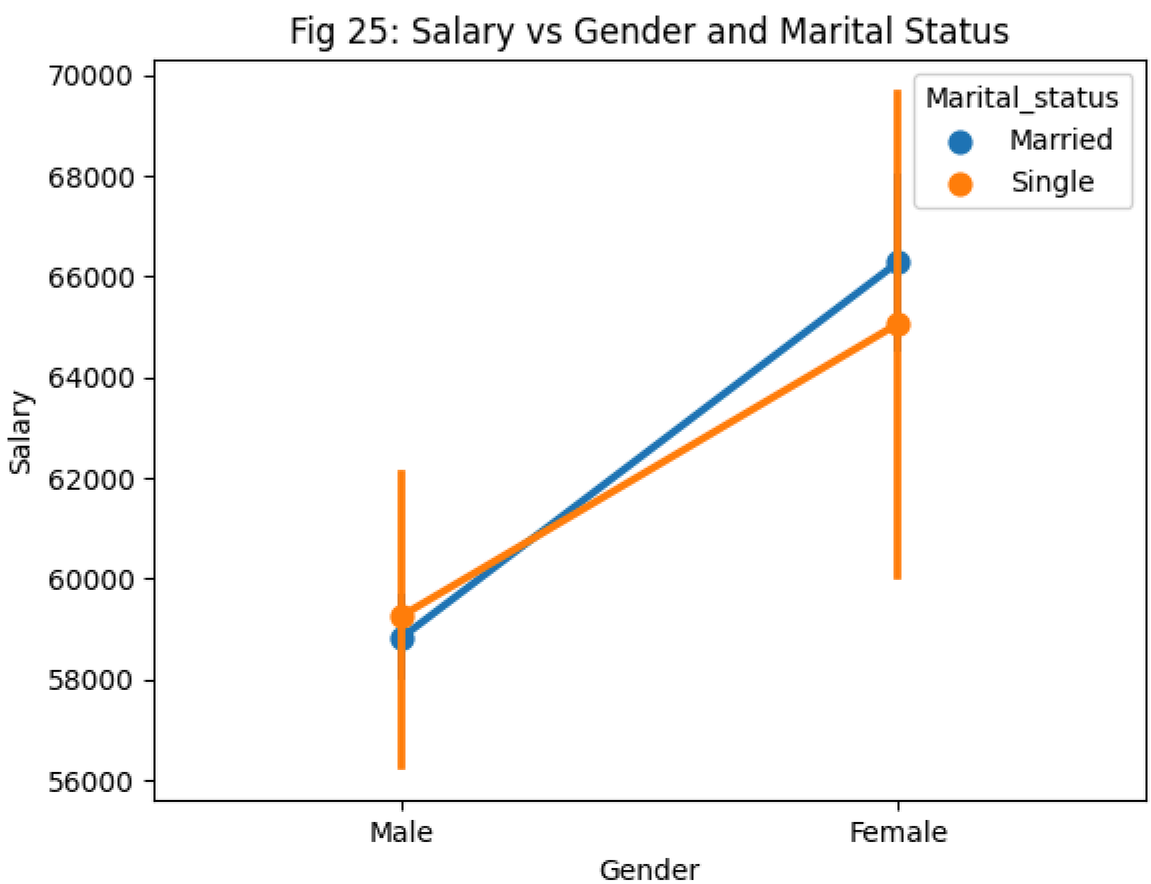
Observations and Insights:

Male and Female customers are able to buy more expensive cars if their salary is high.

Salary vs Gender and Marital Status

In [142…

```
# Point Plot for Salary vs Gender and Marital Status

sns.pointplot(data=df, x='Gender', y='Salary', hue='Marital_status')
plt.title('Fig 25: Salary vs Gender and Marital Status')
plt.xlabel('Gender')
plt.ylabel('Salary')
plt.show()
```

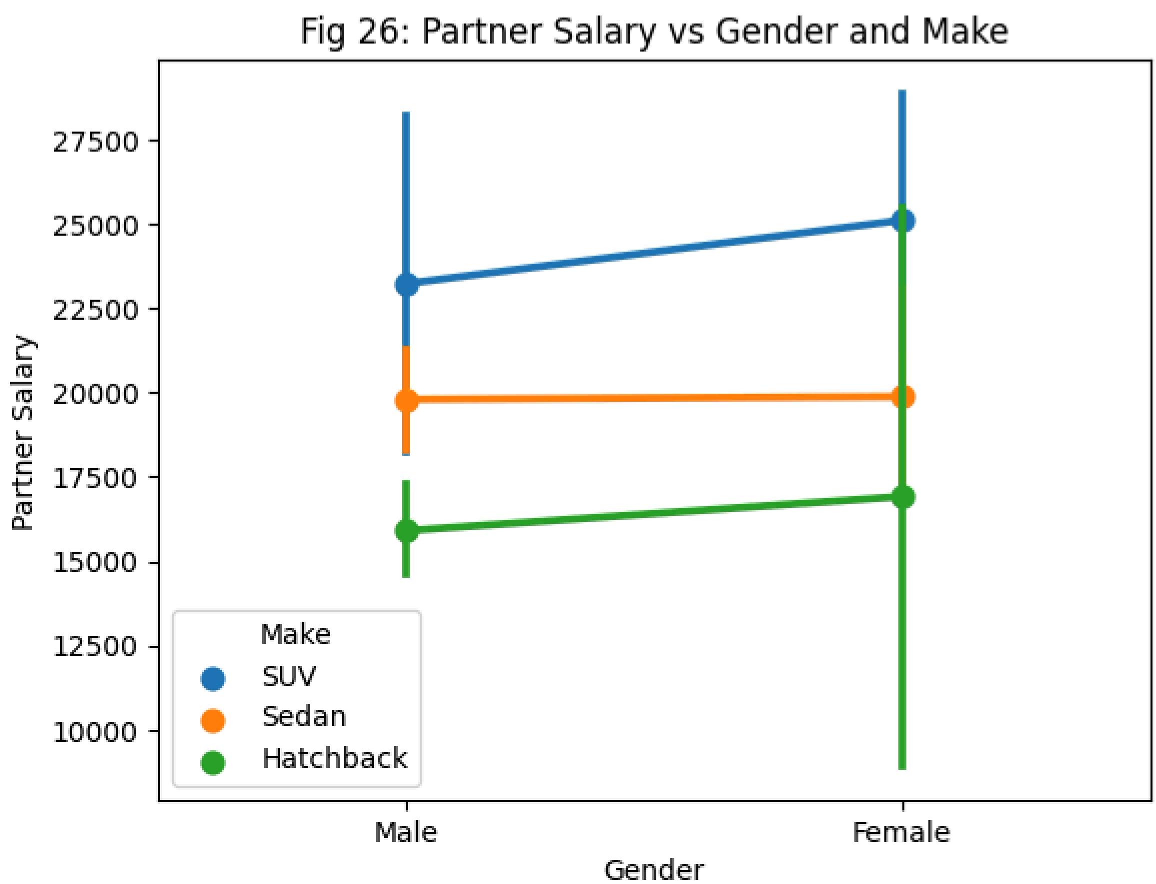


Observations and Insights:
Female customers salary is high in comparison to Male customers whether they are married or not.

