Por medio del presente documento, se hace descripción completa del modelo predictivo de *Machine Learning* desarrollado por Franco, Malkum y Osorio (2022) con el objetivo de predecir de la manera más acertada posible el precio de las viviendas de la ciudad colombiana de Santiago de Cali.

Para el objetivo anterior, es relevante reconocer la ausencia de información para la capital del departamento del Valle del Cauca. Sin embargo, se hace uso de los datos de propiedad individual de las ciudades de Bogotá y Medellín provenientes de la plataforma de *Properati.*

Además, se debe tener en cuenta el interés enfático de la compañía para la cual se desarrolla el presente informe en querer evitar una tragedia como la sucedida en el renombrado “fiasco de Zillow” donde se sobreestimó el valor de un gran número de propiedades. Por tanto, al momento de ser adquiridas, generaron fuertes cargas presupuestales para la compañía en cuestión, que para el tercer trimestre de 2021 llegó incluso a reportar pérdidas por $304 millones de dólares que requirieron de un agresivo recorte de personal.

Como resultado, se plantea una métrica de evaluación distinta a minimizar el error cuadrático medio para entrenar el modelo. En primer lugar, se conserva la forma funcional cuadrática cuando se sobreestima el precio, pero, cuando el precio predicho es menor al real, la penalización se hace de forma lineal (por lo que, de hecho, toma un valor negativo). Con lo anterior, el modelo va a tender a subestimar un poco más sus predicciones o en caso de sobreestimar, hacerlo en menor proporción. No obstante, no se pretende que no se compre ninguna propiedad bajo la lógica de que es la mejor manera de minimizar gastos, pues de ser así, sencillamente no habría negocio. Para lo anterior, se establece una penalidad más alta una vez se subestiman los precios por más de 40 millones. A continuación, se muestra la métrica construida:

Con la información anterior, se entrenaron modelos de regresión con regularización, *Random Forest* con árboles de regresión, *xgboost* y *superlearner.* De los anteriores el X entregó los mejores resultados (describir resultados de la métrica).