Actividad 7

José Daniel Gaytán Villarreal
Grupo 3

21 de marzo de 2019

Resumen

Se tomó una lista de datos correspondientes a una región y se obtuvo la correlación entre sus variables, graficando e intrepretando las relaciones resultantes.

1. Introducción

Un caso muy común en el análisis de datos es el de tener series de datos que se dificulta el leerlas a través de un programa, ya sea porque contiene carácteres especiales o simplemente carece de un orden establecido. Es preciso, para todo programador, el saber cómo modificar una serie de este tipo de manera que sea posible su análisis computacional.

Por otro lado, existe también entre cada variable de una serie de datos cierta *correlación*, ya sea directa o indirecta. Una de las funciones del lenguaje de programación de Python es el encontrar dicha correlación, de manera numérica en un valor entre -1 y 1, y mostrarla al usuario; lo anterior tiene distintos usos y aplicaciones, siendo uno de ellos, por ejemplo, el realizar un mapeo de datos, utilizando una biblioteca de visualiación de datos como Matplotlyb o Seaborn, de las correlaciones y comprender así el comportamiento de nuestras variables.

A continuación, en la sección 2 se explicará brevemente el procedimiento que se siguió en ésta actividad, mostrando posteriormente, en la sección de resultados, lo que se obtuvo de los mismos. Por último, en la conclusón se emite una pequeña reflexión acerca de nuestros reultados y las implicaciones pragmáticas que del mapeo de correlaciones para el análisis estadístico de datos.

2. Desarrollo

2.1. Metodología

Se comenzó la actividad leyendo un archivo de datos; dado que dicho archivo contenía carácteres especiales que Python no reconocía, se utilizó el siguiente comando para su lectura:

```
df = pd.DataFrame( pd.read_csv("meteo-nogal-09.csv", engine="python" ))
```

Nótese que, a diferencia de actividades anteriores, se utilizó el parámetro *engine* al establecer el DataFrame, debido a que el lector de la biblioteca Panda es incapaz de reconocer los carácteres, mas el lector de Python sí lo es.

Una vez leído el DataFrame, se eliminaron ciertas columnas y variables que nos eran inútiles para el análisis de datos, realizándose de la siguiente manera:

```
df.drop( df.columns[18:36], axis=1, inplace=True )
df.drop( df.columns[2:4], axis=1, inplace=True )
df.head()
```

Por último, ya trabajado el DataFrame, era necesario encontrar la relación existente entre neustras variables. Para lo anterior, simplemente utilizamos la función intrínseca corr() y le asignamos un DataFrame de la siguiente manera:

```
df_corr = df.corr(method='pearson', min_periods=1)
```

En la función anterior, *method* indica el método utilizado para encontrar la correlación y $min_periods$ el valor entre los que oscila la relación.

Por último, se realizaron mapeos, tanto en Seaborn como en MatPlotLyb, de la relación de datos resultante:

```
En Seaborn:
sns_plot = sns.heatmap(df_corr,cmap="viridis",robust=True,square=True,annot=False)

En Matplotlyb:
fig, ax = plt.subplots()
ax.set_xticks(np.arange(len(df_corr)))
ax.set_yticks(np.arange(len(df_corr)))
ax.set_xticklabels(df_corr)
ax.set_yticklabels(df_corr.columns[::-1])

plt.setp(ax.get_xticklabels(), rotation=90, ha="right",rotation_mode="anchor")
ax.set_title("Gráfica de correlaciones")
plt.imshow(datos, cmap='viridis',interpolation='nearest')
```

2.2. Resultados

Del análisis anterior se realizaron las siguientes gráficas, las cuales se muestran a continuación:

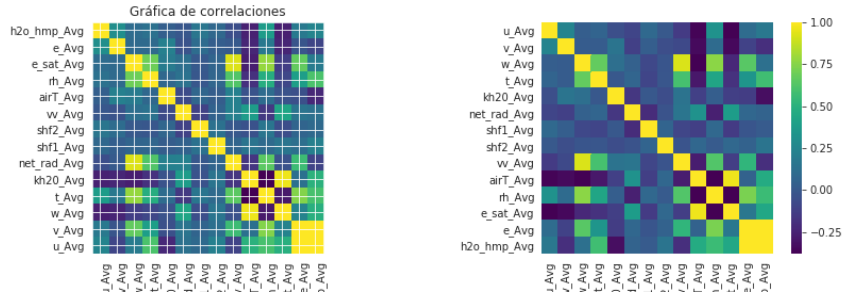

Figura 1: Gráficas que muestran la correlación entre las variables estudiadas, realizadas con Matplotlyb y Seaborn, respectivamente.

Para las correlaciones cuyo valor absoluto era mayor a 0.5, se realizaron gráficas para visualizar dicha correlación, resultando así las siguientes:

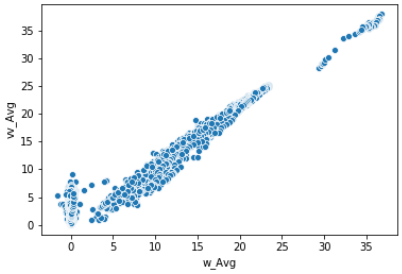

Figura 2: Correlación entre wAvg y vvAvg

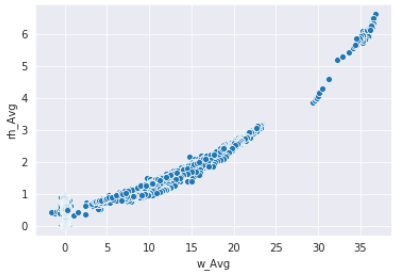

Figura 3: Correlación entre rhAvg y wAvg

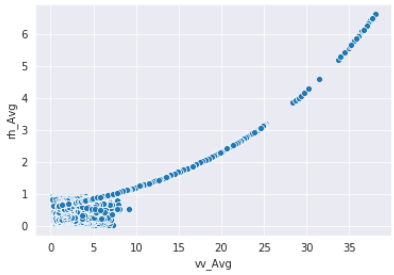

Figura 4: Correlación entre esaitAvg y airTAvg

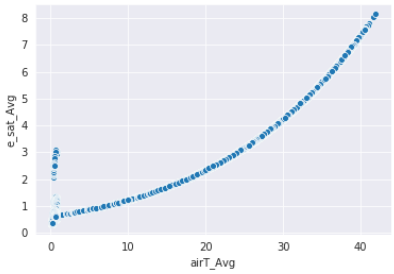

Figura 5: Correlación entre h20hmpAvg y eAvg

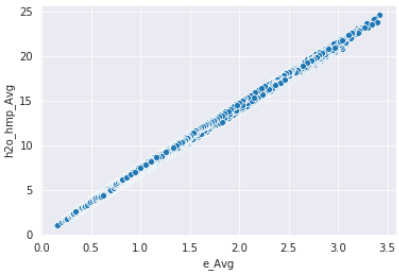

Figura 6: Correlación entre rhAvg y vvAvg

3. Conclusión

De las gráficas, vemos que, mientras más se aproxime el valor de la correlación a 1, la gráfica se aproxima más a una recta, mientras que en el caso contrario, la correlación se pierde y tiende a formar curvas. Podemos, pues, discernir el significado geométrico de la correlación entre dos variables: mientras más se aproxime a uno, la relación se vuelve cada vez más lineal.