

TALENT HIGHWAY

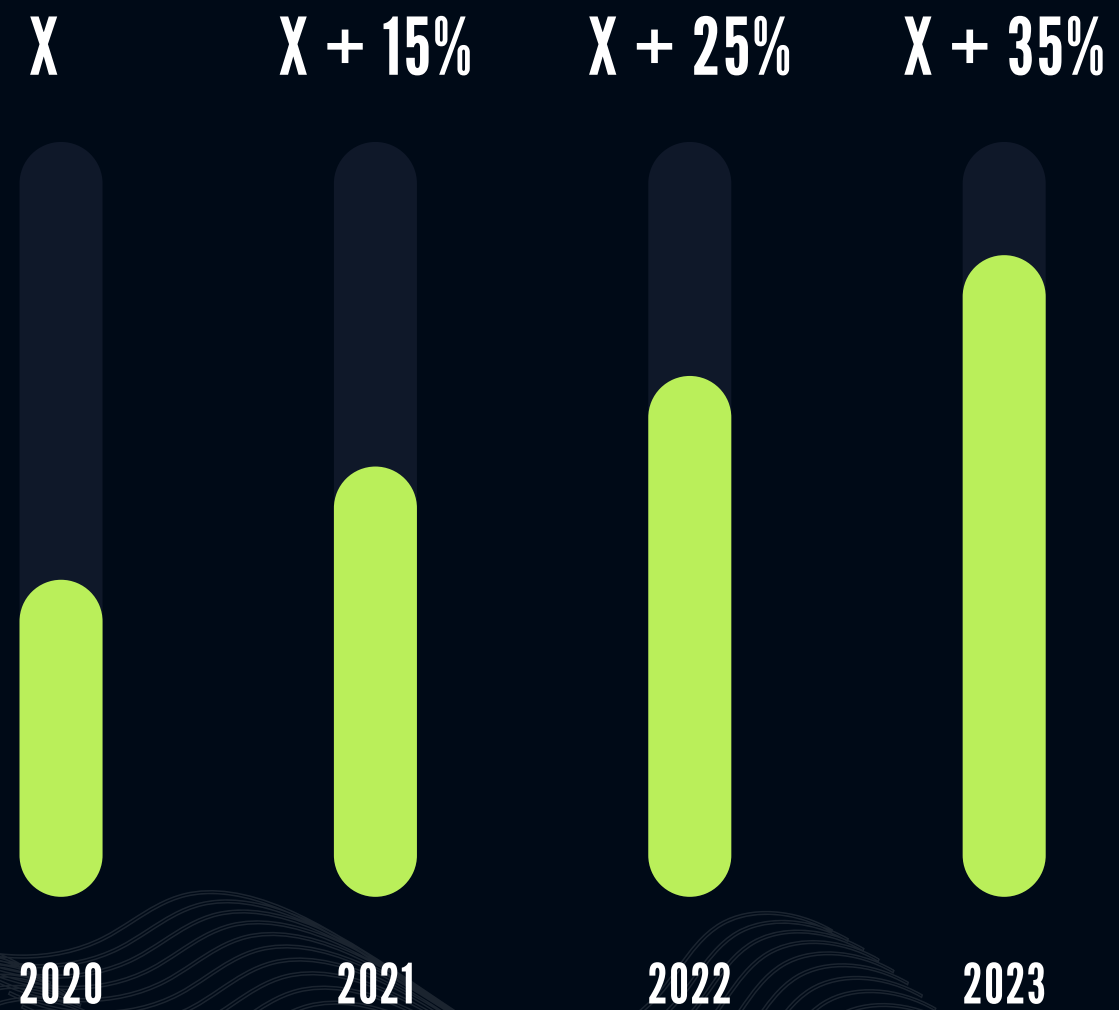
# NUEVAS OPORTUNIDADES

## MERCADO SEGUNDA MANO

MOWATAVE

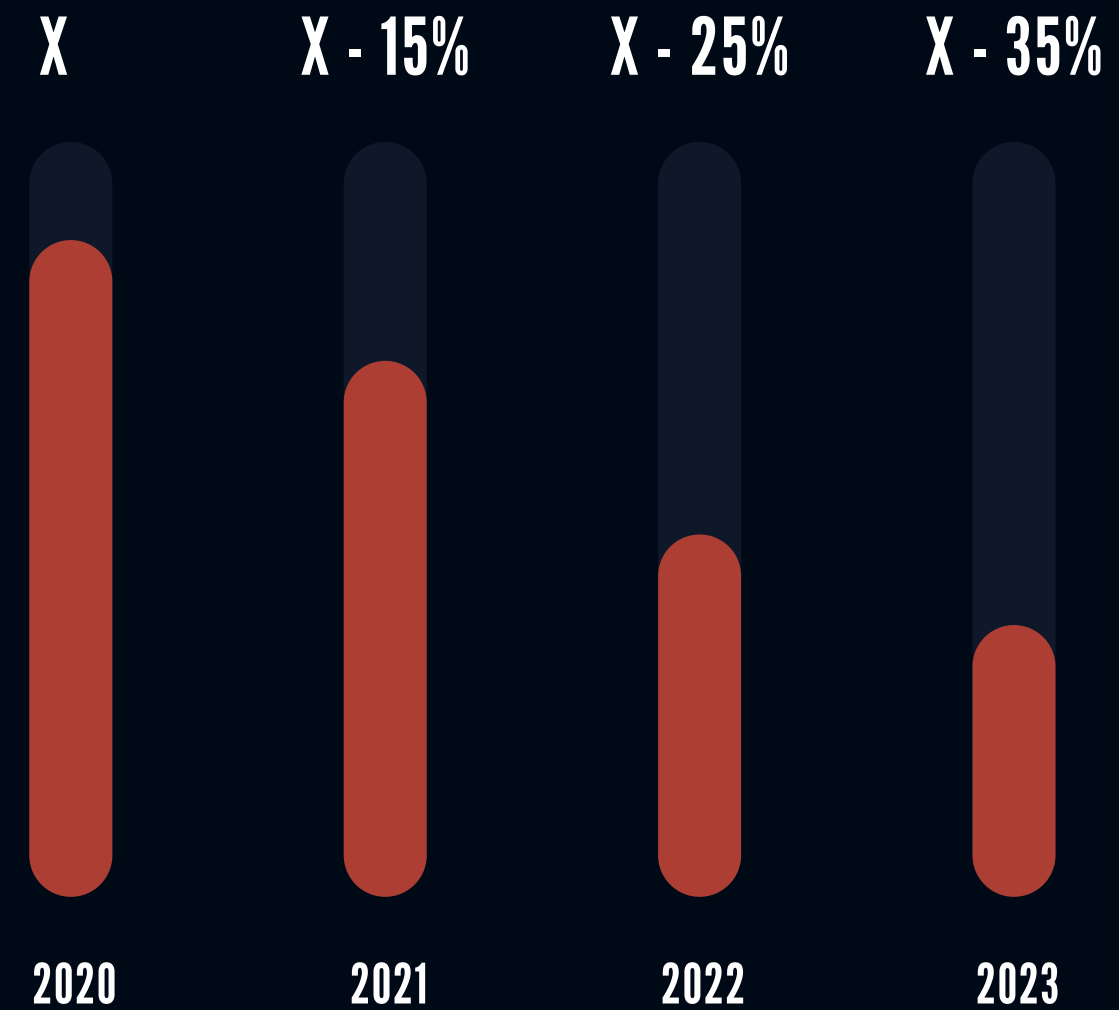
# PREMISAS

1 - Conforme pasan los años, mas unidades de un modelo de coche en concreto aparece en el mercado de 2ª Mano



