Experiment 7

Aim: Analyse Financial Balance Sheet, Cash flows and Profit & Loss.
Balance Sheet Items Forecast using Rebates Payable estimation

Theory: The balance sheet and the profit and loss (P&L) statement are two of the three financial statements companies issue regularly. Such statements provide an ongoing record of a company's financial condition and are used by creditors, market analysts and investors to evaluate a company's financial soundness and growth potential. The third financial statement is called the cash-flow statement.

Balance Sheet
A balance sheet reports a company's assets, liabilities and shareholder equity at a specific point in time. It provides a basis for computing rates of return and evaluating the company's capital structure. This financial statement provides a snapshot of what a company owns and owes, as well as the amount invested by shareholders.

The balance sheet shows a company's resources or assets, and it also shows how those assets are financed—whether through debt under liabilities or by issuing equity as shown in shareholder equity. The balance sheet provides both investors and creditors with a snapshot of how effectively a company's management uses its resources. Just like the other financial statements, the balance sheet is used to conduct financial analysis and to calculate financial ratios. Below are a few examples of the items on a typical balance sheet.

Profit and Loss (P&L) Statement
A P&L statement, often referred to as the income statement, is a financial statement that summarizes the revenues, costs, and expenses incurred during a specific period of time, usually a fiscal year or quarter. These records provide information about a company's ability (or lack thereof) to generate profit by increasing revenue, reducing costs, or both. The P&L statement's many monikers include the "statement of profit and loss," the "statement of operations," the "statement of financial results," and the "income and expense statement."

Balance Sheet Items Forecast using Rebates Payable estimation produces diverse paper which it sales to retailers, publishers and commercial printers both through its own sales team and through the external agents. External agents have an advantage of being paid rebates periodically. Amounts of rebates are significant and being paid periodically (monthly, quarterly, etc.) make sufficient accrualed balances at the end of each period. So it perfectly make sense to estimate those balances of accrued and unpaid rebates for the purpose of cash flow management.

Use regression analysis over historical data in order to estimate rebates balances in the future.

Lab Experiment to be done by students:

1. Import Python Library package Statsmodels which helps to explore data, estimate statistical models, and perform statistical tests.
2. Read historical data on rebates payable and sales
3. Since the size of commission depends from sales revenue. Prepare data series and calculate correlation between rebate and sales.
4. Improve the model in order to increase the predictive power of the model to test different lagging for commission balance against sales and see which will return highest correlation coefficient.
5. Generate datasets with rebates payable lagged and put all datasets in a list
6. Finding the lag providing the best result
7. Brining additional variables to improve the predictive power of the model by implementing Multiple Regression
8. Split Dataset with coefficients for prediction of rebates balances in the future.
9. Compare predicted balances with actuals.

Sarvagya singh
600009200030 -- K1
Lab 7 - FMC

Balance Sheet Items Forecast

using Rebates Payable estimation as example

[Catalyst Paper](#) produces diverse paper which it sales to retailers, publishers and commercial printers both through it own sales team and through the external agents. External agents have an advantage of being paid rebates periodically. Amounts of rebates are significant and being paid periodically (monthly, quarterly, etc.) make sufficient accrualed balances at the end of each period. So it perfectly make sence to estimate those balances of accrued and unpaid rebates for the purpose of cash flow management.

We will use regression analysis over historical data in order to estimate rebates balances in the future.

And before we start I should mention all data used in this excercised was modified from original one, represents no actual data for any actual company and used for demonstration purposes only.

Starting with setting up the working environment for the job. Import neccessary libraries first.

In [ ]:

```
import pandas as pd
import numpy as np
#import statsmodels.api as sm
import statsmodels.formula.api as smf
import matplotlib.pyplot as plt
```

Obviously the size of commission depends from sales revenue. So, first let us see what is the correlation between those two.

In [ ]:

```
# Read historical data on rebates payable and sales
rebates_payable = pd.read_csv("rebates_payable.csv")
sales = pd.read_csv("sales.csv")
data = pd.merge(rebates_payable, sales, how='inner', on = 'Period')

# Prepare data series and calculate correlation between them
data['Sales'] = data['Sales'].astype(float)
data['Rebates_Payable'] = data['Rebates_Payable'].astype(float)

print('The correlation between Rebates Payable and Sales is ' +
      str(round(np.corrcoef(data['Rebates_Payable'], data['Sales'])[1,0],4)))
```

```
The correlation between Rebates Payable and Sales is 0.3219
```

Good to start with. Let us see how we can improve the model in order to increas the predictive power of the model. If we think about that.. Being accrued for different periods like month, quarter, year, rebates got paid with different delay. Let us test different lagging for commision balance against sales and see which will return highest correlation coefficient.

