**Informe Evaluación N°2 Machine Learning**

Maximiliano Aros

**Índice**

[Comprensión del negocio 3](#_w5l5jd7wo7up)

[Comprensión de los datos 4](#_1dy5imkbazoc)

[Preparación de datos 6](#_vflvtp2e4ln7)

[Modelamiento 8](#_bm86qj3tb6ck)

[Regresión Lineal 8](#_8gs35etcjwcg)

[Árbol de Decisión 9](#_9ausnygp9fa1)

[Bosque Aleatorio 10](#_sftrztxstpvx)

[Evaluación del modelo 11](#_qrbml3c9d0gj)

[Implementación 13](#_9wgmh6o1bvra)

[Conclusión 14](#_mdquass8lzco)

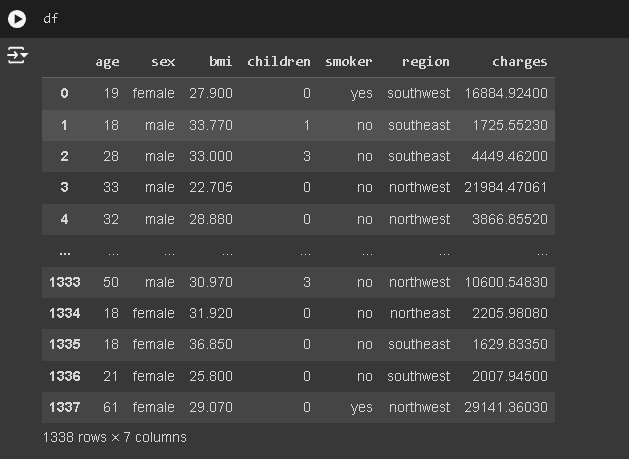
# **Comprensión del negocio**

Safe es una compañía de seguros muy destacada y conocida en Estados Unidos, y actualmente tiene la necesidad de hacer predicciones de seguros en el mercado.

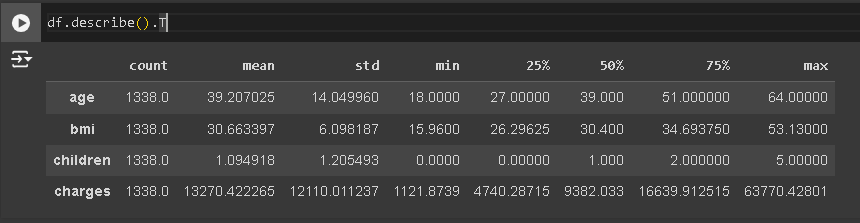
Se debe crear un modelo que pueda ser consistente, preciso y acorde al mercado actual de seguros.

# **Comprensión de los datos**

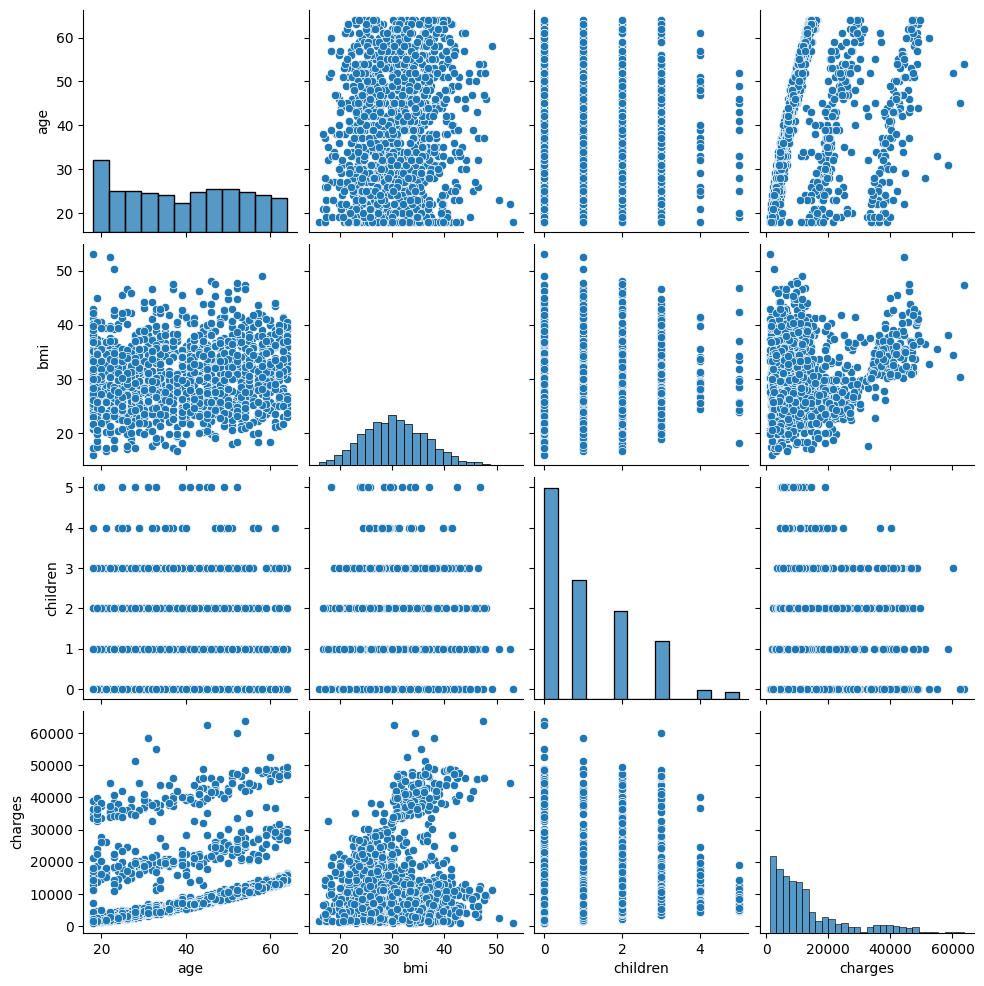
A continuación conoceremos cuáles son los datos, utilizando df podemos apreciar que la tabla posee 7 columnas y 1338 filas. También podemos empezar a conocer los tipos de datos de cada columna.



