

TÌNH HÌNH SỬ DỤNG DỊCH VỤ

Report by Hoang Trung

Tra Da Data Bank



Logo
Name

A. Tổng quan về dữ liệu được sử dụng để phân tích dữ liệu

Tình hình sử dụng dịch vụ của ngân hàng được phân tích dựa trên 3 bảng dữ liệu là:

- aum.csv (Dữ liệu cho biết về mã khách hàng và số dư tài khoản của từng khách hàng)

	customer_id	amount
0	999992	619,716,667
1	999994	53,121,165
2	1000017	0
3	1000018	66,783
4	1000021	77,664,164

- cust.csv (Dữ liệu cho biết về mã khách hàng, loại khách hàng và địa chỉ của khách hàng)

	customer_id	segment	province_city
0	1016846	Regular	HA NOI
1	1016868	Regular	HO CHI MINH
2	1016903	Regular	HO CHI MINH
3	1016936	Regular	HO CHI MINH
4	1016938	Regular	HA NOI

- prod_holding.csv (Dữ liệu cho biết về mã khách hàng và các loại dịch vụ mà khách hàng đang sử dụng tại ngân hàng)

	customer_id	prod_ca	prod_td	prod_credit_card	prod_app	prod_secured_loan	prod_upl
0	1016846	1	0	0	1	0	0
1	1016868	1	0	0	1	0	0
2	1016903	1	0	0	1	0	1
3	1016936	1	0	0	1	0	0
4	1016938	1	0	1	1	0	1

Trong dữ liệu nhận được thì ngân hàng đang cung cấp 6 dịch vụ cho khách hàng gồm có:

- prod_ca: Tài khoản thanh toán (Current Account)
- prod_td: Tiền gửi có kỳ hạn (Time Deposit)
- prod_credit_card: Thẻ tín dụng (Credit Card)
- prod_app: App của ngân hàng
- prod_secured_loan: Vay có tài sản thế chấp (Secured Loan)
- prod_upl: Vay cá nhân không có tài sản đảm bảo (Unsecured Personal Loan)

Để thuận lợi hơn cho việc phân tích dữ liệu. Ta sẽ tiến hành đổi tên cột cho dễ hiểu hơn trước khi tiến hành phân tích dữ liệu.

```
# Đổi tên các cột thành các tên có ý nghĩa hơn
df_product.rename(
    columns={
        'prod_ca': 'Tài khoản thanh toán',
        'prod_td': 'Tiền gửi có kỳ hạn',
        'prod_credit_card': 'Thẻ tín dụng',
        'prod_app': 'Sử dụng App ngân hàng',
        'prod_secured_loan': 'Vay có tài sản thế chấp',
        'prod_upl': 'Vay tín dụng cá nhân'
    },
    inplace=True
)

df_product
```

	customer_id	Tài khoản thanh toán	Tiền gửi có kỳ hạn	Thẻ tín dụng	Sử dụng App ngân hàng	Vay có tài sản thế chấp	Vay tín dụng cá nhân
0	1016846	1	0	0	1	0	0
1	1016868	1	0	0	1	0	0
2	1016903	1	0	0	1	0	1
3	1016936	1	0	0	1	0	0
4	1016938	1	0	1	1	0	1
...
113061	126523	1	1	1	1	0	0
113062	1284395	1	0	1	1	1	0
113063	1286275	1	0	0	1	0	0
113064	1298809	1	1	0	1	0	0
113065	1298427	1	1	0	1	0	1

Kiểm tra sơ bộ dữ liệu của chúng ta.

```
● df_amount = pd.read_csv('aum.csv')
```

```
df_amount.head()
```

	customer_id	amount
0	999992	619,716,667
1	999994	53,121,165
2	1000017	0
3	1000018	66,783
4	1000021	77,664,164

```
df_amount.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 113066 entries, 0 to 113065
Data columns (total 2 columns):
#   Column      Non-Null Count  Dtype
---  -
0   customer_id 113066 non-null int64
1   amount      113066 non-null float64
dtypes: float64(1), int64(1)
memory usage: 1.7 MB
```

Dữ liệu amount gồm 113.066 dòng và không có dữ liệu null. Kiểu dữ liệu cũng đã chính xác nên không cần phải chuẩn hoá nữa.

```
df_customer = pd.read_csv('cust.csv')
```

```
df_customer.head()
```

	customer_id	segment	province_city
0	1016846	Regular	HA NOI
1	1016868	Regular	HO CHI MINH
2	1016903	Regular	HO CHI MINH
3	1016936	Regular	HO CHI MINH
4	1016938	Regular	HA NOI

```
df_customer.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 113066 entries, 0 to 113065
Data columns (total 3 columns):
#   Column          Non-Null Count  Dtype
---  -
0   customer_id     113066 non-null int64
1   segment         113066 non-null object
2   province_city   113066 non-null object
dtypes: int64(1), object(2)
memory usage: 2.6+ MB
```

