What is Regression Analysis?

Regression analysis in business is an approach used to discover the statistical relations among two or more independent and dependent variables. In other words, one variable is independent and its effect on the opposite structured dependent variables is measured in regression analysis. There exist simple or multiple type regressions. If there is only one dependent and independent variable, or predictor variable, we say it as simple regression. Contrary, when there are many independent variables influencing one dependent variable, we say it as multiple regression.

Why Regression Analysis is important?

Regression analysis is all about machine to understand data. It helps businesses to understand the data/informational points that they have and their usage – explicitly the relations among data points. Such analysis helps to make better business decisions, including anything from predicting sales to understanding inventory levels and also supply and demand. Of all the business analysis techniques in machine learning, regression analysis is often referred to as one of the most significant. Business analysts and data professionals are frequently the ones that benefits through regression analysis as it helps them by pulling the relevant data and create reports for organizations department heads, management teams, sales units, board members, or anyone looking for significant data to guide or support decisions. This analysis is used to understand all kinds of patterns that pop up in data. These new insights can be extremely valuable in understanding what can make a difference in your business.

Benefits

Predictive Analytics

Organizations are turning to predictive analytics to help solve difficult problems and uncover new opportunities i.e. forecasting future opportunities and risks. Predictive or Demand analytics are also used to determine customer responses or purchases, as well as promote cross-sell opportunities. Predictive models help businesses attract, retain and grow their most profitable customers. However, demand is not the only dependent variable when it comes to predict businesses. Analyst can go far beyond forecasting impact on direct revenue. For example, forecasting the number of shoppers who will pass in front of a particular billboard and use that data to estimate the maximum to bid for an advertisement. Airlines use predictive analytics to set ticket prices. Hotels try to predict the number of guests for any given night to maximize occupancy and increase revenue. Insurance companies heavily rely on regression analysis to estimate the credit standing of policyholders and a possible number of claims in a given time period. Hence, predictive analytics enables organizations to function more efficiently.

Operation Efficiency:

Operational efficiency is primarily a metric that measures the efficiency of profit earned as a function of operational cost. Understanding the relationships between business happenings and other variables for operational Efficiency can be exceedingly important to make sure your business is prepared and effective.  Regression models can also be used to optimize business processes. A factory manager, for example, can create a statistical model to understand the impact of oven temperature on the shelf life of the cookies baked in those ovens. In a call center, we can analyze the relationship between wait times of callers and number of complaints. Data-driven decision making eliminates guesswork, hypothesis and corporate politics from decision making. This improves the business performance by highlighting the areas that have the maximum impact on the operational efficiency and revenues.

Credit Card

In the credit card company, regression analysis helps in understanding various factors like customer’s risk of credit default, expected consumer behaviour, prediction of credit balance, etc. and based on these results the company implements specific EMI options while minimising the default among risky customers.

Example:

Example covers a bicycle traffic predictor, in which FremontBridge dataset is combined with another dataset, to determine the extent to which weather and seasonal factors—temperature, precipitation, and daylight hours—affect the volume of bicycle traffic through this corridor. The NOAA makes available their weather station data (used station ID USW00024233) and we can easily use Pandas to join the two data sources. We will perform a simple linear regression to relate weather and other information to bicycle counts, in order to estimate how a change in any one of these parameters affects the number of riders on a given day.

CELL1:

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

CELL2:

#!curl -o FremontBridge.csv https://data.seattle.gov/api/views/65db-xm6k/rows.csv?accessType=DOWNLOAD

counts = pd.read\_csv('FremontBridge.csv', index\_col='Date', parse\_dates=True)

weather = pd.read\_csv('2942484.csv', index\_col='DATE', parse\_dates=True)

CELL3:

daily = counts.resample('d').sum()

daily['Total'] = daily.sum(axis=1)

daily = daily[['Total']] # remove other columns

CELL4:

days = ['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun']

for i in range(7):

    daily[days[i]] = (daily.index.dayofweek == i).astype(float)

CELL5:

from pandas.tseries.holiday import USFederalHolidayCalendar

cal = USFederalHolidayCalendar()

holidays = cal.holidays('2012', '2016')

daily = daily.join(pd.Series(1, index=holidays, name='holiday'))

daily['holiday'].fillna(0, inplace=True)

CELL6:

def hours\_of\_daylight(date, axis=23.44, latitude=47.61):

    """Compute the hours of daylight for the given date"""

    days = (date - pd.datetime(2000, 12, 21)).days

    m = (1. - np.tan(np.radians(latitude))

         \* np.tan(np.radians(axis) \* np.cos(days \* 2 \* np.pi / 365.25)))

    return 24. \* np.degrees(np.arccos(1 - np.clip(m, 0, 2))) / 180.

daily['daylight\_hrs'] = list(map(hours\_of\_daylight, daily.index))

daily[['daylight\_hrs']].plot()

plt.ylim(8, 17)

CELL7:

# temperatures are in 1/10 deg C; convert to C

weather['TMIN'] /= 10

weather['TMAX'] /= 10

weather['Temp (C)'] = 0.5 \* (weather['TMIN'] + weather['TMAX'])

# precip is in 1/10 mm; convert to inches

weather['PRCP'] /= 254

weather['dry day'] = (weather['PRCP'] == 0).astype(int)

daily = daily.join(weather[['PRCP', 'Temp (C)', 'dry day']])

CELL8:

daily['annual'] = (daily.index - daily.index[0]).days / 365.

daily.head()

CELL9:

from sklearn.linear\_model import LinearRegression

model = LinearRegression(fit\_intercept=True)

# Drop any rows with null values

daily.dropna(axis=0, how='any', inplace=True)

column\_names = ['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun', 'holiday',

                'daylight\_hrs', 'PRCP', 'dry day', 'Temp (C)', 'annual']

X = daily[column\_names]

y = daily['Total']

model = LinearRegression(fit\_intercept=False)

model.fit(X, y)

daily['predicted'] = model.predict(X)

CELL10:

daily[['Total', 'predicted']].plot(alpha=0.5);