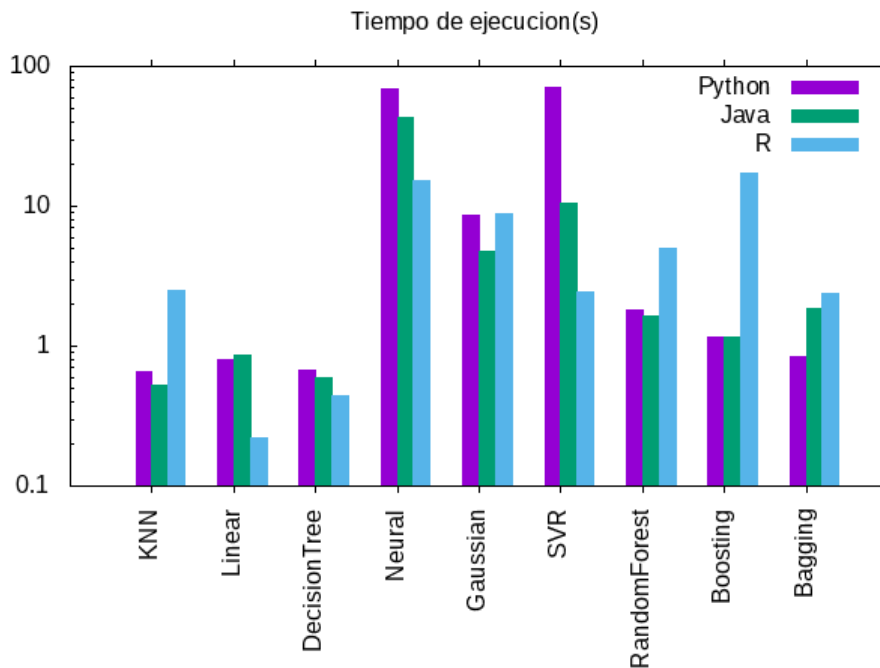
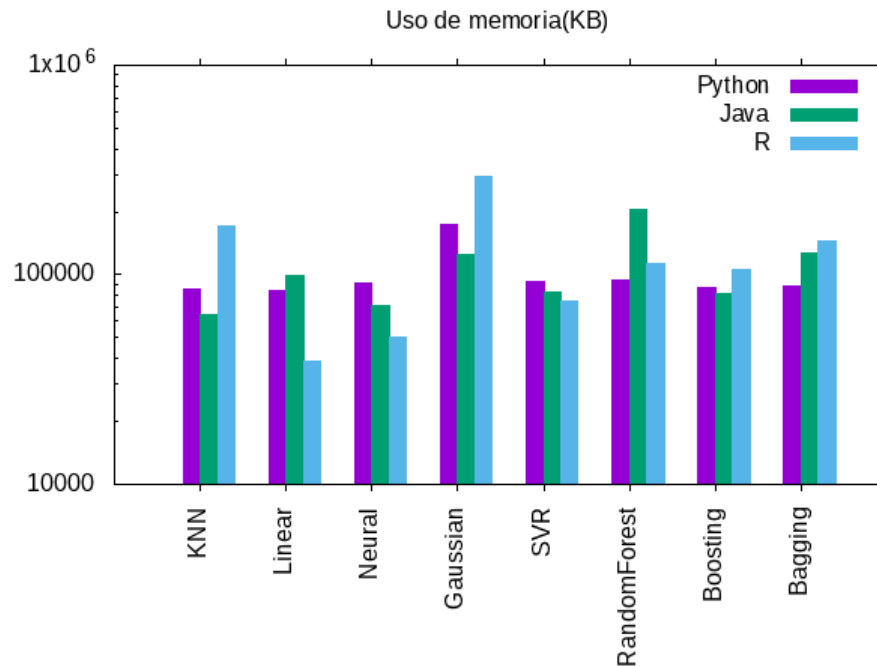