Điều tương tự cũng đúng với dữ liệu custom (gồm 113066 dòng dữ liệu và không có dữ liệu null)

```
df_product = pd.read_csv('prod_holding.csv')
```

```
df_product.head()
```

✓ 0.0s

	customer_id	prod_ca	prod_td	prod_credit_card	prod_app	prod_secured_loan	prod_upl
0	1016846	1	0	0	1	0	0
1	1016868	1	0	0	1	0	0
2	1016903	1	0	0	1	0	1
3	1016936	1	0	0	1	0	0
4	1016938	1	0	1	1	0	1

```
df_product.info()
```

✓ 0.0s

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 113066 entries, 0 to 113065
Data columns (total 7 columns):
#   Column                Non-Null Count  Dtype
---  -
0   customer_id           113066 non-null  int64
1   prod_ca               113066 non-null  int64
2   prod_td               113066 non-null  int64
3   prod_credit_card      112070 non-null  float64
4   prod_app              113066 non-null  int64
5   prod_secured_loan     113066 non-null  int64
6   prod_upl              113066 non-null  int64
dtypes: float64(1), int64(6)
memory usage: 6.0 MB
```

```
df_product.info()
```

✓ 0.0s

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 113066 entries, 0 to 113065
Data columns (total 7 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   customer_id                          113066 non-null  int64
1   Tài khoản thanh toán                 113066 non-null  int64
2   Tiền gửi có kỳ hạn                   113066 non-null  int64
3   Thẻ tín dụng                         112070 non-null  float64
4   Sử dụng App ngân hàng                113066 non-null  int64
5   Vay có tài sản thế chấp              113066 non-null  int64
6   Vay tín dụng cá nhân                 113066 non-null  int64
dtypes: float64(1), int64(6)
memory usage: 6.0 MB
```

Dữ liệu df_product có dữ liệu null ở cột thẻ tín dụng.

Ta sẽ tiến hành thay thế các dòng dữ liệu bị trống trong cột thẻ tín dụng này bằng các dữ liệu xuất hiện nhiều nhất trong cột Thẻ tín dụng (**tính mode**)

```
credit_card_mode = df_product['Thẻ tín dụng'].mode()[0]

df_product['Thẻ tín dụng'].fillna(value=credit_card_mode, inplace=True)

df_product.info()
```

✓ 0.0s

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 113066 entries, 0 to 113065
Data columns (total 5 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   customer_id                          113066 non-null  int64
1   Tiền gửi có kỳ hạn                   113066 non-null  int64
2   Thẻ tín dụng                         113066 non-null  float64
3   Vay có tài sản thế chấp              113066 non-null  int64
4   Vay tín dụng cá nhân                 113066 non-null  int64
dtypes: float64(1), int64(4)
memory usage: 4.3 MB
```

Đến đây tất cả dữ liệu của 3 Data Frame đã không còn giá trị null nào nữa.

Với dữ liệu của df_product thì chưa về dạng chuẩn. Vì thế ta sẽ tiến hành Unpivot dữ liệu này như sau.

```
# Ta sẽ tiến hành Unpivot dữ liệu trên trước khi tiến hành vẽ biểu đồ tình hình sử dụng các dịch vụ tại ngân hàng
df_product_unpivot = pd.melt(df_product, id_vars=['customer_id'], var_name='Sản phẩm', value_name='Sử dụng dịch vụ')
```

```
df_product_unpivot
```

✓ 0.0s

	customer_id	Sản phẩm	Sử dụng dịch vụ
0	1016846	Tiền gửi có kỳ hạn	0
1	1016868	Tiền gửi có kỳ hạn	0
2	1016903	Tiền gửi có kỳ hạn	0
3	1016936	Tiền gửi có kỳ hạn	0
4	1016938	Tiền gửi có kỳ hạn	0
...
452259	126523	Vay tín dụng cá nhân	0
452260	1284395	Vay tín dụng cá nhân	0
452261	1286275	Vay tín dụng cá nhân	0
452262	1298809	Vay tín dụng cá nhân	0
452263	1298427	Vay tín dụng cá nhân	1

452264 rows × 3 columns