Partner Salary vs Gender and Make

In [143…

```
# Point Plot for Partner Salary vs Gender and Make

sns.pointplot(data=df, x='Gender', y='Partner_salary', hue='Make')
plt.title('Fig 26: Partner Salary vs Gender and Make')
plt.xlabel('Gender')
plt.ylabel('Partner Salary')
plt.show()
```

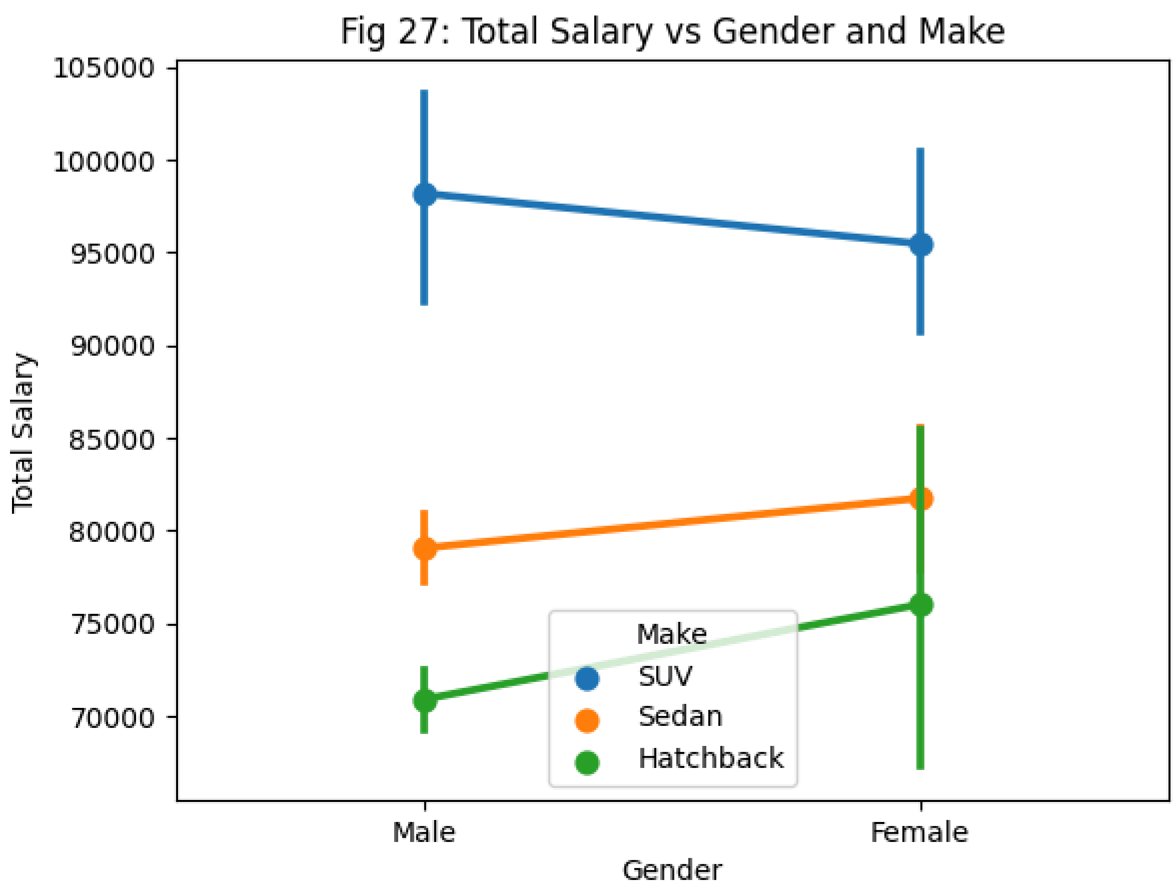


Observations and Insights:
Male and Female customers are able to buy more expensive cars if their partner salary is high.

Total Salary vs Make and Gender

In [144...

```
# Point Plot for Total Salary vs Gender and Make

sns.pointplot(data=df, x='Gender', y='Total_salary', hue='Make')
plt.title('Fig 27: Total Salary vs Gender and Make')
plt.xlabel('Gender')
plt.ylabel('Total Salary')
plt.show()
```