A continuación conoceremos el promedio, desviación estándar, valor mínimo, valor máximo y los cuartiles.



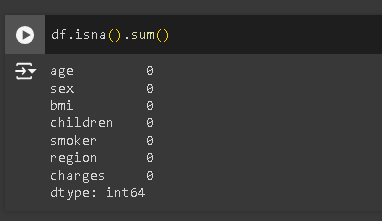
Por último podemos apreciar todos los gráficos generados



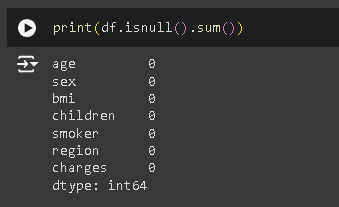
# **Preparación de los datos**

Una vez conocemos los datos tenemos que prepararlos para poder trabajarlos.

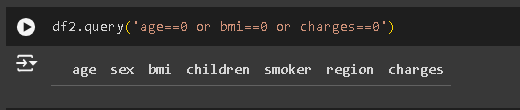
La siguiente celda buscará valores atípicos igual a NaN (not a number). Podemos apreciar que la tabla no posee valores NaN.



La siguiente celda buscará valores atípicos igual a Null. Podemos apreciar que la tabla no posee valores Nulos.



La siguiente celda buscará valores atípicos igual a 0. Podemos apreciar que la tabla no posee valores igual a 0.

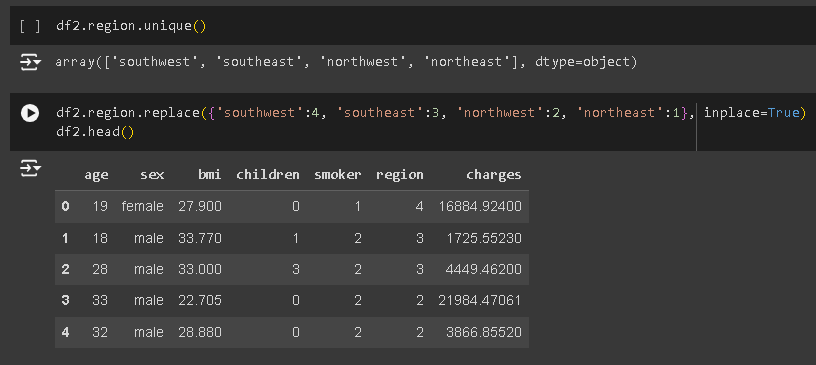


El siguiente paso de esta etapa sería convertir los valores string en number , y asignarles un número para poder trabajar de mejor manera con todos los datos.

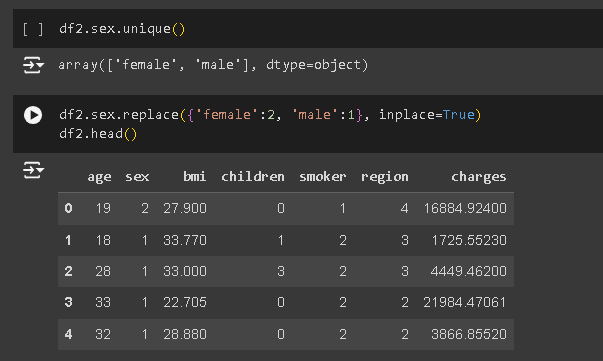
Comenzaremos por la columna Smoker.