```
# Lọc bỏ các giá trị 0 (không sử dụng dịch vụ)
df_product_unpivot = df_product_unpivot[df_product_unpivot['Sử dụng dịch vụ'] != 0]
```

```
df_product_unpivot
```

✓ 0.0s

	customer_id	Sản phẩm	Sử dụng dịch vụ	
	28	105425	Tiền gửi có kỳ hạn	1
	29	105426	Tiền gửi có kỳ hạn	1
	33	1557167	Tiền gửi có kỳ hạn	1
	39	1032411	Tiền gửi có kỳ hạn	1
	60	1032915	Tiền gửi có kỳ hạn	1

	448738	1053452	Vay tín dụng cá nhân	1
	449474	2238343	Vay tín dụng cá nhân	1
	449559	1068493	Vay tín dụng cá nhân	1
	449665	168881	Vay tín dụng cá nhân	1
	452263	1298427	Vay tín dụng cá nhân	1

39255 rows × 3 columns

Sau khi Unpivot dữ liệu thì ta sẽ loại bỏ các giá trị = 0 trong Data Frame df_product_unpivot mới được tạo này (Vì 0 là không sử dụng dịch vụ nên ta không cần quan tâm đến điều này)

B. Tổng quan về tình hình sử dụng dịch vụ tại ngân hàng

- Số lượng khách hàng tại ngân hàng: **113.066 khách hàng**
- Tổng số tiền gửi tại ngân hàng: 7.403.701.295.491 VND (Hơn 7.4 triệu tỷ VND)
- Top 5 khách hàng gửi nhiều tiền nhất tại ngân hàng

```
# Có những khách gửi tiền rất nhiều
df_amount.sort_values(by='amount', ascending=False).head()
```

✓ 0.0s

	customer_id	amount
7471	104565	132,500,000,000
10804	1074871	115,000,000,000
48652	1356804	104,166,666,667
35548	123361	81,248,531,017
99692	38828	79,132,290,616

- Số tiền gửi ít nhất tại ngân hàng là **0 VND**
- Số tiền gửi nhiều nhất tại ngân hàng là **132.500.000.000 VND**
- Ngân hàng tổng cộng có khách hàng tại 41 tỉnh thành. Trong đó 2 tỉnh thành có số lượng khách hàng chiếm đa số là TP Hồ Chí Minh và Hà Nội (Lần lượt chiếm 34% số lượng khách hàng trên cả nước). Dưới đây là thông tin về các tỉnh thành có số lượng khách hàng lớn nhất trên cả nước.

Chúng ta có thể thấy lượng khách hàng không có thông tin về địa chỉ cũng không quá lớn chỉ tầm 4%. Vì thế ta sẽ không xóa dữ liệu này đi mà cứ để lại để tiếp tục tiến hành phân tích.

```
df_customer_province = df_customer.groupby(['province_city']).agg(
    total_customers = ('segment', 'count')
).sort_values(by='total_customers', ascending=False).reset_index()

df_customer_province['% customers'] = df_customer_province['total_customers'] / sum(df_customer_province['total_customers']) * 100

df_customer_province.head()
```

✓ 0.0s

	province_city	total_customers	% customers
0	HA NOI	38153	34
1	HO CHI MINH	38048	34
2	No Info	5040	4
3	HAI PHONG	3161	3
4	THANH HOA	2051	2

- Khách hàng của ngân hàng được chia làm 3 hạng theo thứ tự lần lượt là Regular, Silver và Gold. Trong đó số lượng khách hàng của nhóm Regular (tức khách hàng thường) chiếm tỷ trọng lớn nhất (Khách hàng Regular chiếm tỷ lệ 81%, Silver là 16% còn Gold là 3%).

```
df_customer_segment = df_customer.groupby(['segment']).agg(
    total_customers = ('customer_id', 'count')
).reset_index()

df_customer_segment['% customers'] = df_customer_segment['total_customers'] / sum(df_customer_segment['total_customers']) * 100

df_customer_segment = df_customer_segment.sort_values(by='total_customers', ascending=False)