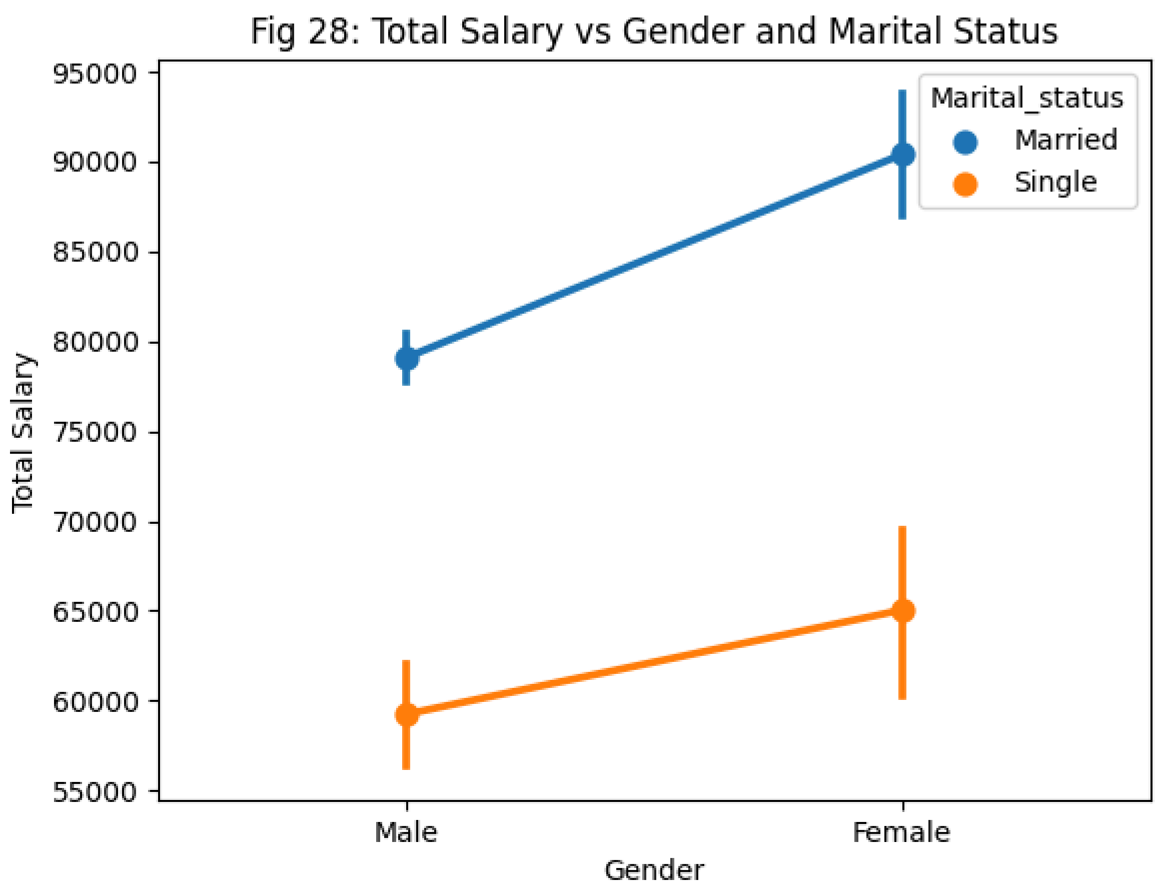
Observations and Insights:

Male and Female customers are able to buy more expensive cars if their total salary is high.

Total Salary vs Gender and Marital Status

In [145…

```
# Point Plot for Total Salary vs Make and Marital Status

sns.pointplot(data=df, x='Gender', y='Total_salary', hue='Marital_status')
plt.title('Fig 28: Total Salary vs Gender and Marital Status')
plt.xlabel('Gender')
plt.ylabel('Total Salary')
plt.show()
```

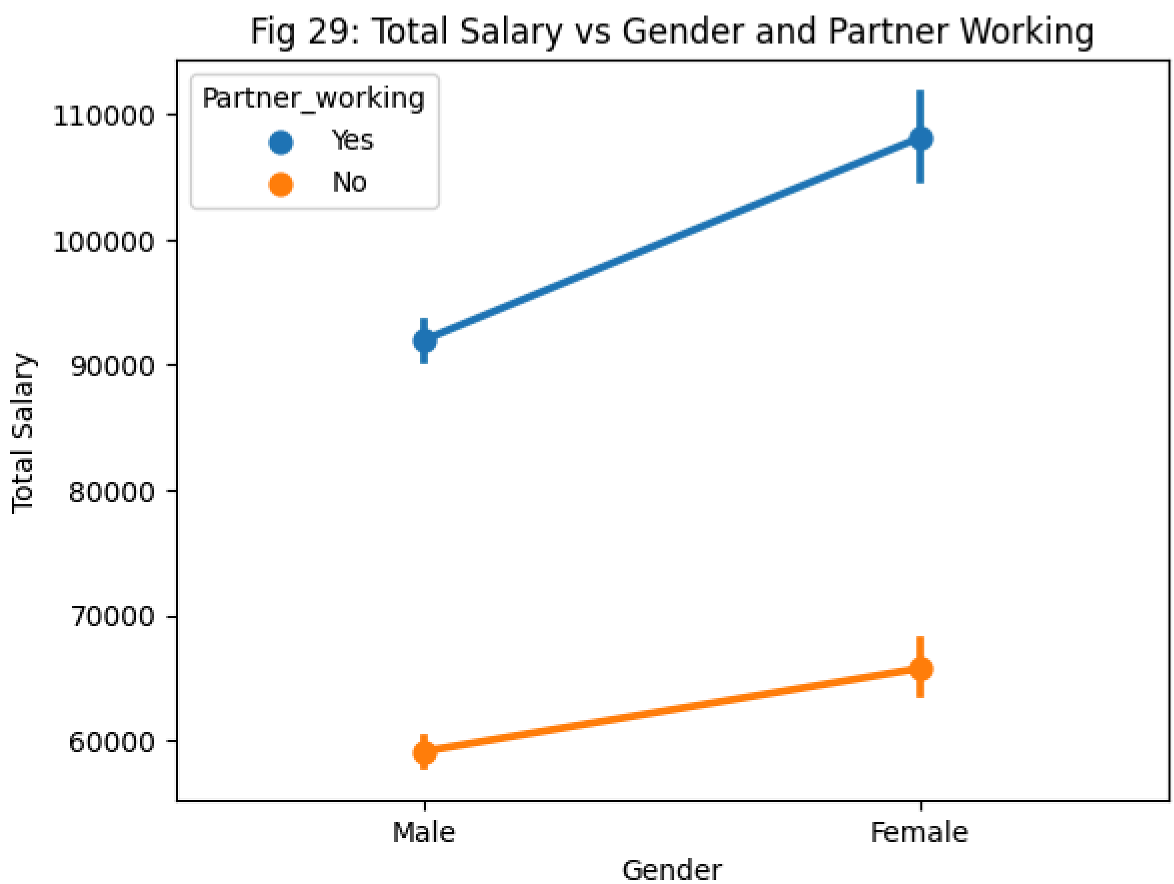


Observations and Insights:
Female customers total salary is high in comparison to Male customers whether they are married or not.