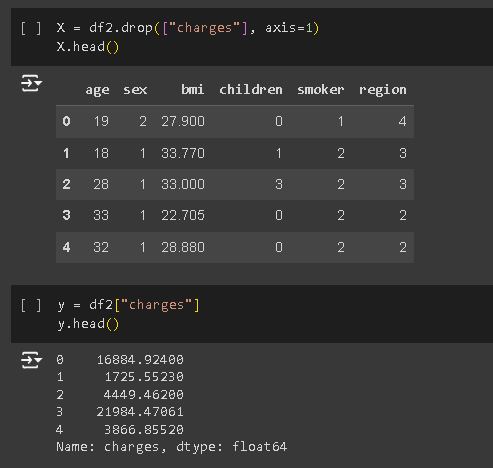
Ahora, lo mismo con la columna Región.



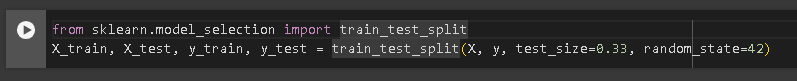
Y lo mismo con la última columna, Sex.



Casi terminando, crearemos una variable X y una variable Y, las cuales almacenarán en el caso de X, toda la tabla exceptuando la columna Charges, y en el caso de Y, sólo almacenará la columna Charges de la tabla.



Por último importamos la clase train\_test\_split que utilizaremos para poder dividir los datos conjuntos de entrenamiento y de prueba. A continuación con el 0.33 estamos especificando el porcentaje de datos que utilizaremos para como prueba, es decir un 33% y por ende el 67% restante será de entrenamiento.



# **Modelamiento**

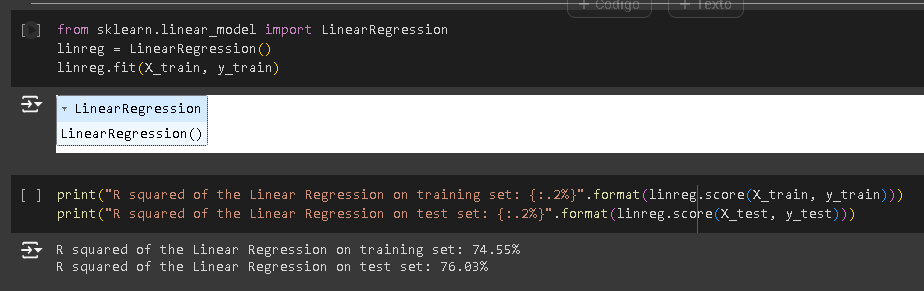
Con los datos ya preparados, comenzaremos a darle forma al modelo, Se estrenarán tres modelos distintos: Regresión Lineal, Árbol de Decisión y un Random Forrest.

## **Regresión Lineal**

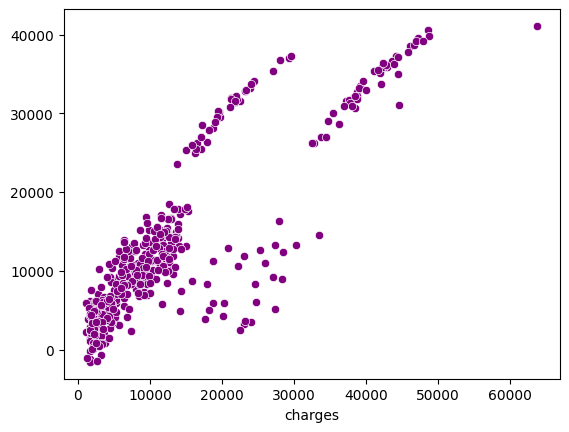
Lo primero que haremos será importar la clase LinearRegression.

Luego de instanciar el objeto “linreg”, se entrena el modelo llamando al método “fit()” con x\_train y y\_train, declarados anteriormente.

Por último se imprime el coeficiente de determinación (R-cuadrado) del modelo, tanto en el conjunto de entrenamiento como en el conjunto de prueba(x\_train con y\_train , y x\_test con y\_test) utilizando el método “score()”.



Ahora haremos un gráfico de dispersión para visualizar las predicciones frente a los valores reales en el conjunto de prueba.