df_customer_segment
```

✓ 0.0s

	segment	total_customers	% customers
1	Regular	91166	81
2	Silver	18268	16
0	Gold	3632	3

- Về tình hình sử dụng dịch vụ tại ngân hàng. Chúng ta có thể theo dõi con số trong bảng dưới đây.

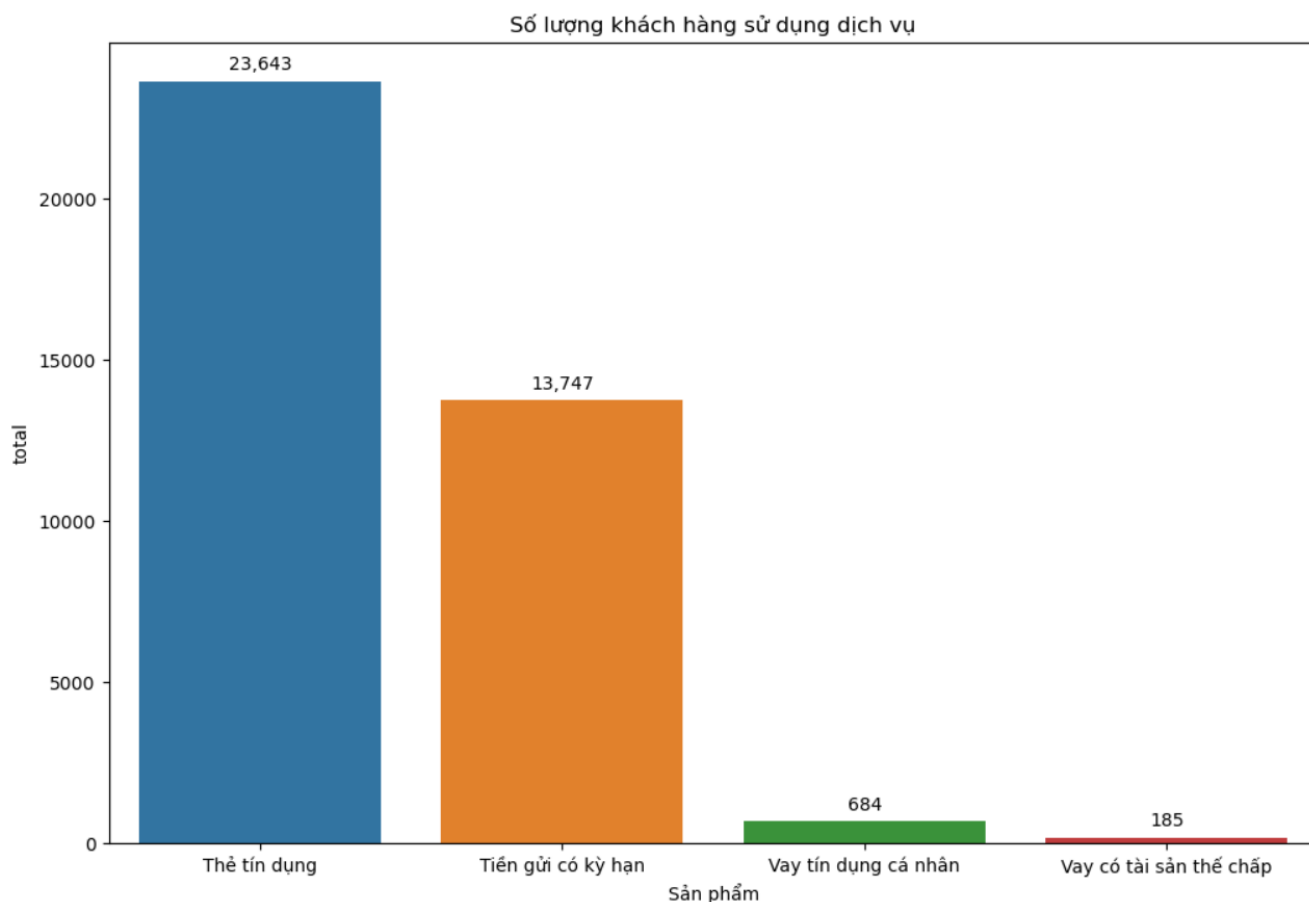
```
# vẽ biểu đồ xem mỗi sản phẩm có bao người sử dụng
df_product_groupby = df_product_unpivot.groupby(['Sản phẩm']).agg(
    total = ('sử dụng dịch vụ', 'count')
).sort_values(by='total', ascending=False).reset_index()

df_product_groupby['% sử dụng'] = round(df_product_groupby['total'] / len(df_customer) * 100, 2)