Total Salary vs Gender and Partner Working

In [146…

```
# Point Plot for Total Salary vs Gender and Partner Working

sns.pointplot(data=df, x='Gender', y='Total_salary', hue='Partner_working')
plt.title('Fig 29: Total Salary vs Gender and Partner Working')
plt.xlabel('Gender')
plt.ylabel('Total Salary')
plt.show()
```

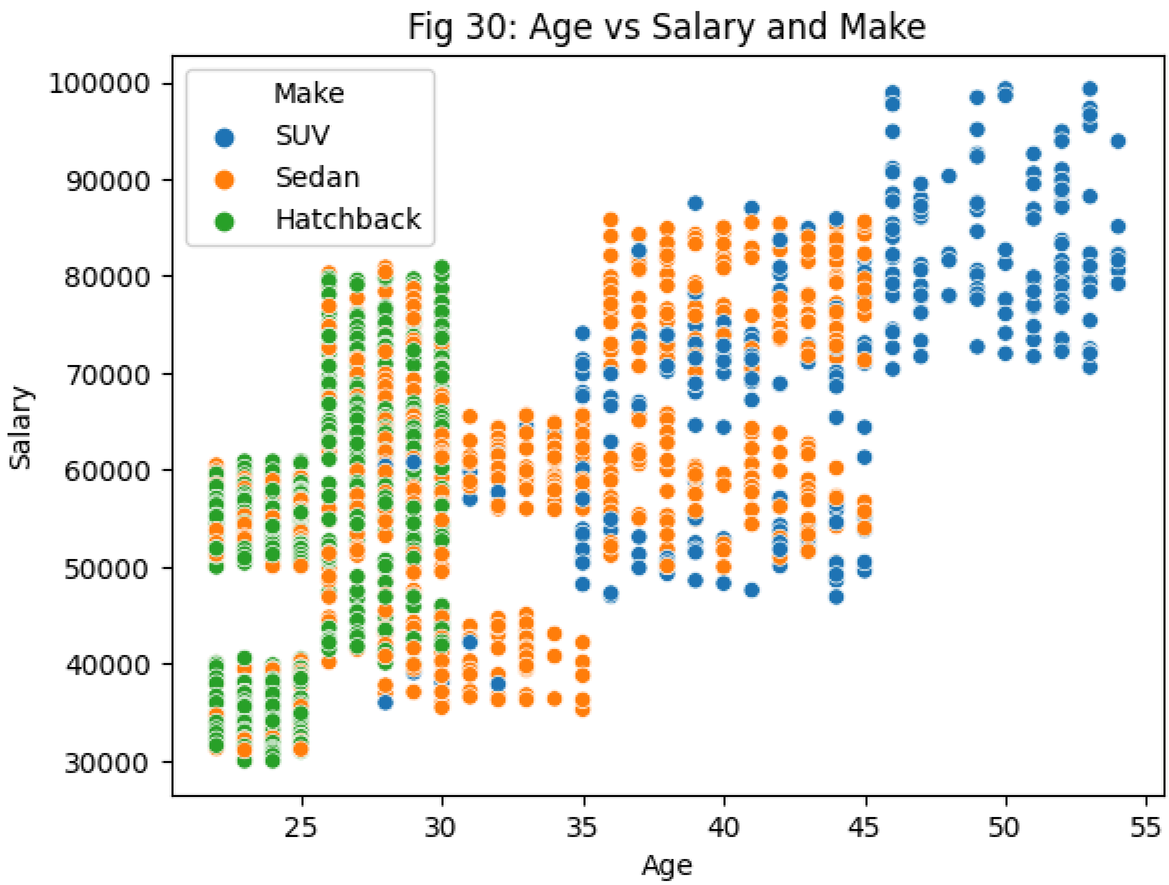


Observations and Insights:
Female customers total salary is high in comparison to Male customers if their partner is working.

Age vs Salary and Make

In [147…

```
# Scatter Plot for Age vs Salary and Make

sns.scatterplot(data=df, x='Age', y='Salary', hue='Make')
plt.title('Fig 30: Age vs Salary and Make')
plt.xlabel('Age')
plt.ylabel('Salary')
plt.show()
```



Observations and Insights:

Customers are able to buy more expensive cars when their Age and Salary increases.

Age vs Price and Make

In [148…

```
# Scatter Plot for Age vs Price and Make

sns.scatterplot(data=df, x='Age', y='Price', hue='Make')
plt.title('Fig 31: Age vs Price and Make')
plt.xlabel('Age')
plt.ylabel('Price')
plt.show()
```

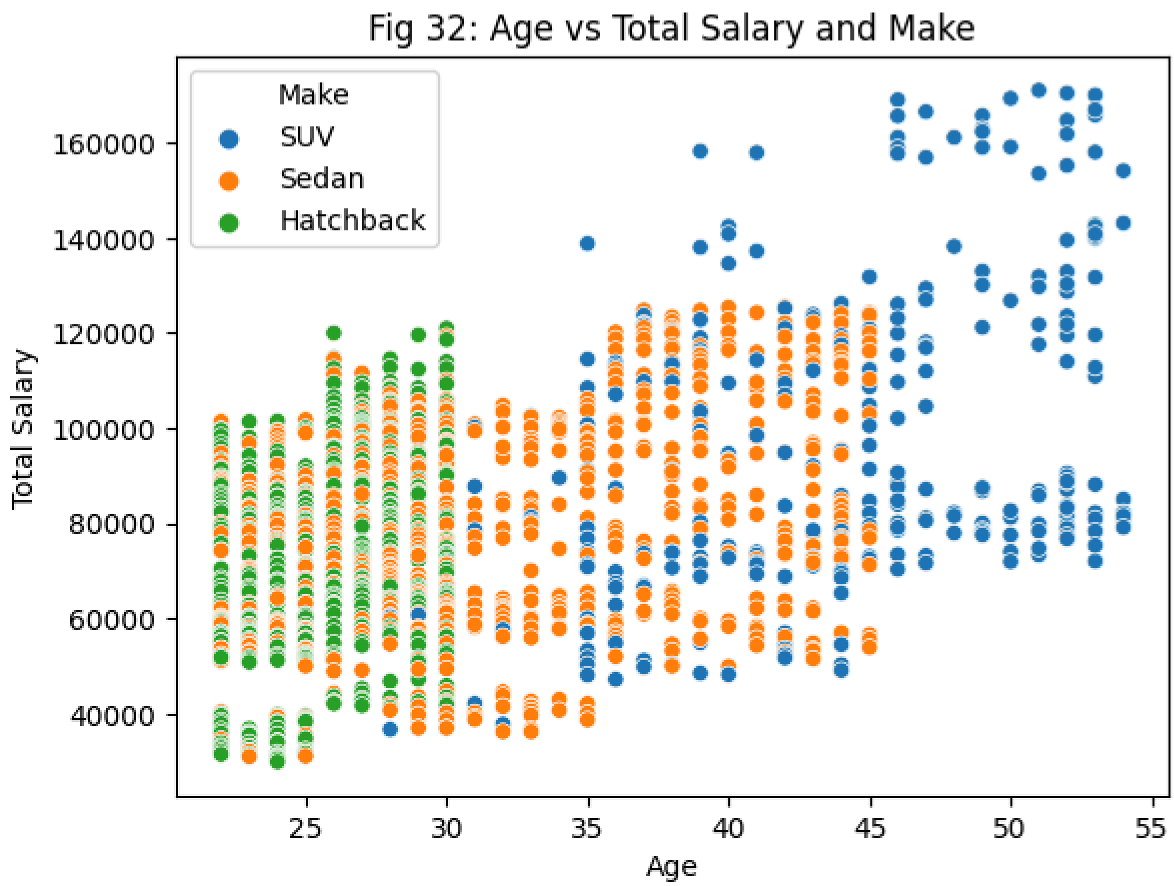
Observations and Insights:

Customers are able to buy more expensive cars when their Age increases.