Unidades venta  
2ª Mano

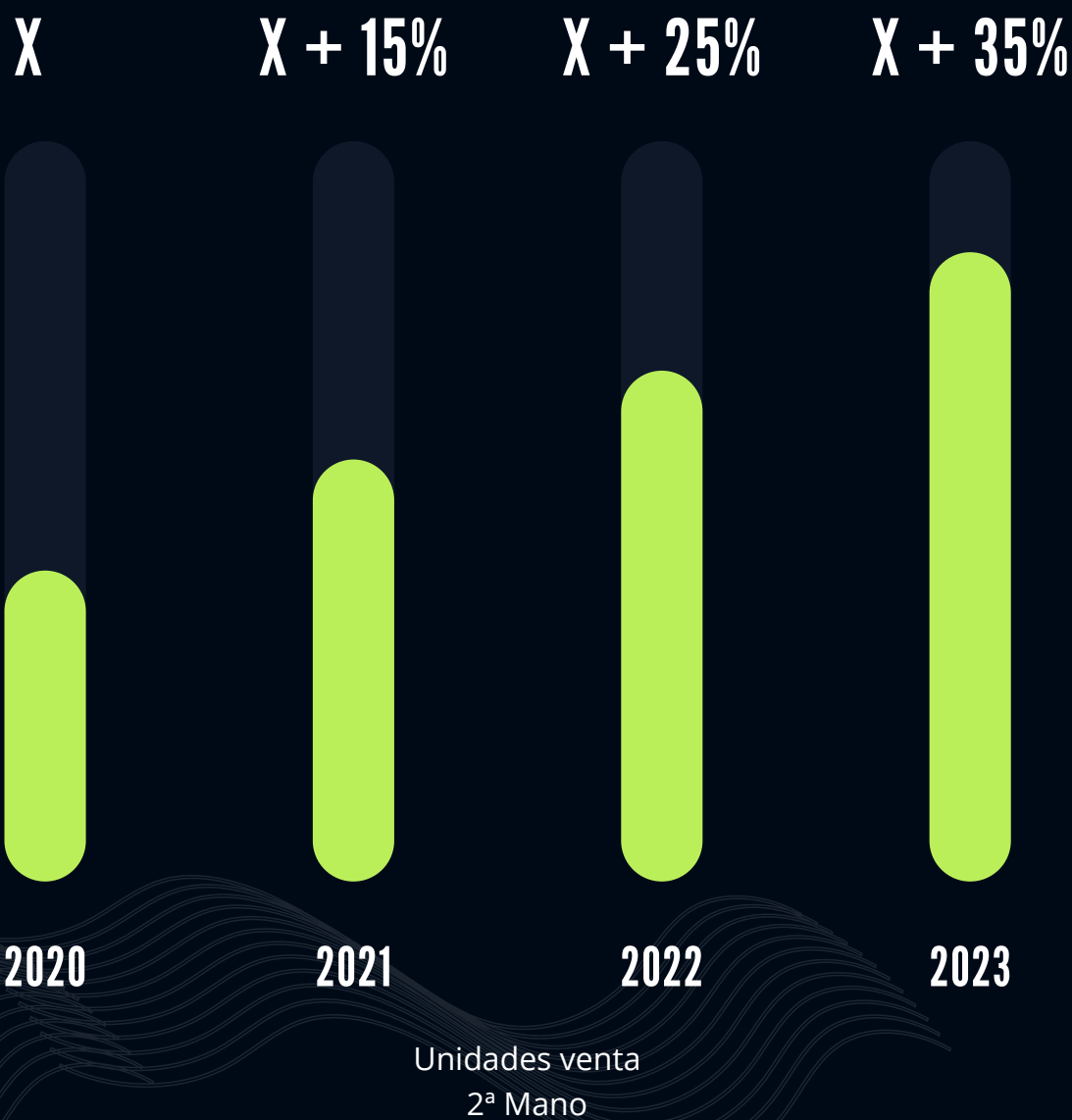
2- Conforme pasan los años, el precio en el mercado de 2ª Mano de ese mismo modelo disminuye



Precio venta  
2ª Mano

# PREMISAS

1 - Conforme pasan los años, mas unidades de un modelo de coche en concreto aparece en el mercado de 2ª Mano