df_product_groupby
```

✓ 0.0s

	Sản phẩm	total	% sử dụng
0	Thẻ tín dụng	23643	21
1	Tiền gửi có kỳ hạn	13747	12
2	Vay tín dụng cá nhân	684	1
3	Vay có tài sản thế chấp	185	0



Ta có thể dễ dàng nhận thấy là % khách hàng sử dụng dịch vụ tại ngân hàng là chưa cao.

Ở đây mình sẽ chỉ tập trung vào các dịch vụ có thể đem lại nguồn thu cho ngân hàng đó là Thẻ tín dụng, Vay tín dụng cá nhân, Vay có tài sản thế chấp. Còn các dịch vụ khác như sử dụng App ngân hàng, Tài khoản thanh toán, Tiền gửi có kỳ hạn thì ta sẽ tạm thời không quan tâm trong bài phân tích này.

Nhìn 1 cách tổng quan thì dịch vụ đem lại doanh thu cho ngân hàng mà khách có sử dụng đạt tỷ lệ cao nhất là thẻ tín dụng chỉ đạt tỷ lệ là 21% (Khá thấp). Trong khi đó các dịch vụ mang lại nhiều doanh thu hơn cho ngân hàng là Vay tín dụng cá nhân và Vay có tài sản thế chấp tổng lại chưa tới 2%. Vì thế để cải thiện doanh thu thì chúng ta cần đi tìm hiểu tại sao lại có việc này và làm sao để cải thiện nó.

C. Phân tích dữ liệu tình hình sử dụng dịch vụ của khách hàng

1. Phân tích tình hình phát hành thẻ tín dụng tại ngân hàng

Bạn hãy xem kết quả được tính toán về tỷ lệ khách hàng có thẻ tín dụng theo từng Segment ở bảng dưới đây.

	segment	total_customers_dont_have_credit_cards	total_customers	% customers
0	Regular	77921	91166	85
1	Silver	8587	18268	47
2	Gold	1919	3632	53

Với kết quả tính toán được thì chúng ta có thể dễ dàng nhận ra là tỷ lệ khách hàng Silver và Gold chưa có thẻ tín dụng là rất lớn (Lần lượt là 47% với nhóm khách hàng Silver và 53% đối với nhóm khách hàng Gold)

```

df_total.groupby(['segment']).agg(
    average_amount = ('amount', 'mean')
).reset_index()

```

[181] ✓ 0.0s

...	segment	average_amount
0	Gold	1,589,434,830
1	Regular	9,411,988
2	Silver	42,304,616

Qua kiểm tra bạn có thể dễ dàng thấy được số tiền gửi trung bình của nhóm khách hàng Silver và Gold đều ở mức từ 42 triệu đồng trở lên. Vì thế không có lý do gì chúng ta không cấp cho 2 nhóm khách hàng này thẻ tín dụng cả.

Tiếp theo ta sẽ đi sâu vào nhóm khách hàng Regular. Mặc dù tỷ lệ tiền trong tài khoản của họ chỉ đạt 9.4 triệu nhưng có thực sự là tất cả người gửi tiền đều không có nhiều tiền hay không?