Age vs Total Salary and Make

In [149…

```
# Scatter Plot for Age vs Total Salary and Make

sns.scatterplot(data=df, x='Age', y='Total_salary', hue='Make')
plt.title('Fig 32: Age vs Total Salary and Make')
plt.xlabel('Age')
plt.ylabel('Total Salary')
plt.show()
```



Observations and Insights:

Customers are able to buy more expensive cars when their Age and Salary increases.

Question 1: Do men tend to prefer SUVs more compared to women?

In [154…

```
df_gmc = df.groupby(['Gender','Make']).agg(Count=('Make','count')).sort_values(by=[
df_gmc
```

Out[154…

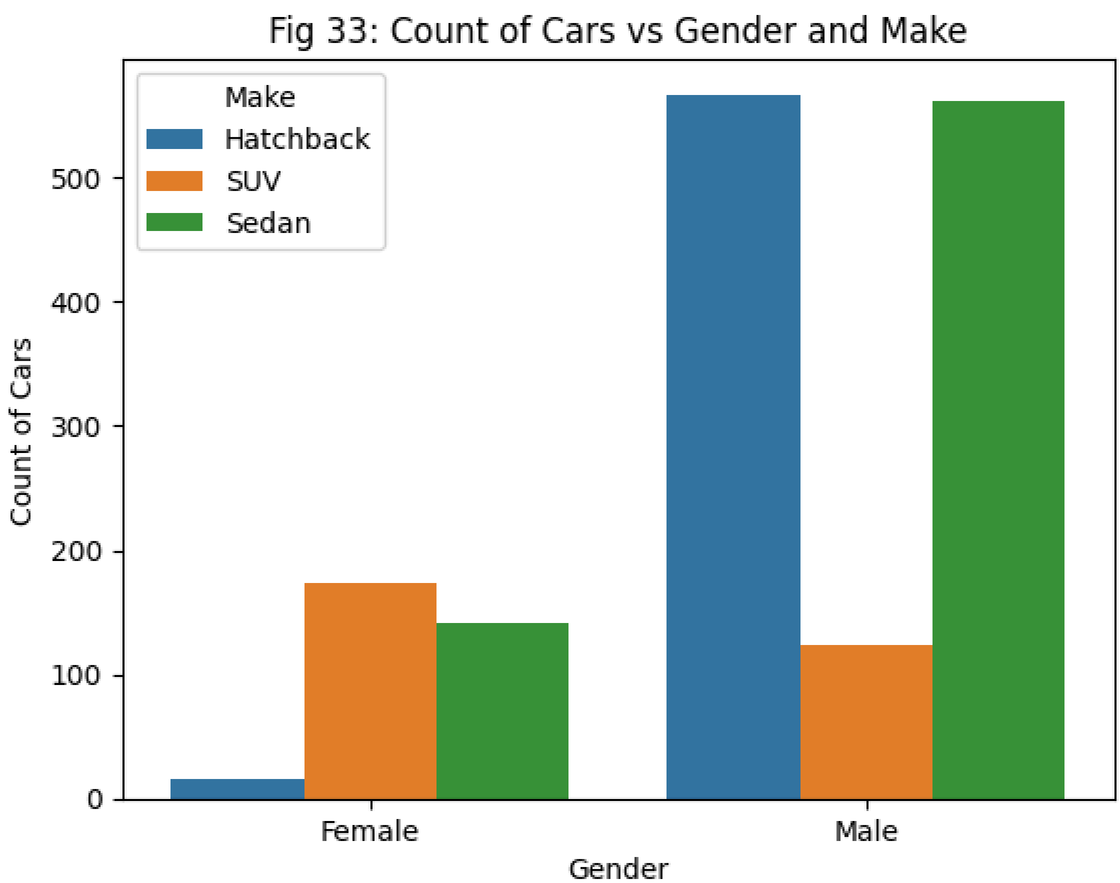
	Gender	Make	Count
0	Female	Hatchback	15
1	Male	Hatchback	567
2	Female	SUV	173
3	Male	SUV	124
4	Female	Sedan	141
5	Male	Sedan	561

No. Men prefer Sedan or Hatchback cars as first car than SUV cars compared to women.

Women prefer SUV cars as first car than Sedan or Hatchback cars compared to men.

In [155…

```
# Bar Plot for Count of Cars vs Gender and Make

sns.barplot(data=df_gmc, x='Gender', y='Count', hue='Make')
plt.title('Fig 33: Count of Cars vs Gender and Make')
plt.xlabel('Gender')
plt.ylabel('Count of Cars')
plt.show()
```



Question 2: What is the likelihood of a salaried person buying a Sedan?

In [156…

```
df_pmc = df.groupby(['Profession','Make']).agg(Count=('Make','count')).sort_values(
df_pmc
```