In [ ]:

```
# generate datasets with rebates payable lagged and put all datasets in a list
alldata =[data]
for lag in [1,2,3,4]:
    data_lagged = pd.concat([data.drop(columns=['Sales']), data['Sales'].shift(lag)], ax
is = 1).dropna()
```

```
    alldata.append(data_lagged)
```

In [ ]:

```
# finding the lag providing the best result
best_result = np.corrcoef(alldata[0]['Rebates_Payable'],alldata[0]['Sales'])[1,0]
lagging = 0

for dataset in alldata:
    print(str(lagging) + 'months lag: correlation coefficient = ' +
          str(round(np.corrcoef(dataset['Rebates_Payable'],dataset['Sales'])[1,0],4)))
    lagging = lagging + 1

    if best_result < np.corrcoef(dataset['Rebates_Payable'],dataset['Sales'])[1,0]:
        best_result = np.corrcoef(dataset['Rebates_Payable'],dataset['Sales'])[1,0]
        bestlagging = lagging - 1

data = alldata[bestlagging]

print('Hirest correlation of ' + str(round(best_result,4)) + ' returns dataset with '+ s
tr(bestlagging) + ' months lagging')
```

```
0months lag: correlation coefficient = 0.3219
1months lag: correlation coefficient = 0.4509
2months lag: correlation coefficient = 0.47
3months lag: correlation coefficient = 0.3211
4months lag: correlation coefficient = 0.1763
Hirest correlation of 0.47 returns dataset with 2 months lagging
```

Lagging of one variable against another significantly improved the model and increased correlation coefficient. We can further improve the predictive power of the model by brining additional variables to the consideration.

Brining additional variables - Multiple Regression

We could create a number of features and test its correlation against the raw data. However, it is alway good to ask a knowledgebale person. So instead of going purely statistical way we made a shortcut speaking with the subject mater expert from accounts payable department. She was kind describing the rebates progam to us and we learned there are three levels of rebates. Monthly paid say immediately after the end of the month. Quarterly paid at the month following the end of the quarter. Finaly, the annual ones paid in March following the reporting year.

In [ ]:

```
Quarterly_payments = {'Jan': True, 'Feb': False, 'Mar': False, 'Apr': True, 'May': False
, 'Jun': False,
                      'Jul': True, 'Aug': False, 'Sep': False, 'Oct': True, 'Nov': False,
'Dec': False}
Annual_rebate_accrual = {'Jan': 12, 'Feb': 1, 'Mar': 2, 'Apr': 3, 'May': 4, 'Jun': 5,
                      'Jul': 6, 'Aug': 7, 'Sep': 8, 'Oct': 9, 'Nov': 10, 'Dec': 11}

data['Mth'] = data['Period'].str.split('-')                            # Column we need
temporarily
data['Mth'] = data['Period'].str.split('-', expand = True)

data['Quarterly_payments'] = data['Mth'].map(Quarterly_payments)
data['Annual_rebate_accrual'] = data['Mth'].map(Annual_rebate_accrual)

data = data.drop(columns = ['Mth'])                                     # Do not need the
column any more
```

Extra data and methodology of multiple regression modeling drastically improved the predictive power of the model:

In [ ]:

```
results = smf.ols('Rebates_Payable ~ Sales + Quarterly_payments + Annual_rebate_accrual',
```

```
data = data).fit()
print(results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:        Rebates_Payable   R-squared:                       0.797
Model:                            OLS   Adj. R-squared:                  0.776
Method:                 Least Squares   F-statistic:                     36.70
Date:                Tue, 03 Apr 2018   Prob (F-statistic):           7.74e-10
Time:                        15:40:36   Log-Likelihood:                -511.76
No. Observations:                  32   AIC:                             1032.
Df Residuals:                      28   BIC:                             1037.
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
=====
                                 coef    std err          t      P>|t|      [0.025
0.975]
------------------------------------------------------------------------------------------
-------
Intercept                   4.758e+06   6.09e+06      0.781      0.441   -7.72e+06     1.
72e+07
Quarterly_payments[T.True]  2.611e+06   8.67e+05      3.011      0.005    8.35e+05     4.3
9e+06
Sales                          0.1270      0.043      2.954      0.006       0.039
0.215
Annual_rebate_accrual       1.018e+06   1.34e+05      7.613      0.000    7.44e+05     1.
29e+06
==============================================================================
Omnibus:                        4.878   Durbin-Watson:                   1.466
Prob(Omnibus):                  0.087   Jarque-Bera (JB):                3.280
Skew:                           0.647   Prob(JB):                        0.194
Kurtosis:                       3.885   Cond. No.                     2.21e+09
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specifie
d.
[2] The condition number is large, 2.21e+09. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Usually we split the data sample into train and test portions. We did not do that here because the data sample is small. So we went with coefficients to use those for prediction of rebates balances in the future. After we had applied those coefficients making our predictions we were able to compare predicted balances with actuals and were pretty satisfied with the result.

****This project became a crucial contributor to the forecasting project, which is internally known in company as xModel.**