```
df_amount[(df_amount['customer_id'].isin(df_customer_regular_dont_have_credit_cards)) & (df_amount['amount'] >= 50000000)].sort_values(by='amount', ascending=False)
```

✓ 0.0s

	customer_id	amount
44629	131403	1,753,961,359
82780	247254	1,666,666,667
91866	284550	1,500,000,002
57037	1553077	1,250,994,228
34646	122886	1,185,841,072
...
46189	1329546	50,000,000
44361	1310671	50,000,000
81174	2442889	50,000,000
69758	2255465	50,000,000
25073	1168716	50,000,000

2867 rows x 2 columns

Trong số tổng cộng 77.921 khách hàng ở hạng Regular thì có 2.867 khách hàng có số tiền trong tài khoản tại ngân hàng là trên 50 triệu đồng và chưa có thẻ tín dụng. Thậm chí có cả những những khách có hơn 1 tỷ trong tài khoản nhưng vẫn chưa có thẻ tín dụng và vẫn ở hạng Regular.

Đây là 1 điều rất vô lý vì đây hoàn toàn là các khách hàng rất tiềm năng của ngân hàng.

Thông qua dữ liệu đã phân tích, ta rút ra kết luận như sau.

- Hãy cố gắng cấp phát hết thẻ tín dụng cho nhóm khách hàng Silver và Gold.
- Kiểm tra nhóm các khách hàng Regular có lượng tiền trong tài khoản lớn và tiến hành nâng hạng cho họ. Tiếp đó hãy cấp phát thẻ tín dụng có hạn mức cao cho họ.

Đây là 2 phương án chúng ta có thể xem xét dựa theo kết quả của việc phân tích dữ liệu.

2. Phân tích tình hình khách hàng với sản phẩm cho vay có tài sản thế chấp.

Đầu tiên ta sẽ xem xét xem có bao nhiêu khách hàng có số dư tài khoản trên 50 triệu đồng.

	customer_id	amount
7471	104565	132,500,000,000
10804	1074871	115,000,000,000
48652	1356804	104,166,666,667
35548	123361	81,248,531,017
99692	38828	79,132,290,616
...
87715	2555051	50,000,000
99765	3886721	50,000,000
44139	1308239	50,000,000
5726	1028870	50,000,000
49301	1365085	50,000,000
9824 rows × 2 columns		

Chúng ta có 9.824 khách hàng / 113.066 khách hàng có số dư tài khoản trên 50 triệu đồng (Chiếm 8.69% tổng số lượng khách hàng) mà chưa hề sử dụng dịch vụ vay có thể chấp tại ngân hàng.

Bạn có thể thấy đây là nhóm khách hàng có thể tiến hành gợi ý họ các gói vay có tài sản thế chấp như mua xe máy, mua điện thoại, mua ô tô, xây nhà, ... với lãi suất ưu đãi.

Đa phần hiện tại gần như khách hàng nào sử dụng dịch vụ của ngân hàng thì đều cài App của ngân hàng.

Bạn có thể hiện ra các quảng cáo về dịch vụ trả góp lãi suất thấp hoặc lãi 0 đồng chẳng hạn khi khách hàng tiến hành mua sắm các sản phẩm có giá trị vài chục triệu, chẳng hạn như mua tivi, tủ lạnh, xe máy, ...