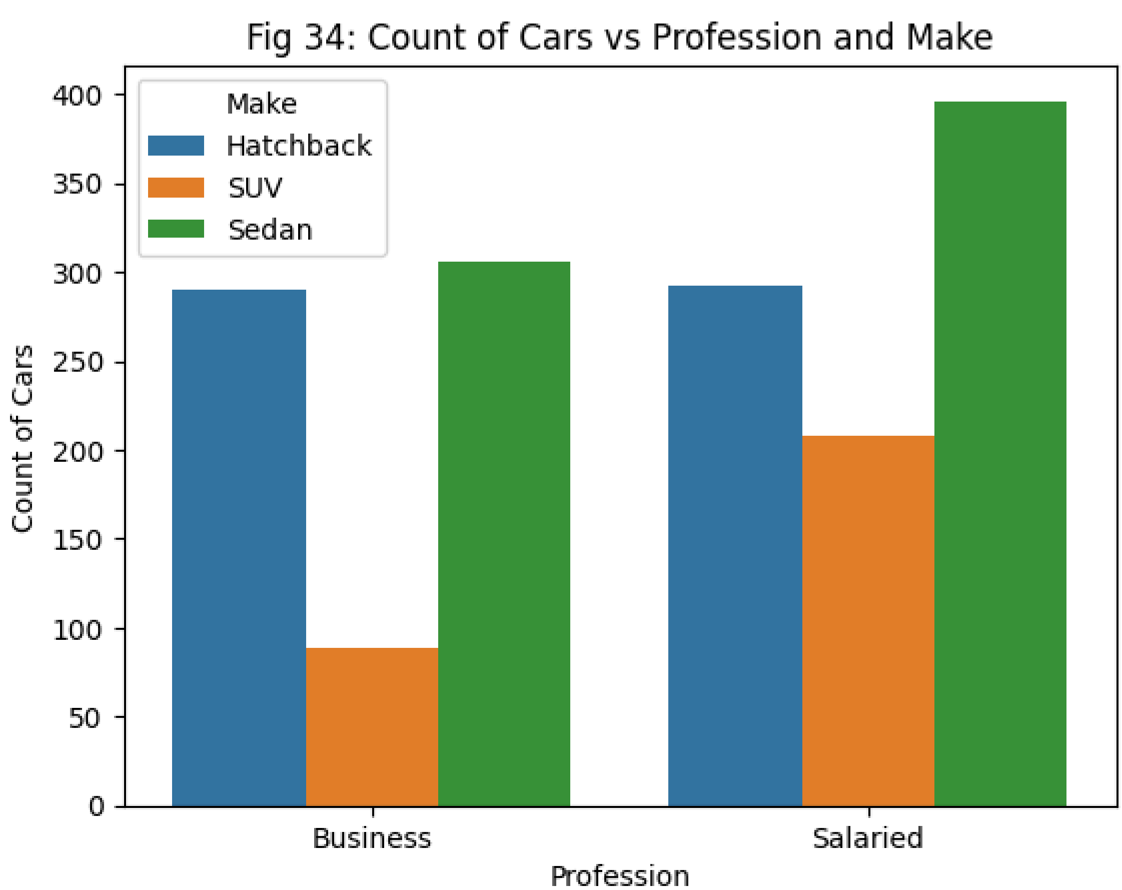
Out[156…

	Profession	Make	Count
0	Business	Hatchback	290
1	Salaried	Hatchback	292
2	Business	SUV	89
3	Salaried	SUV	208
4	Business	Sedan	306
5	Salaried	Sedan	396

Maximum number of cars bought by Salaried class customers were Sedan. So there is high probability that Salaried class customers will buy Sedan car (first car) in future as well.

In [157…

```
# Bar Plot for Count of Cars vs Profession and Make

sns.barplot(data=df_pmc, x='Profession', y='Count', hue='Make')
plt.title('Fig 34: Count of Cars vs Profession and Make')
plt.xlabel('Profession')
plt.ylabel('Count of Cars')
plt.show()
```



Question 3: What evidence or data supports Sheldon Cooper's claim that a salaried male is an easier target for a SUV sale

over a Sedan sale?

In [158…

```
df_gpm_ext = df.loc[df['Profession'] == 'Salaried']

df_gpm = df_gpm_ext.groupby(['Gender','Make']).agg(Count=('Make','count')).sort_val
df_gpm
```

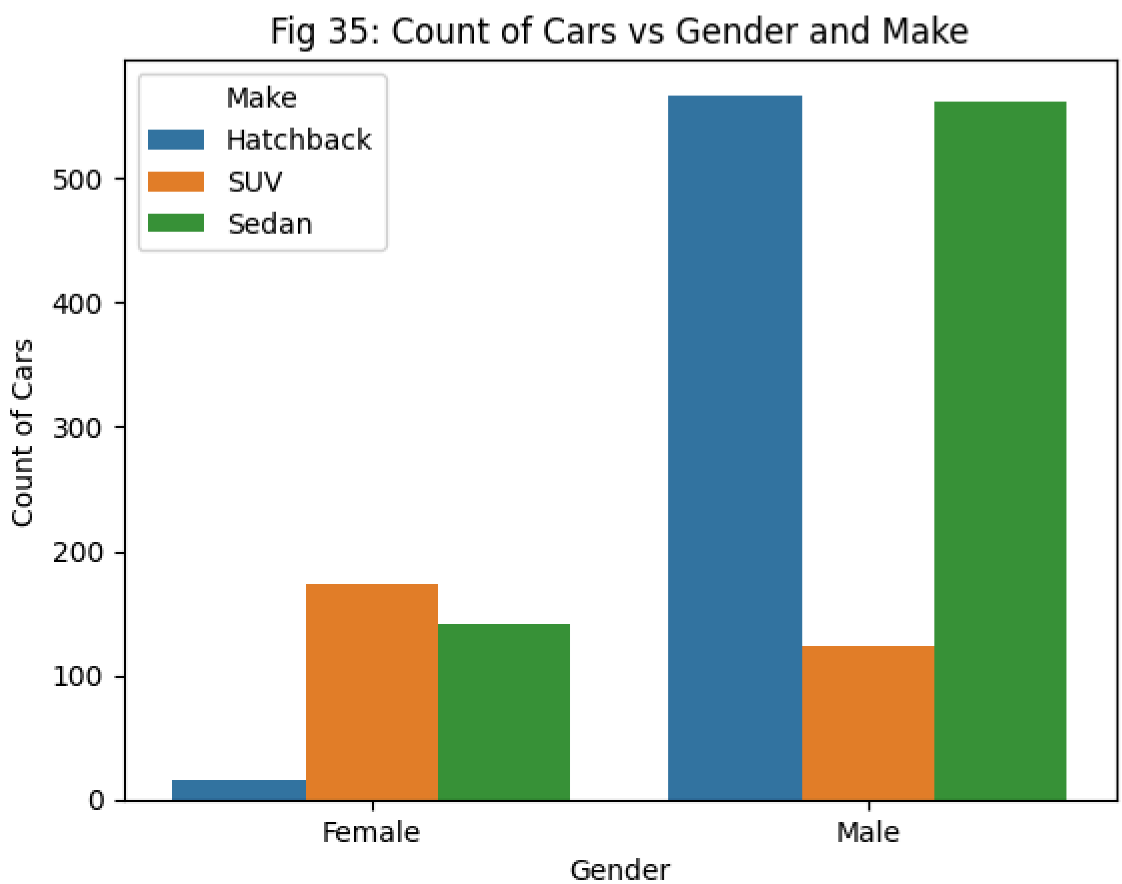
Out[158…

	Gender	Make	Count
0	Female	Hatchback	15
1	Female	SUV	118
2	Female	Sedan	91
3	Male	Hatchback	277
4	Male	SUV	90
5	Male	Sedan	305

Salaried males have bought Sedan cars in higher number than SUV cars. They will be an easier target for a SUV car sale over Sedan car in case they want to own a second car or replace the existing car. It will also help in increasing the company revenue as SUV is the most expensive car in comparison to Sedan and Hatchback cars.