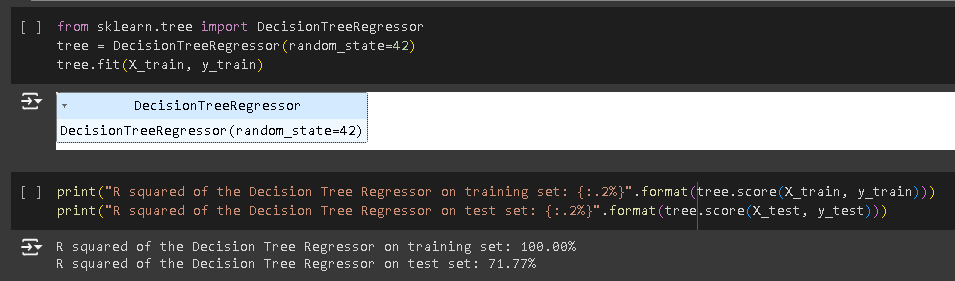


## **Árbol de Decisión**

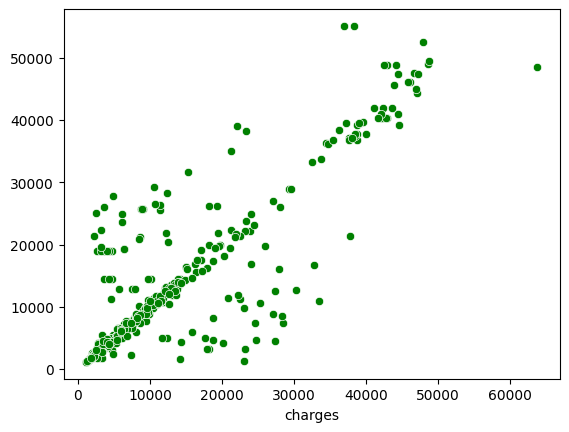
Lo primero que haremos será importar la clase DecisionTreeRegresor.

Luego de instanciar el objeto “tree”, se entrena el modelo llamando al método “fit()” con x\_train y y\_train, declarados anteriormente.

Por último se imprime el coeficiente de determinación (R-cuadrado) del modelo, tanto en el conjunto de entrenamiento como en el conjunto de prueba(x\_train con y\_train , y x\_test con y\_test) utilizando el método “score()”.



Ahora haremos un gráfico de dispersión para visualizar las predicciones frente a los valores reales en el conjunto de prueba.

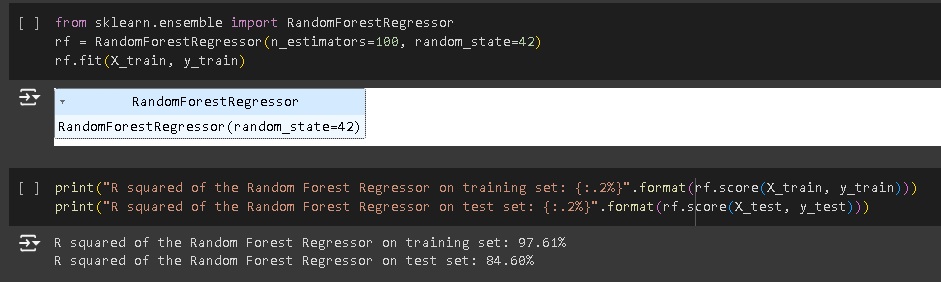


## **Bosque Aleatorio**

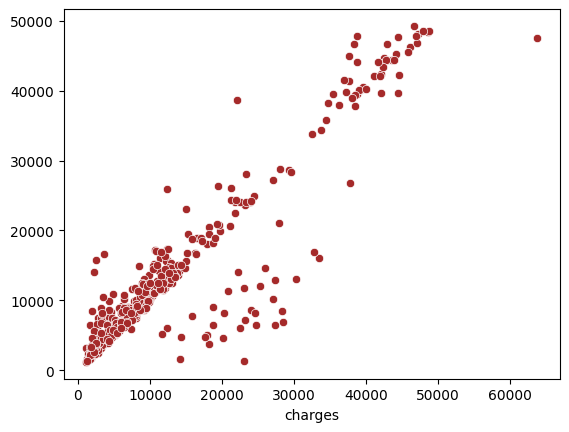
Lo primero que haremos será importar la clase RandomForestRegresor.

Luego de instanciar el objeto “rf”, se entrena el modelo llamando al método “fit()” con x\_train y y\_train, declarados anteriormente.

Por último se imprime el coeficiente de determinación (R-cuadrado) del modelo, tanto en el conjunto de entrenamiento como en el conjunto de prueba(x\_train con y\_train , y x\_test con y\_test) utilizando el método “score()”.