Ngoài ra với các nhóm khách hàng này thì bạn có thể cho nhân viên ngân hàng gọi điện tới họ và tư vấn họ về các khoản vay có lãi suất thấp chẳng hạn.

3. Phân tích tình hình sử dụng dịch vụ theo tỉnh thành

	province_city	total_customers	total_amount
0	HO CHI MINH	5331	4,353,079,795,512
1	HA NOI	1347	1,166,663,527,274
2	HAI PHONG	363	168,296,982,602
3	NAM DINH	349	270,086,862,352
4	THUA THIEN HUE	282	90,385,594,872
5	NGHE AN	213	82,126,080,333
6	THANH HOA	188	96,885,106,392
7	BAC GIANG	182	104,463,216,842
8	DA NANG	150	87,927,318,915
9	HOA BINH	139	66,227,669,199
10	No Info	136	29,393,235,135
11	THAI BINH	127	56,533,844,810
12	HA TINH	116	37,185,711,889
13	QUANG NINH	101	54,852,059,326
14	HAI DUONG	91	51,972,147,936
15	PHU THO	87	50,039,135,553
16	VINH PHUC	86	57,313,837,692
17	QUANG BINH	73	67,368,139,037
18	THAI NGUYEN	56	47,459,620,105
19	NHA TRANG	45	14,121,706,548
20	BAC NINH	45	71,163,095,666
21	CAN THO	40	25,753,921,013

Các bạn hãy cùng xem kết quả được tính toán ở bảng trên.

Nếu nhìn vào các bạn có thể thấy lượng khách hàng của ngân hàng và cả lượng tiền gửi đều tập trung vào 2 thành phố lớn là TP Hồ Chí Minh và Hà Nội.

Tuy nhiên nếu tính toán số tiền gửi tính trên đầu người và sắp xếp lại theo thứ tự giảm dần thì ta sẽ có bảng sau.

	province_city	total_customers	total_amount	% customers	% amount	amount / customer
0	BAC NINH	45	71,163,095,666	0	1	1,581,402,126
1	VUNG TAU	21	21,254,849,030	0	0	1,012,135,668
2	GIA LAI	7	7,044,867,050	0	0	1,006,409,579
3	QUANG BINH	73	67,368,139,037	1	1	922,851,220
4	HA NOI	1347	1,166,663,527,274	14	16	866,119,916
5	THAI NGUYEN	56	47,459,620,105	1	1	847,493,216
6	HO CHI MINH	5331	4,353,079,795,512	54	61	816,559,707
7	BINH THUAN	30	24,363,623,039	0	0	812,120,768
8	NAM DINH	349	270,086,862,352	4	4	773,887,858
9	LONG AN	29	21,713,230,624	0	0	748,732,090
10	VINH PHUC	86	57,313,837,692	1	1	666,439,973
11	CAN THO	40	25,753,921,013	0	0	643,848,025
12	DA NANG	150	87,927,318,915	2	1	586,182,126
13	PHU THO	87	50,039,135,553	1	1	575,162,478
14	BAC GIANG	182	104,463,216,842	2	1	573,973,719
15	HAI DUONG	91	51,972,147,936	1	1	571,122,505
16	DA LAT	3	1,636,166,067	0	0	545,388,689
17	QUANG NINH	101	54,852,059,326	1	1	543,089,696
18	THANH HOA	188	96,885,106,392	2	1	515,346,311
19	BINH DUONG	18	8,966,563,222	0	0	498,142,401
20	BINH DINH	30	14,414,174,794	0	0	480,472,493

Chúng ta sẽ tập trung vào Top 20 tỉnh thành có lượng tiền gửi tính trên đầu người cao nhất. Dựa theo con số về GDP thu thập được của Việt Nam thì Top 20 tỉnh thành có GDP cao nhất là.