In [159…

```
# Bar Plot for Count of Cars vs Gender and Make

sns.barplot(data=df_gmc, x='Gender', y='Count', hue='Make')
plt.title('Fig 35: Count of Cars vs Gender and Make')
plt.xlabel('Gender')
plt.ylabel('Count of Cars')
plt.show()
```



Question 4: How does the amount spent on purchasing automobiles vary by gender?

In [160…

```
df_pcg = df.groupby(['Gender','Make']).agg(Total_Purchase_Amount=('Price','sum')).s
df_pcg
```

Out[160…

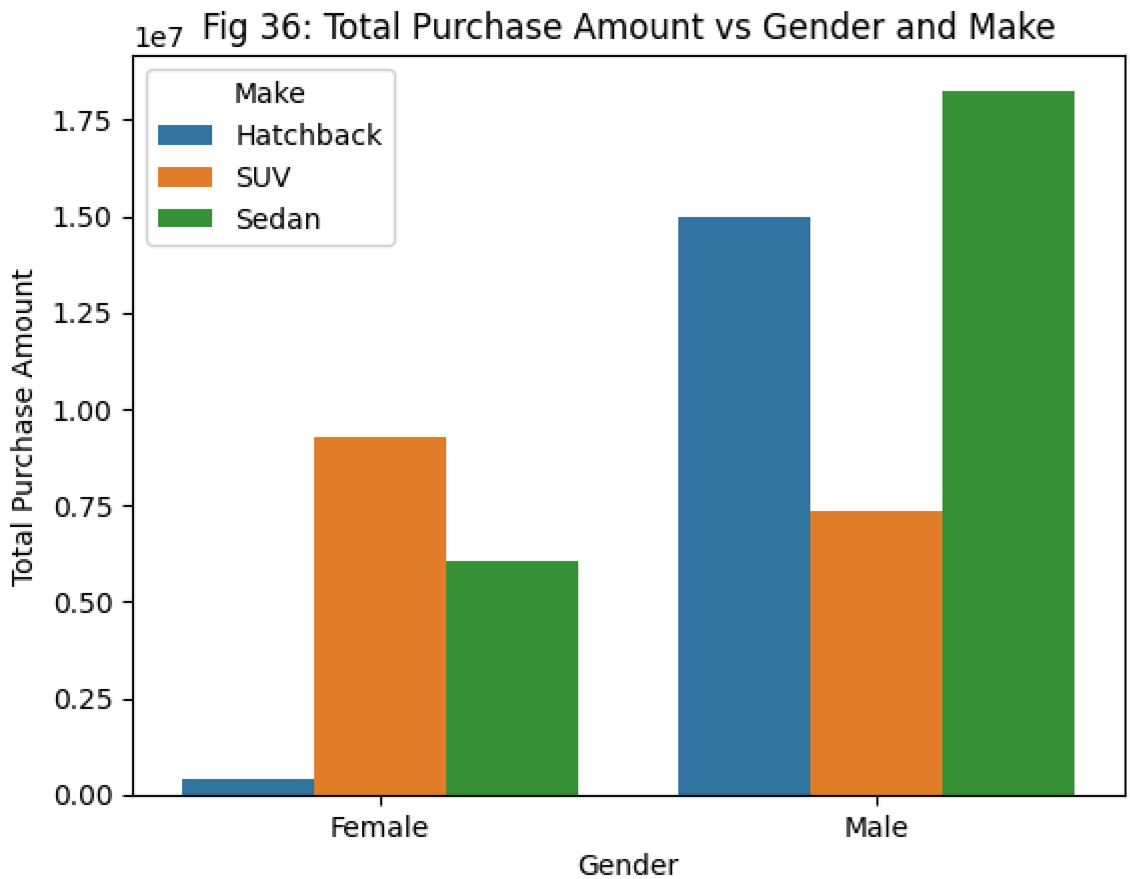
	Gender	Make	Total_Purchase_Amount
0	Female	Hatchback	412000.0
1	Female	SUV	9252000.0
2	Female	Sedan	6031000.0
3	Male	Hatchback	14996000.0
4	Male	SUV	7328000.0
5	Male	Sedan	18261000.0

Female customers spent most amount on purchasing the SUV cars than Sedan and Hatchback cars. Male customers spent most amount on purchasing the Sedan cars than Hatchback and SUV cars.

In [161…

```
# Bar Plot for Total Purchase Amount vs Gender and Make
```

```
sns.barplot(data=df_pcg, x='Gender', y='Total_Purchase_Amount', hue='Make')
plt.title('Fig 36: Total Purchase Amount vs Gender and Make')
plt.xlabel('Gender')
plt.ylabel('Total Purchase Amount')
plt.show()
```



Question 5: How much money was spent on purchasing automobiles by individuals who took a personal loan?

In [162…

```
df_pcp = df.groupby(['Gender','Personal_loan']).agg(Total_Purchase_Amount=('Price',
df_pcp
```

Out[162…

	Gender	Personal_loan	Total_Purchase_Amount
0	Female	No	8762000.0
1	Female	Yes	6933000.0
2	Male	No	20228000.0
3	Male	Yes	20357000.0

Total purchase amount for males who took personal loan is 20357000.0 USD.

Total purchase amount for females who took personal loan is 6933000.0 USD.

It seems female customers spent more amount on buying the cars when do not have personal loan. However male customers spent almost same amount on buying the cars whether they have personal loan or not.

In [163…
```
# Bar Plot for Total Purchase Amount vs Personal Loan and Gender

sns.barplot(data=df_pcp, x='Gender', y='Total_Purchase_Amount', hue='Personal_loan'
plt.title('Fig 37: Total Purchase Amount vs Personal Loan and Gender')
plt.xlabel('Personal Loan')
plt.ylabel('Total Purchase Amount')
plt.show()
```

Question 6: How does having a working partner influence the purchase of higher-priced cars?