## 1 Descripción del problema.

## 2 Herramientas.

Las herramientas que se pueden usar para abordar la resolución del problema es amplia. Es por ello, que resulta difícil elegir la más adecuada. Tras un estudio de las bibliotecas disponibles para los distintos lenguajes de programación que domino, me decante por weka(java),sklearn(python) y R. Para determinar cual de las 3 es la más conveniente en este caso, realizare pruebas de ejecución de los diferentes algoritmos y compararemos el tiempo de ejecución y el uso de memoria. Además tendré en cuenta la facilidad para tratar los datos y la biblioteca "en si". Los resultados obtenidos se muestran en las siguientes gráficas:





Observando los resultados podemos concluir que sklearn y weka ofrecen un rendimiento similar, mientras que R ofrece un mejor rendimiento para los algoritmos básicos pero es inferior en la ejecución de multclasificadores. En cuanto al tratamiento de los datos tanto de lectura como para generar el archivo de resultados, tanto en R como en sklearn se hace de forma cómoda. Otra cuestión a tener en cuenta, es la buena documentación con la que cuenta sklearn. Por tanto, poniendo en la balanza estos apuntes me decanto por Python como lenguaje a usar.

En particular, se usaran las siguientes bibliotecas:

- NumPy.
- Pandas.
- Scikit Learn.
- XGBoost.

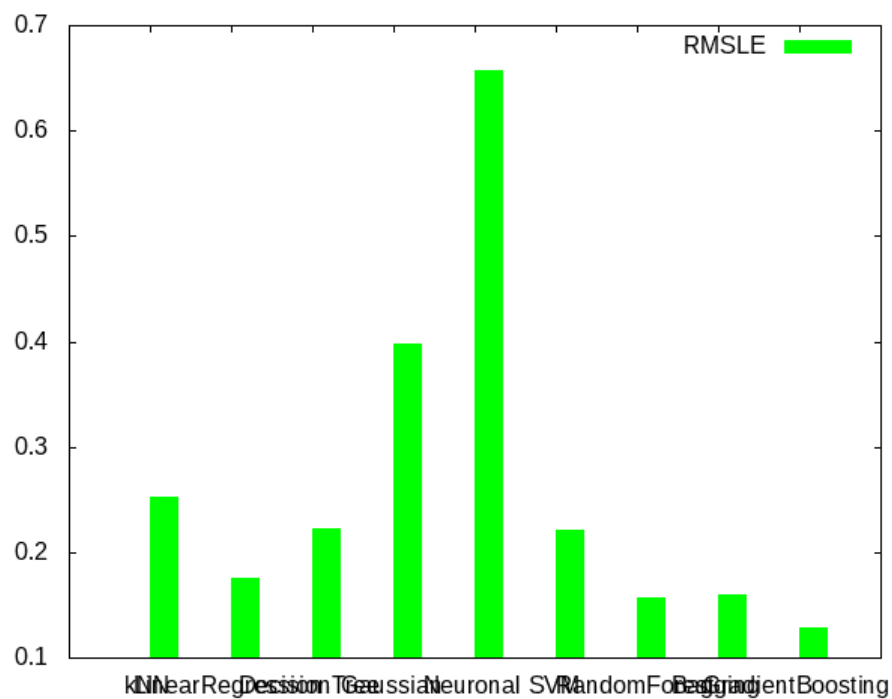
### 3 Algoritmos.

A priori desconocemos que algoritmo se adapta mejor a nuestro problema, es por ello que realizaremos un estudio comparando varios algoritmos. Los algoritmos elegidos son los siguientes:

- algoritmos básicos como KNN, Linear Regression y Arboles de regresion
- algoritmos potentes como NeuralNetwork, Gaussian y SVM

- multclasificadores como Random Forest, Boosting y Bagging

En la siguiente gráfica se muestran los resultados obtenidos.



Se puede observar que los algoritmos que menor error proporcionan son los multclasificadores. Por tanto, al igual que los competidores, nos decantamos por esta opción.