STT	Tên tỉnh, thành phố	Tổng GRDP (tỉ VNĐ)	Tổng GRDP (tỉ USD)
1	Thành phố Hồ Chí Minh	1.479.227	63,65
2	Thủ đô Hà Nội	1.196.004	51,39
3	Bình Dương	459.041	19,28
4	Đồng Nai	434.990	18,35
5	Bà Rịa – Vũng Tàu	390.293	16,79
6	Hải Phòng	365.585	15,97
7	Quảng Ninh	269.244	11,55
8	Thanh Hóa	252.672	10,91
9	Bắc Ninh	243.032	11,11
10	Nghệ An	175.586,80	8,01
11	Hải Dương	169.179	7,36
12	Long An	156.357	6,74
13	Bắc Giang	155.876	6,68
14	Vĩnh Phúc	153.121	6,62
15	Thái Nguyên	142.950	6,43
16	Hung Yên	131.997	5,72
17	Đà Nẵng	125.219	5,42
18	Quảng Ngãi	121.342,17	5,29
19	Quảng Nam	116.374	5,06
20	Kiên Giang	116.042	5,05

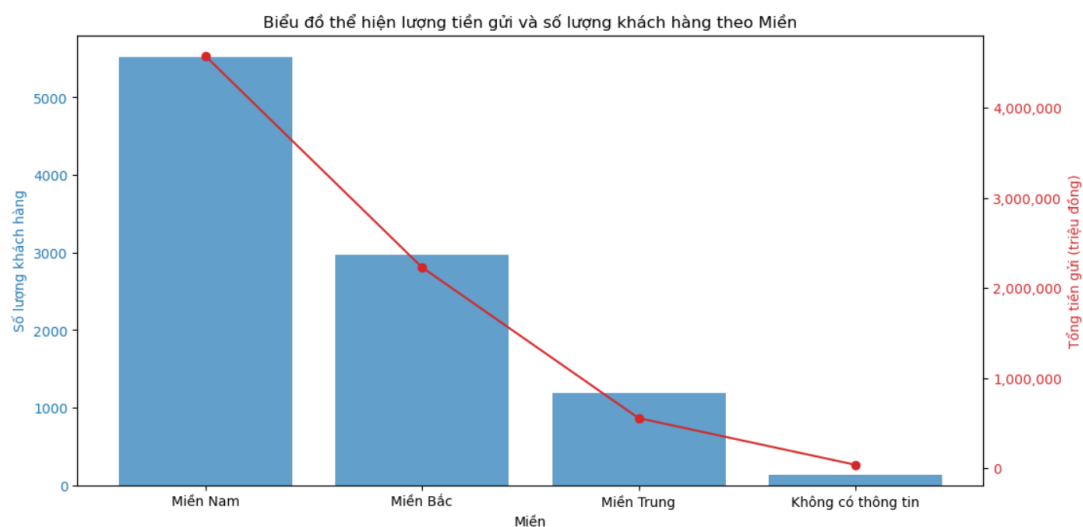
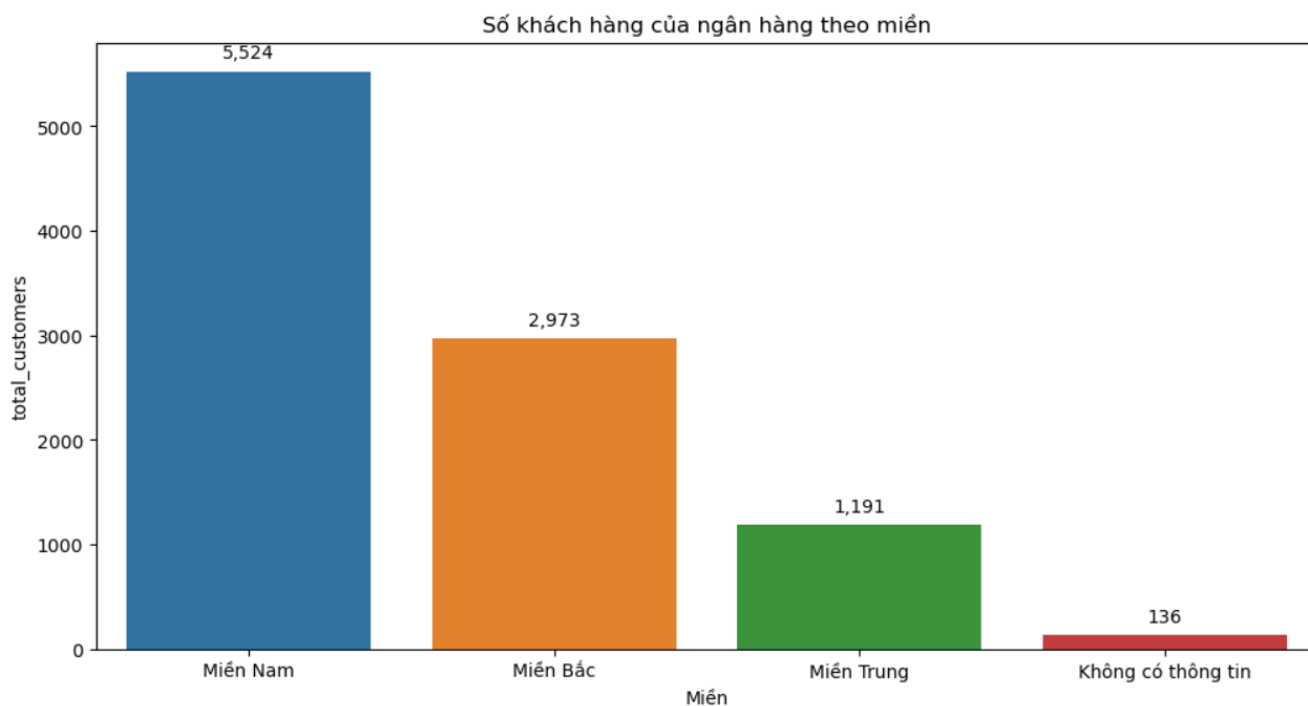
Trong số này chúng ta có thể nhặt ra được các tỉnh thành sau là đối tượng có lượng tiền gửi tính trên đầu người cao, ngoài ra GDP của tỉnh thành đó cũng cao nữa.

Các tỉnh thành đó là:

- Bắc Ninh
- Vũng Tàu
- Hà Nội
- Thái Nguyên
- Hồ Chí Minh
- Long An

Ngoài ra nếu chúng ta phân nhóm các tỉnh thành theo 3 miền Bắc, Trung Nam thì ta sẽ có con số tính toán như sau.

	Miền	total_customers	% total customers
0	Miền Nam	5524	56
1	Miền Bắc	2973	30
2	Miền Trung	1191	12
3	Không có thông tin	136	1



Nhìn vào đây có thể thấy khách hàng của ngân hàng chủ yếu nằm ở thị trường Miền Nam. Thị trường Miền Trung tuy có khách hàng nhưng còn khá ít.

Đây có thể là 1 thị trường tiềm năng mà ngân hàng chưa quá quan tâm nên số lượng khách hàng còn ít.

Vì thế nếu muốn phát triển thì chúng ta nên chọn thị trường này thay vì các thị trường miền Nam và miền Bắc đã có quá nhiều khách hàng rồi.

D. Kết luận

Tổng kết lại để có thể đáp ứng được bài toán tăng trưởng 20% doanh thu của doanh nghiệp thì chúng ta cần phải làm các công việc sau.

- Phát hành toàn bộ thẻ tín dụng cho các khách hàng Silver và Gold của ngân hàng.
- Kiểm tra nhóm khách hàng Regular và tiến hành nâng hạng cho các khách hàng có số dư tiền trong tài khoản tương đối nhiều.
- Quảng cáo trên App ngân hàng với nhóm các khách hàng có nhiều tiền trong tài khoản, thường sẽ liên quan đến các hoạt động vay tín dụng để tiêu dùng cá nhân khi khách hàng tiêu dùng mua sắm.
- Tập trung phát triển và thu hút khách hàng Miền Trung (Miền đang có ít khách hàng nhất doanh nghiệp). Ngoài ra tiếp tục tăng cường cho khách hàng Miền Bắc, Miền Nam sử dụng nhiều dịch vụ khác như thẻ tín dụng, vay có tài sản thế chấp, vay không có tài sản đảm bảo.