In [164…
```
df_wph = df.groupby(['Partner_working','Make']).agg(Mean_Price_Amount=('Price','mea
df_wph
```

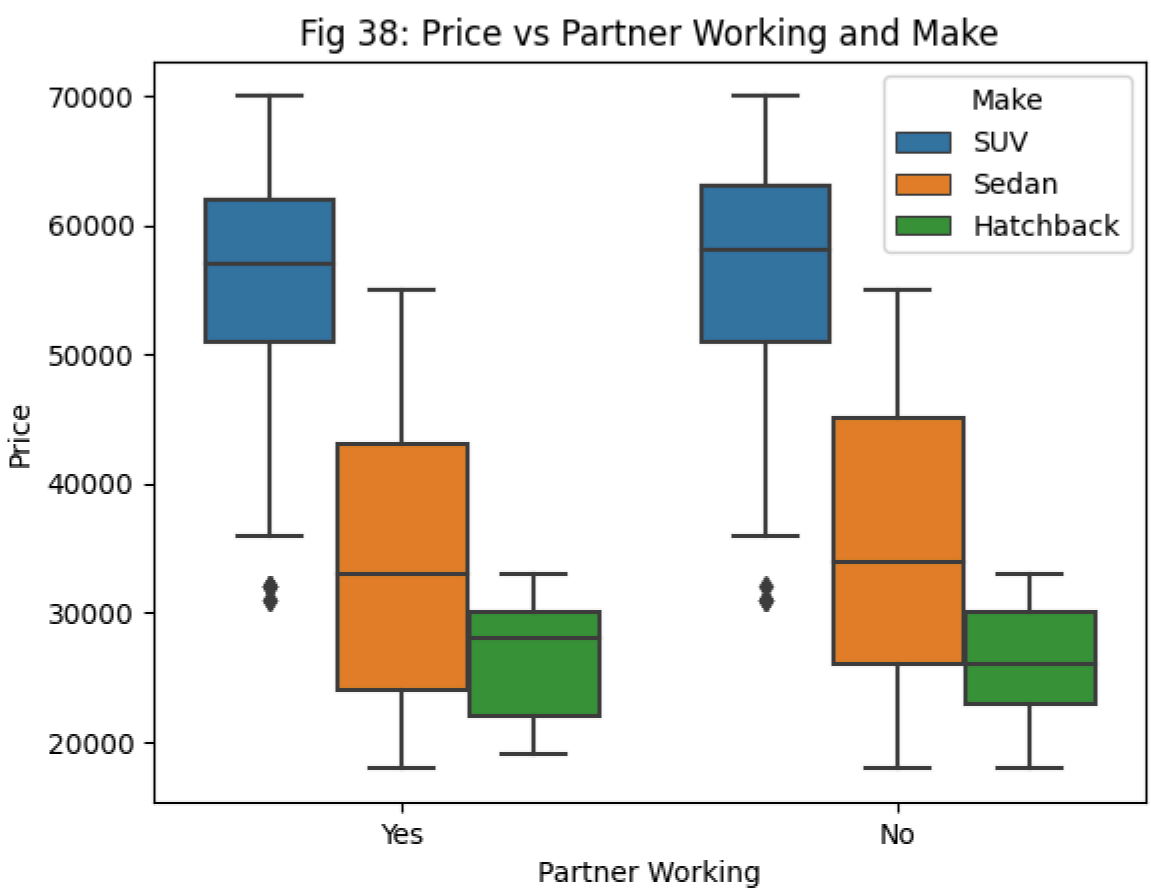
Out[164…

	Partner_working	Make	Mean_Price_Amount	Median_Price_Amount
0	No	Hatchback	26323.843416	26000.0
1	No	SUV	56173.611111	58000.0
2	No	Sedan	35354.166667	34000.0
3	Yes	Hatchback	26614.617940	28000.0
4	Yes	SUV	55496.732026	57000.0
5	Yes	Sedan	34082.125604	33000.0

Median price amount for SUV car is more than mean price amount when a customer has working partner. It means customers having working partner prefer to buy more expensive cars.

In [165…

```
# Box Plot for Price vs Partner Working and Make

sns.boxplot(data=df, x='Partner_working', y='Price', hue='Make')
plt.title('Fig 38: Price vs Partner Working and Make')
plt.xlabel('Partner Working')
plt.ylabel('Price')
plt.show()
```



Actionable Insights & Recommendations

Actionable Insights:

1. There are more male customers than female customers. There are more married customers than single customers.
2. Price of SUV car is highest followed by Sedan and Hatchback cars. Female customers are able to buy more expensive cars than male customers. Salaried class customers are able to buy more expensive cars than business class customers. Married customers are able to buy more expensive cars than single customers. Customers without house and personal loan are able to buy more expensive cars.
3. Customers with higher salary are able to buy SUV cars and customers with lesser salary are able to buy Sedan and Hatchback cars. Female customers are having higher salary than male customers. Customers without house land personal loan are having higher salary.
4. Customers with higher partner salary are able to buy SUV cars and customers with lesser partner salary are able to buy Sedan and Hatchback cars. Female customers are having higher partner salary than male customers.
5. Customers with higher total salary are able to buy SUV cars and customers with lesser total salary are able to buy Sedan and Hatchback cars. Female customers are having higher total salary than male customers. Married customers are having higher total salary than single customers. Customers without house and personal loan are having higher total salary.
6. Customers with higher age bought more SUV cars than Sedan and Hatchback cars.

Recommendations:

1. Austo Motor Company need to sell more SUV cars in comparison to Sedan and Hatchback cars to increase the revenue. New SUV car models can be launched to attract more customers.
2. Male customers can be targeted more for SUV car purchase in case it is their first car. Male customers can also be targeted more for SUV car purchase in case they want to buy second car or replace the existing car with new car.
3. Female customers can be targeted more for SUV car purchase in case it is their first car. Female customers can be targeted more for Sedan and Hatchback car purchase in case they want to buy second car.
4. Salaried and Business class customers can be targeted more for SUV and Hatchback car purchase (as second car) if they already have Sedan car.
5. Customers with higher salary, partner salary and total salary can be targeted more for SUV car purchase if it is their first car. Customers with higher salary, partner salary and total salary can be targeted more for Sedan and Hatchback car purchase if they already have SUV car.
6. Married and Single customers can be targeted more for SUV car purchase if it is their first car. Married and Single customers can be targeted more for Sedan and Hatchback

car purchase if they already have SUV car.

7. Customers without house and personal loan can be targeted more for SUV car purchase in case it is their first car. Customers without house and personal loan can be targeted more for Sedan and Hatchback car purchase if they already have SUV car.
8. Customers with higher age can be targeted more for SUV car purchase in case it is their first car. Customers with higher age can be targeted more for Sedan and Hatchback car purchase in case they already have SUV car.