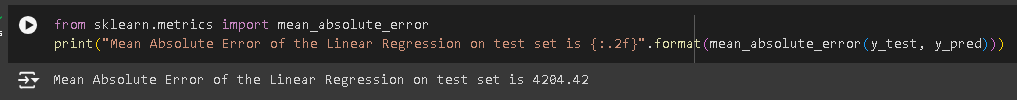
Ahora haremos un gráfico de dispersión para visualizar las predicciones frente a los valores reales en el conjunto de prueba.



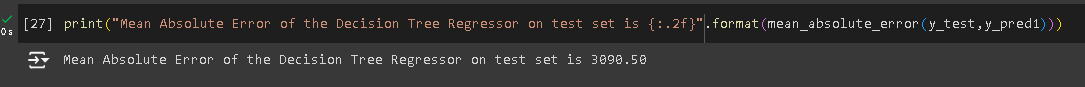
# **Evaluación del modelo**

En esta etapa se determina la calidad del modelo teniendo en cuenta el análisis de determinadas métricas y criterios estadísticos del mismo, comparando los resultados con resultados previos.

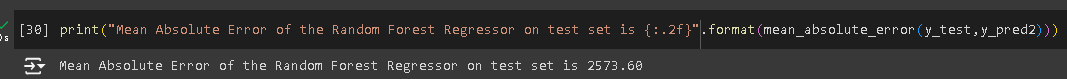
Con el resultado dado podemos decir que el modelo de **regresión linear** pudo predecir el precio de cada diamante en el conjunto de prueba con un error de ±$4204.42 del precio real.



Con el resultado dado podemos decir que el modelo de **árbol de decisión** pudo predecir el precio de cada diamante en el conjunto de prueba con un error de ±$3090.50 del precio real.



Con el resultado dado podemos decir que el modelo de **bosque aleatorio** pudo predecir el precio de cada diamante en el conjunto de prueba con un error de ±$2573.60 del precio real.



Por ende podemos decir y confirmar, que de los 3 modelos de predicciones utilizados, el más óptimo y preciso en este caso es el bosque aleatorio

# **Implementación**

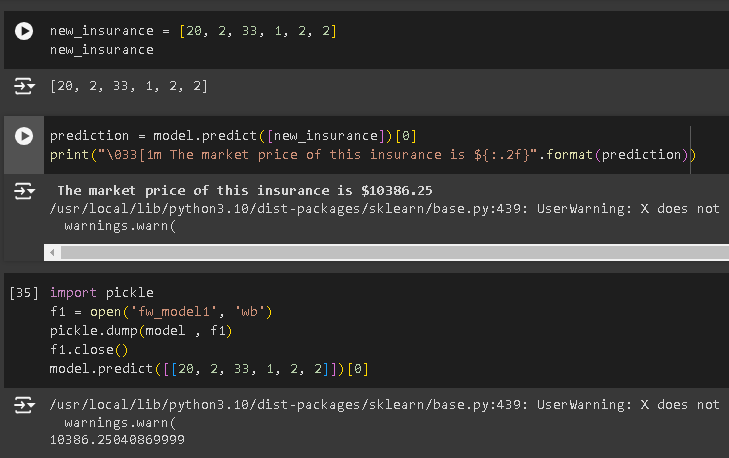
Aquí el modelo ya ha sido construido y evaluado.

Guardar el modelo finalizado en un archivo con formato “pickle”, nos puede ahorrar mucho tiempo ya que no tenemos que entrenar nuestro modelo cada vez que ejecutamos la aplicación. Una vez que guardamos nuestro modelo con dicho formato, más tarde puede cargarse para realizar la predicción.

A continuación crearemos un nuevo seguro para luego hacer la predicción.

Para esto le daremos los valores 20 (20 años de edad), 2 (Sexo femenino), 33 (indice de masa corporal), 1 (1 hijo), 2 (No es fumador) y 2 (region al noroeste).

El precio del seguro predicho por el modelo utilizando el bosque aleatorio es de $10386,25



# **Conclusión**

Se pudo abordar el desafío de Safe de realizar predicciones de seguros en el mercado. Se ha llevado a cabo un proceso de comprensión del negocio y preparación de datos, seguido de la implementación de tres modelos de machine learning: Regresión Lineal, Árbol de Decisión y Bosque Aleatorio.

Los resultados muestran que el modelo de Bosque Aleatorio ha demostrado ser el más efectivo para predecir el precio de mercado de un seguro, superando a los otros dos modelos en términos de precisión. Sin embargo, se reconoce que aún existe margen para mejorar la calidad del modelo.