Data description

Description of data

Data was extracted from the expense claim system by our operations team and given to me in .xls file.

The data contained the following information:

- Name of the person who submitted the claim;
- Date of submission;
- Department code of the person;
- Amount of the claim (in OMR);

To prepare the data for further analysis I decided to include additional information and change some of the data:

- I have changed the date of submission so it only contains the month;
- Added a flag to indicate whether the person is a manager or not;
- Added a flag to indicate whether the person is a practitioner or not;
- Changed the department code to the name of the department for easier analysis;
- Removed the person name from the analysis;

Once I completed the above-mentioned modifications and added new data from various other data sources, I have saved the file in a comma-separated values (.csv) file.

This .csv file is the one I used during my analysis.

Remark

Usually the claim is submitted every month and it should contain the claimed value for the previous month. It happens occasionally that some people combine multiple months into a single reimbursement record and submit it all-together.

Although each month would show up as a separate line in the expense claim, since I did not have direct access to the expense claim system I was not able to verify whether the .xls – that was given to me by

our operations team – showed these cases correctly or not (e.g. each and every claimed amount was assigned to the relevant month or they were combined and the month was picked randomly by operation).

I did not consider the above to create a huge distortion in the data and further analysis supported my theory about the same.

Data analysis

Once I loaded the .csv file into a data frame in my Notebook I have carried out a couple of basic analytical steps to better understand the nature of the data and the approach that can be used to solve the business problem.

The file has 5 columns and contains 591 records:

```
In [3]: df.head()
```

Out[3]:

	MANAGER	PRACTITIONER	MONTH	AMOUNT	LOB
0	0	0	11	29.0	MARKETING
1	0	0	10	29.0	MARKETING
2	0	0	12	28.0	MARKETING
3	0	1	1	19.0	PFS
4	0	1	2	19.0	PFS

```
In [49]: df.shape
Out[49]: (591, 5)
```

Illustration 1: Data frame head and shape

The columns have the following type:

```
In [5]: df.dtypes
Out[5]: MANAGER           int64
        PRACTITIONER      int64
        MONTH             int64
        AMOUNT          float64
        LOB              object
        dtype: object
```

Illustration 2: Data frame column data types

Basic analysis of the data frame shows that the average (mean) reimbursement amount is ~38 INR:

```
In [7]: df.describe()
```

Out[7]:

	MANAGER	PRACTITIONER	MONTH	AMOUNT
count	591.000000	591.000000	591.000000	591.000000
mean	0.179357	0.389171	6.084602	37.700623
std	0.383976	0.487975	3.385855	28.238504
min	0.000000	0.000000	1.000000	2.620000
25%	0.000000	0.000000	3.000000	21.615000
50%	0.000000	0.000000	6.000000	29.852000
75%	0.000000	1.000000	9.000000	43.732000
max	1.000000	1.000000	12.000000	176.742667

Illustration 3: Basic analysis of the data frame

The basic analysis also showed that the data is equally distributed across the time period (it contained data from 2018 and early 2019).

Let us do a box plot to see whether our data supports this theory:

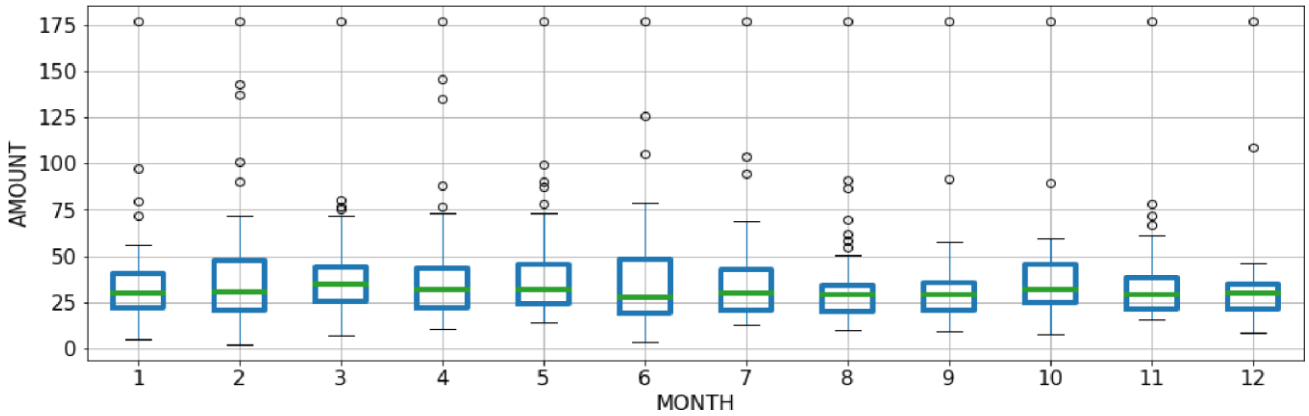

Illustration 4: Box plot showing the claimed amounts grouped by month

As you can see although there are outliers in each month the median for the quartiles are very close to each other. This suggests that the actual season or quarter does not have any impact on the claimed amount.

Let us do a group by on the data set to see it in numbers:

```
In [21]: df.groupby('MONTH').mean()
```

Out[21]:

MONTH	MANAGER	PRACTITIONER	AMOUNT
1	0.169492	0.389831	35.206000
2	0.188679	0.358491	40.680340
3	0.196429	0.321429	39.666571
4	0.218182	0.363636	40.595582
5	0.200000	0.380000	41.774220
6	0.196078	0.352941	37.458667
7	0.160000	0.380000	38.093673
8	0.166667	0.395833	34.781118
9	0.173913	0.391304	33.334080
10	0.170213	0.404255	37.530312
11	0.139535	0.465116	36.019504
12	0.151515	0.545455	35.586657

Illustration 5: Data frame grouped by MONTH and aggregated

None of the months seems to deviate too much from the average monthly claim (as we have seen above it is around ~38 OMR).

All-right, as a last basic data analysis let us see how the amounts are distributed. The following histogram shows that most of the amounts are between 15 and 35 OMR.

```
In [50]: hist = df['AMOUNT'].hist(bins=25)
```

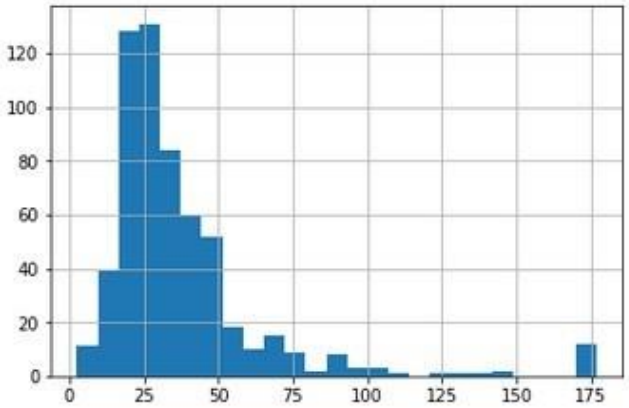

Illustration 6: Histogram of amounts claimed