In [1]:
```
from ucimlrepo import fetch_ucirepo

# fetch dataset
wholesale_customers = fetch_ucirepo(id=292)

# data (as pandas dataframes)
X = wholesale_customers.data.features
y = wholesale_customers.data.targets

# metadata
print(wholesale_customers.metadata)

# variable information
print(wholesale_customers.variables)
```

```
{'uci_id': 292, 'name': 'Wholesale customers', 'repository_url': 'https://archive.ic
s.uci.edu/dataset/292/wholesale+customers', 'data_url': 'https://archive.ics.uci.edu/
static/public/292/data.csv', 'abstract': 'The data set refers to clients of a wholesa
le distributor. It includes the annual spending in monetary units (m.u.) on diverse p
roduct categories', 'area': 'Business', 'tasks': ['Classification', 'Clustering'], 'c
haracteristics': ['Multivariate'], 'num_instances': 440, 'num_features': 7, 'feature_
types': ['Integer'], 'demographics': [], 'target_col': ['Region'], 'index_col': None,
'has_missing_values': 'no', 'missing_values_symbol': None, 'year_of_dataset_creatio
n': 2013, 'last_updated': 'Mon Feb 05 2024', 'dataset_doi': '10.24432/C5030X', 'creat
ors': ['Margarida Cardoso'], 'intro_paper': None, 'additional_info': {'summary': Non
e, 'purpose': None, 'funded_by': None, 'instances_represent': None, 'recommended_data
_splits': None, 'sensitive_data': None, 'preprocessing_description': None, 'variable_
info': '1)\tFRESH: annual spending (m.u.) on fresh products (Continuous);\r\n2)\tMIL
K: annual spending (m.u.) on milk products (Continuous);\r\n3)\tGROCERY: annual spend
ing (m.u.)on grocery products (Continuous);\r\n4)\tFROZEN: annual spending (m.u.)on f
rozen products (Continuous)\r\n5)\tDETERGENTS_PAPER: annual spending (m.u.) on deterg
ents and paper products (Continuous) \r\n6)\tDELICATESSEN: annual spending (m.u.)on a
nd delicatessen products (Continuous); \r\n7)\tCHANNEL: customersâ€™ Channel - Horeca
(Hotel/Restaurant/CafÃ©) or Retail channel (Nominal)\r\n8)\tREGION: customersâ€™ Regi
on â€“ Lisnon, Oporto or Other (Nominal)\r\nDescriptive Statistics:\r\n\r\n\t(Minimu
m, Maximum, Mean, Std. Deviation)\r\nFRESH (\t3, 112151, 12000.30, 12647.329)\r\nMILK
\t(55, 73498, 5796.27, 7380.377)\r\nGROCERY\t(3, 92780, 7951.28, 9503.163)\r\nFROZEN
\t(25, 60869, 3071.93, 4854.673)\r\nDETERGENTS_PAPER (3, 40827, 2881.49, 4767.854)\r
\nDELICATESSEN (3, 47943, 1524.87, 2820.106)\r\n\r\nREGION\tFrequency\r\nLisbon\t77\r
\nOporto\t47\r\nOther Region\t316\r\nTotal\t440\r\n\r\nCHANNEL\tFrequency\r\nHoreca\t
298\r\nRetail\t142\r\nTotal\t440\r\n', 'citation': None}}
              name     role         type demographic description units  \
0          Channel  Feature  Categorical        None        None  None
1           Region   Target  Categorical        None        None  None
2            Fresh  Feature      Integer        None        None  None
3             Milk  Feature      Integer        None        None  None
4          Grocery  Feature      Integer        None        None  None
5           Frozen  Feature      Integer        None        None  None
6  Detergents_Paper  Feature      Integer        None        None  None
7        Delicassen  Feature      Integer        None        None  None

  missing_values
0             no
1             no
2             no
3             no
4             no
5             no
6             no
7             no
```

In [2]:
```
X
```

Out[2]:

	Channel	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Delicassen
0	2	12669	9656	7561	214	2674	1338
1	2	7057	9810	9568	1762	3293	1776
2	2	6353	8808	7684	2405	3516	7844
3	1	13265	1196	4221	6404	507	1788
4	2	22615	5410	7198	3915	1777	5185
...	...	...	...	...	...	...	...
435	1	29703	12051	16027	13135	182	2204
436	1	39228	1431	764	4510	93	2346
437	2	14531	15488	30243	437	14841	1867
438	1	10290	1981	2232	1038	168	2125
439	1	2787	1698	2510	65	477	52

440 rows × 7 columns

In [4]:
```
y
```

Out[4]:

	Region
0	3
1	3
2	3
3	3
4	3
...	...
435	3
436	3
437	3
438	3
439	3

440 rows × 1 columns

In [5]:
```
X.isnull().sum()
```

Out[5]:
```
Channel             0
Fresh               0
Milk                0
Grocery             0
Frozen              0
Detergents_Paper    0
Delicassen          0
dtype: int64
```

In [6]:
```
X.describe()
```

Out[6]:

	Channel	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Del
count	440.000000	440.000000	440.000000	440.000000	440.000000	440.000000	440.
mean	1.322727	12000.297727	5796.265909	7951.277273	3071.931818	2881.493182	1524.
std	0.468052	12647.328865	7380.377175	9503.162829	4854.673333	4767.854448	2820.
min	1.000000	3.000000	55.000000	3.000000	25.000000	3.000000	3.
25%	1.000000	3127.750000	1533.000000	2153.000000	742.250000	256.750000	408.
50%	1.000000	8504.000000	3627.000000	4755.500000	1526.000000	816.500000	965.
75%	2.000000	16933.750000	7190.250000	10655.750000	3554.250000	3922.000000	1820.
max	2.000000	112151.000000	73498.000000	92780.000000	60869.000000	40827.000000	47943.

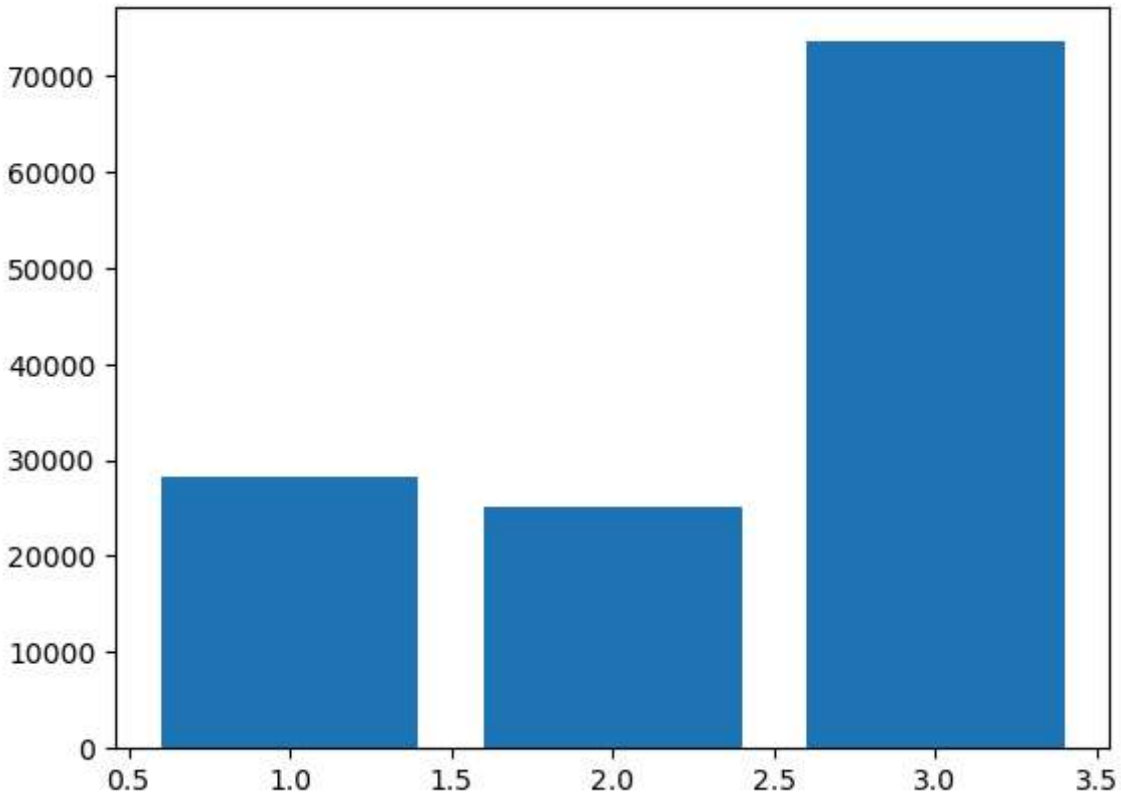
In [7]:
```
X.head()
```

Out[7]:

	Channel	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Delicassen
0	2	12669	9656	7561	214	2674	1338
1	2	7057	9810	9568	1762	3293	1776
2	2	6353	8808	7684	2405	3516	7844
3	1	13265	1196	4221	6404	507	1788
4	2	22615	5410	7198	3915	1777	5185

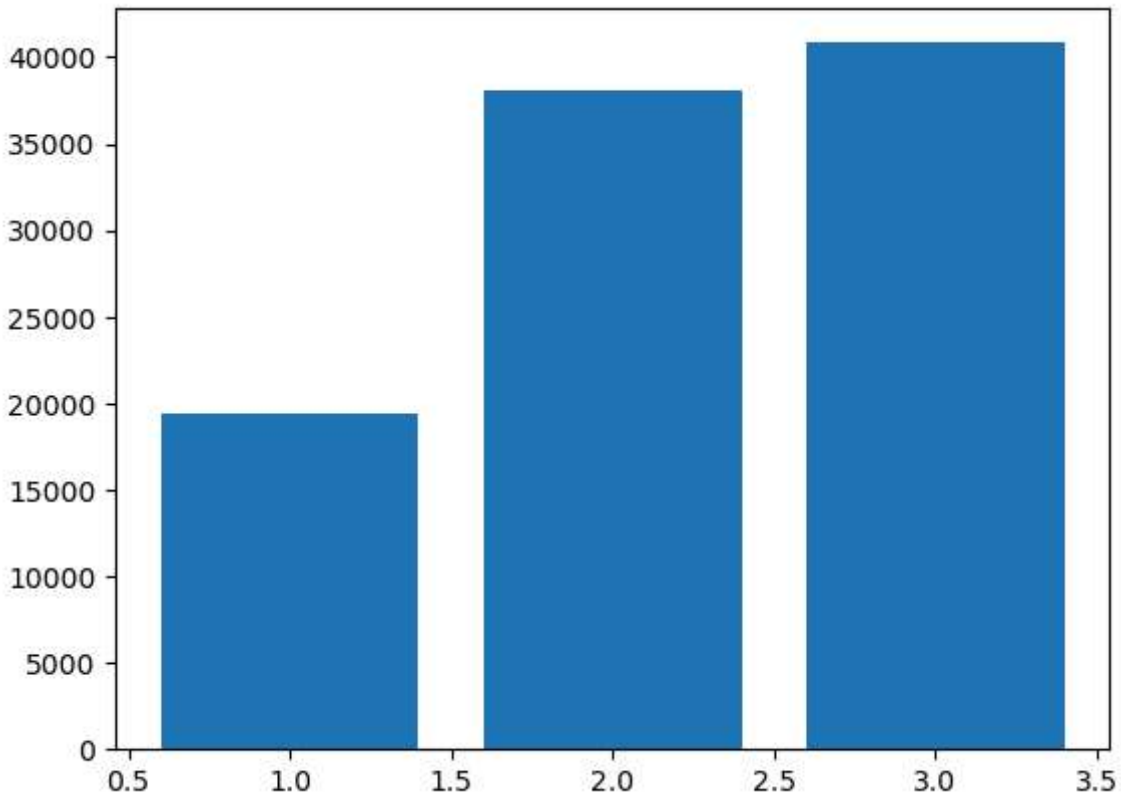
In [18]:
```
plt.bar(y.Region,X.Milk)
```

Out[18]:
```
<BarContainer object of 440 artists>
```



In [20]:
```
plt.bar(y.Region,X.Detergents_Paper)
```

Out[20]:
```
<BarContainer object of 440 artists>
```



In [ ]: