Modelos de dataset electorales

Hemos llevado a cabo una serie de modelizaciones a partir de los datasets de las distintas elecciones, con el objetivo de comprobar si los resultados tenían o no sentido. Afortunadamente, creemos de veras que estamos en el primer caso. En total se han probado ocho modelos, contenidos cada uno de ellos en un cuaderno, siete de ellos supervisados, y un octavo de *clusterización*. Muy brevemente, son los siguientes:

- ***RForest_Clasif_PP_PSOE***, como su nombre indica, desarrolla un modelo random forest de clasificación de la diferencia en porcentaje entre PP y PSOE en el dataset las elecciones de abril de 2019.

- ***RForest_Regresion_PP_PSOE,*** que modeliza la misma variable del mismo dataset que en cuaderno anterior, pero en esta ocasión mediante regresión.

- ***SVM_Clasif_porcentaje_Vox_N19,*** tratándose de un modelo de support vector machines (SVM) de clasificación del porcentaje de voto de Vox en noviembre de 2019.

- ***SVM_Regresion_porcentaje_Vox_N19,*** en esta ocasión aplicamos regresión a la misma variable que el cuaderno anterior.

- ***RNeuronal_Clasif_dif_PodIU_PSOE_D15,*** en el que usamos una red neuronal para modelar la diferencia de voto entre, por un lado, la suma de Podemos e IU, y, por otro, el PSOE, en las elecciones de diciembre de 2015.

- ***RNeuronal_Regresion_dif_PodIU_PSOE_D15,*** donde tratamos el mismo problema que en cuaderno anterior, en este caso mediante regresión.

- ***Lasso_Regresion_diff_PP_16_19,*** en el que implementamos una regresión lineal regularizada con Lasso para modelar la diferencia de voto al PP entre las elecciones de 2016 y la de abril de 2019.

- ***Cluster_Santander,*** donde tomamos las secciones electorales del municipio de Santander en los comicios de abril 2019 y aplicamos dos clusterizaciones, representándolas gráficamente mediante Geopandas.

Estructura de los cuadernos

Los siete modelos supervisados tienen la misma estructura. El primer paso es la definición de la variable objetivo, de las que, como se ve, hemos considerado unas pocas de las infinitas posibles. En el caso de clasificación hemos intentado crear una distribución de cinco posibles valores bastante balanceada. La definición de la variable objetivo en ocasiones es simplemente

una de las columnas del dataset, como es el caso del porcentaje de voto de Vox, y en otros casos es más complicada, resultado por ejemplo de combinar dos datasets, como ocurre con la evolución del voto al PP entre 2016 y 2019.

El siguiente paso es la división del dataset entre *train* y *test*, cosa que efectuamos sin gran novedad. El tratamiento de *train* incluye: **1)** la selección de las columnas que podemos llamar válidas, que no incluyen las del voto o porcentaje de voto a partidos, pues lo que queremos es modelizar según a los datos socioeconómicos; **2)** la sustitución de los valores en NaN por la media de sus respectivas columnas; **3)** el tratamiento de la columna categórica, que suele ser la provincia o la CCAA; **4)** el análisis de la correlación entre columnas y la eliminación de algunas de ellas con un dato muy elevado; y **5)** la selección de las filas con datos válidos en la columna objetivo. Esta misma serie de procesos, y en el mismo orden, se lleva a cabo en el dataset de test.

Antes de la modelización llevamos a cabo los tests de relevancia F-test y Mutual Information Regression para ver cuáles pueden ser las columnas más importantes. En todos los casos la columna geográfica (CCAA o provincia) es siempre de las más importantes, junto en general a las de renta per cápita y las de afiliación a la Seguridad Social. En el caso de los modelos de SVM tomamos directamente estas columnas para reducir el tiempo de ajuste. El último paso es la modelización para lo que utilizamos *grid search*, con los hiperparámetros principales, antes del ajuste (fit) definitivo. En muchos casos nos encontramos con casos claros de *overfitting,* con lo que tenemos que tomar otros valores y/o reducir el número de columnas, como ocurre en especial en el caso de los modelos de random forest.

En general los resultados los podemos considerar como aceptables. Los modelos de clasificación dan un *accuracy* cercano al 60%, frente al 20% que sería el que encontraríamos en el caso de una clasificación aleatoria. En los de regresión, el error generado suele estar en su mayoría en el entorno de la división estándar o por debajo. Los detalles de cada caso se pueden consultar, evidentemente, en cada cuaderno.

En el caso del cuaderno de la clusterización de las secciones de Santander, hemos aplicado varios modelos presentes en la librería de SKlearn, como KMeans. En el caso de DBSCAM no hemos tenido éxito en el modelo. Tras una clusterización según el voto, llevamos a cabo otra de acuerdo a los datos socioeconómicos que, de hecho, da resultados muy similares, lo que mostraría indiciariamente una correlación entre ambos conjuntos de variables.