Performe 'Exploratory Data Analysis' on dataset 'Samplesuperstore' As a Bussiness Manager try to find out the weak areas where you can work to make more profit. What all bussiness problems you can derive by Explorating the data? I used python to perform EDA on this Dataset In [1]: **import** pandas **as** pd import numpy as np import seaborn as sns import matplotlib.pyplot as plt import warnings warnings.filterwarnings('ignore') In [2]: df=pd.read_csv("C:/Users/DELL/Downloads/SampleSuperstore.csv") Basic Data Insights In [3]: df.sample(5) Sales Quantity Out[3]: Ship Mode City State Postal Code Region Category Sub-Category Profit Segment Country Discount 6717 Standard Class Home Office United States Hot Springs Arkansas 71901 South Office Supplies Paper 25.920 0.0 12.7008 **2561** Standard Class 37211 Office Supplies 0.2 3.3408 Consumer United States Tennessee Labels 0.2 12.8151 **1559** Standard Class Corporate United States Seattle 98103 West Office Supplies Binders 35.352 Washington Office Supplies 12.672 0.2 1.4256 First Class Charlotte 28205 496 Standard Class West Office Supplies Binders 119.616 0.2 40.3704 Consumer United States Costa Mesa California In [4]: df.head() Ship Mode Segment Sales Quantity Discount **Profit** Out[4]: Country State Postal Code Region Category 41.9136 **0** Second Class Consumer United States 42420 Bookcases 261.9600 0.00 Henderson Kentucky South Furniture Second Class Consumer Henderson Kentucky 42420 South Furniture Chairs 731.9400 0.00 West Office Supplies 2 Second Class Corporate United States Los Angeles California 90036 Labels 14.6200 0.00 6.8714 3 Standard Class Consumer United States Fort Lauderdale 33311 -383.0310 4 Standard Class Consumer United States Fort Lauderdale Storage 22.3680 0.20 2.5164 Florida 33311 South Office Supplies df.tail() In [5]: **Sub-Category** Sales Quantity Discount Out[5]: Ship Mode Segment Country City State Postal Code Region Category Profit 0.2 4.1028 Second Class Consumer United States 33180 Furniture 25.248 9990 Standard Class Consumer United States Costa Mesa California 0.0 15.6332 92627 Furniture Furnishings 91.960 West 9991 Standard Class Consumer United States Costa Mesa 92627 Phones 258.576 0.2 19.3932 Technology 9992 Standard Class Consumer United States Costa Mesa California 0.0 13.3200 West Office Supplies Paper 29.600 9993 Second Class Consumer United States Westminster California Appliances 243.160 0.0 72.9480 West Office Supplies In [6]: df.shape Out[6]: (9994, 13) In [7]: df.info() <class 'pandas.core.frame.DataFrame'> RangeIndex: 9994 entries, 0 to 9993 Data columns (total 13 columns): # Column Non-Null Count Dtype 9994 non-null object O Ship Mode Segment 9994 non-null object Country 9994 non-null object City 9994 non-null object State 9994 non-null object Postal Code 9994 non-null int64 Region 9994 non-null object Category 9994 non-null object Sub-Category 9994 non-null object 9994 non-null float64 Sales 10 Quantity 9994 non-null int64 11 Discount 9994 non-null float64 12 Profit 9994 non-null float64 dtypes: float64(3), int64(2), object(8) memory usage: 1015.1+ KB In [8]: df.describe() Out[8]: **Postal Code** Profit Sales Quantity Discount **count** 9994.000000 9994.000000 9994.000000 9994.000000 9994.000000 0.156203 mean 55190.379428 229.858001 3.789574 28.656896 **std** 32063.693350 623.245101 2.225110 0.206452 234.260108 min 1040.000000 0.444000 1.000000 0.000000 -6599.978000 **25**% 23223.000000 17.280000 2.000000 0.000000 1.728750 **50%** 56430.500000 8.666500 54.490000 3.000000 0.200000 **75**% 90008.000000 209.940000 5.000000 0.200000 29.364000 max 99301.000000 22638.480000 14.000000 0.800000 8399.976000 Number of Unique values in each column In [9]: for i in df.columns: print(i,len(df[i].unique())) Ship Mode 4 Segment 3 Country 1 City 531 State 49 Postal Code 631 Region 4 Category 3 Sub-Category 17 Sales 5825 Quantity 14 Discount 12 Profit 7287 Check for null values In [10]: df.isnull().sum() Out[10]: Ship Mode Segment Country City State Postal Code Region Category Sub-Category Sales Quantity Discount Profit dtype: int64 Data Visualization In [11]: sns.pairplot(df) Out[11]: <seaborn.axisgrid.PairGrid at 0x1f4c4a6b370> 100000 -80000 