Practica 1: Modelos de Regresión Lineal

Un fondo VC pequeño de Silicon Valley nos solicita la realización de un modelo que pueda permitirles brindar orientación sobre si deben invertir o no en startups en función del plan de negocios que le presenten los emprendedores. Es de vital importancia para los VC saber como se van a emplear los fondos ya que las inversiones son naturalmente de riesgo. El éxito de una startup esta en parte definido por sólidos planes de negocio, además del talento como equipo y que realmente su producto resuelva un problema.
Se nos ha proporcionado un dataset muy acotado que cuenta con 50 registros y surge del historial de inversiones de nuestro cliente. Las variables explicativas son:

- R&D Spend: Gasto (en dólares) en investigación y desarrollo.
- Administration: Gastos (en dólares) administrativos.
- Marketing Spend: Gasto (en dólares) en campañas de marketing
- State: Estado donde se encuentra la compañía.

La variable objetivo es:

- Profit: Ingresos que genera cada startup (en dólares)

Los datos presentados son reales y fueron obtenidos de: https://www.kaggle.com/karthickveerakumar/startup-logistic-regression

data_analysis.py

En este archivo realizaremos la etapa preliminar al modelado. El objetivo es crear nuestro data frame en base al archivo csv y conocer preliminarmente cual es la relación entre nuestras variables, para comenzar el modelado. Se proporciona un main() para probar el programa.

`create_dataframe(filename)`

Esta función recibe el nombre de un archivo del tipo csv y devuelve un data frame.

`plot_scatter(x, y, x_label, y_label)`

Dado valores de x e y, además del nombre que se desea como etiqueta de cada eje, mostrar un gráfico que muestra la dispersion de puntos para cada par x, y.

startup-prediction.py

Es hora de desarrollar el programa que dará soporte a los emprendedores. Las decisiones sobre qué variable/s van a emplear es a cargo de ustedes, los científicos de datos. Se les proporcionará las funciones que sirven de guía, pero son libres de agregar cuantas funciones consideren necesarias. Recuerden que es obligatorio en cada una, presentar una descripción de qué hace. Se deberán valer de dos funciones previamente trabajadas por ustedes, creadas en el archivo `data_analysis.py`.

`train_model(dataframe)`

Dado un Data frame válido, esta función debe entrenar un modelo de regresión lineal y retornarlo. Para ello se debe inicializar LinearRegression(), definir cuales serán las variables explicativas de nuestro modelo y recordar que la objetivo será *Profit.*

`betas(regr_model)`

El objetivo es retornar una tupla con los beta asociados al modelo, si es simple tendremos beta_1 y beta_0. En cambio, si es multiple tendremos beta_n, beta_(n-1), …, beta_0. Dado el modelo, utilizando los coeficientes y el valor donde intercepta desarrollar la función.

`mse(regr_model, dataframe)`

Sin utilizar la función reservada mean_squared_error, calcular el MSE de este modelo. Recordar lo visto en clase para aplicarlo a este caso. Retornar el valor obtenido.

`expected_profit()`

Definir para esta función la cantidad de variables que consideres necesarias, debe ser una por cada variable predictora. El objetivo es dado valores de gastos, retornar nuestra estimación de ingresos/ganancias (la declaración del dataset no es claro en eso)

Nuestro programa en acción

El cliente nos ha consultado por lo siguiente: La startup californiana X (nombre borrado por un acuerdo de confidencialidad) ha presentado el siguiente BP para su siguiente año:

- Gastos en R&D: 40.000 USD
- Gastos administrativos: 12.000 USD
- Gastos en Marketing: 129.300 USD
- Ganancias esperadas: 180.000 USD.

Según lo que diga tu modelo, ¿invertimos o no? Espero tu respuesta: https://forms.gle/rSSJRjScqPeFuSUC7

NOTA: Si tu programa respeta los conceptos de encapsulación, debe ser fácilmente adaptable para procesar y generar modelos de regresión lineal sea cual sea el dataset. Te propongo a que lo pongas a prueba y si generas un modelo interesante te invito a que lo compartas los próximos 10 minutos de la clase practica de la semana que viene.

** La solución creada por mí será lanzada el lunes 30/8 antes de clase.*

Autoría de esta práctica: Tomas Nozica