Postal Code 60000 40000 20000 20000 15000 Sales 10000 5000 (B)(B) (B) (e))) **(((**(((O))(O OO $\mathbf{m}_{\mathbf{m}}$ 0.000 O)))(O •)))))) •) •(• • ((**•**)**)**)• 0.8 **OHO** 0.6 Discount **(1)** • • • • • • (O)D)((O)D) 0.0 - (0.00 00) 000 (000) 01(000) 01(000) 7500 5000 -2500-5000 50000 20000 0.00 0.75 -5000 100000 0 10000 10 0.25 0.50 5000 0 Postal Code Sales Quantity Discount Profit In [12]: fig, axes = plt.subplots(1,1,figsize=(12,7)) sns.heatmap(df.corr()) plt.show() - 1.0 - 0.8 - 0.6 - 0.4 - 0.2 - 0.0 Postal Code Sales Quantity Profit Discount In [13]: fig, axes=plt.subplots(1, 2, figsize=(14, 5)) sns.barplot(data=df.groupby ('Sub-Category')['Sales', 'Profit'].agg(sum), x='Sales', y='Profit', ax=axes[1]) df.groupby('Sub-Category')['Sales','Profit'].agg(sum).plot(kind='bar',ax=axes[0]) plt.xticks (rotation=90) plt.show() Total profit Vs sales Sales Profit 50000 300000 40000 250000 30000 200000 Profit 20000 150000 10000 100000 50000 -10000 -20000 Copiers Envelopes Fasteners Furnishings -Phones Storage Supplies Tables 27118.792 Binders 78479.206 189238.631 Chairs Machines Paper 149528.03 167380.318 Bookcases 114879.9963 107532.161000000001 Sub-Category In [14]: fig, axes=plt.subplots(1,2,figsize=(14,5)) fig.suptitle("Total Sales Vs Quantity") sns.barplot(data=df.groupby ('Sub-Category')['Sales','Quantity'].agg(sum),x='Sales',y='Quantity',ax=axes[1]) df.groupby('Sub-Category')['Sales', 'Quantity'].agg(sum).plot(kind='bar',ax=axes[0]) plt.xticks (rotation=90) plt.show() Total Sales Vs Quantity 6000 Sales Quantity 300000 5000 250000 4000 200000 150000 2000 100000 1000 50000 Binders Chairs 114879.9963 167380.318 Appliances Bookcases 107532.161000000001 189238.631 Sub-Category In [15]: fig, axes = plt.subplots(1,2,figsize=(14,5)) df.groupby('Sub-Category')['Profit','Quantity'].agg(sum).plot(kind='bar',ax=axes[0]).set_title('Quantity & Profit Relation based on sub_category') df.groupby('Sub-Category')['Profit','Discount'].agg(sum).plot(kind='bar',ax=axes[1]).set_title('Discount & Profit Relation based on sub_category') plt.xticks (rotation=90) plt.show() Quantity & Profit Relation based on sub_category Discount & Profit Relation based on sub_category Profit Profit Quantity Discount 50000 50000 40000 40000 30000 30000 20000 20000 10000 10000 -10000 -10000-20000 -20000 Storage -Supplies -Tables -Supplies -Tables -Machines Paper Phones Fasteners Labels Machines Binders Envelopes Fasteners Labels Binders Chairs Copiers Furnishings Chairs Copiers Accessories Accessories Appliances Paper Furnishings Envelopes **Sub-Category** Sub-Category In [16]: fig,axes= plt.subplots(2,2,figsize=(16,8)) fig.suptitle("Distribution plots", fontsize=16) sns.distplot(df['Sales'], ax=axes[0,0]) sns.distplot(df['Profit'], ax=axes[0,1]) sns.distplot(df['Discount'], ax=axes[1,0]) sns.distplot(df['Quantity'], ax=axes[1,1]) plt.show() Distribution plots 0.008 0.0025 0.007 0.006 0.0020 0.005 Density 0.0015 0.004 0.003 0.0010 0.002 0.0005 0.001 0.0000 0.000 5000 10000 15000 20000 -4000-2000 2000 4000 6000 8000 Profit Sales 25 0.8 20 0.6 10 0.2 0.0 12 0.2 0.4 0.6 0.8 10 14 Quantity Discount In [17]: fig,ax= plt.subplots(1,1,figsize=(12,7)) sns.countplot(data=df, x='Quantity', hue='Region') plt.show() 800 Region South West 700 Central East 600 500 400 · 300 200 100 Quantity In [18]: fig, axes = plt.subplots(2,2,figsize=(16,8)) fig.suptitle("Sales with different shipping modes and Segments", fontsize=16) sns.barplot(data=df,x="Ship Mode",y="Sales",ax=axes[0,0]) sns.lineplot(data=df, x="Ship Mode", y="Sales", ax=axes[0,1]) sns.barplot(data=df, x="Segment", y="Sales", ax=axes[1,0]) sns.lineplot(data=df, x="Segment", y="Sales", ax=axes[1,1]) plt.show() Sales with different shipping modes and Segments 300 280 250 260 200 Sales 240 29 150 100 220 50 200 Standard Class Standard Class Second Class First Class Same Day Second Class First Class Same Day Ship Mode Ship Mode 280 250 270 260 200 250 s es 240 100 230 220 50 210 Home Office Home Office Corporate Consumer Corporate Consumer Segment Segment Some Important Findings

GRIP: The Spark Foundation

Author: Shalaka Patil

Data Science And Bussiness Analytics Intern

Task 3: Exploratory Data Analysis- Retail