## PUNTO ROTURA VOLUMEN (PRV)

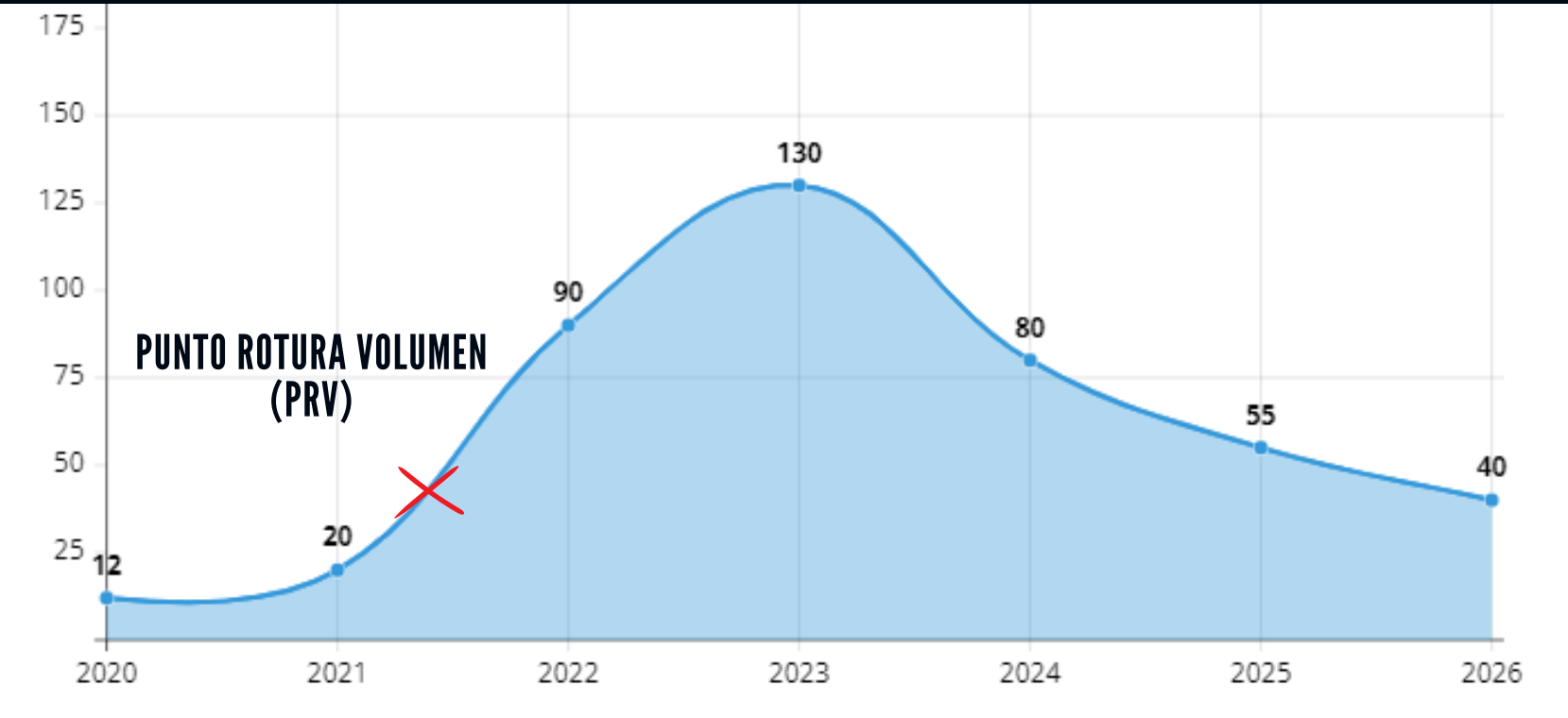


TABLA EJEMPLO  
VOLUMEN COCHES EN VENTA 2º MANO / AÑOS

# PUNTO ROTURA PRECIO (PRP)

2- Conforme pasan los años, el precio en el mercado de 2ª Mano de ese mismo modelo disminuye

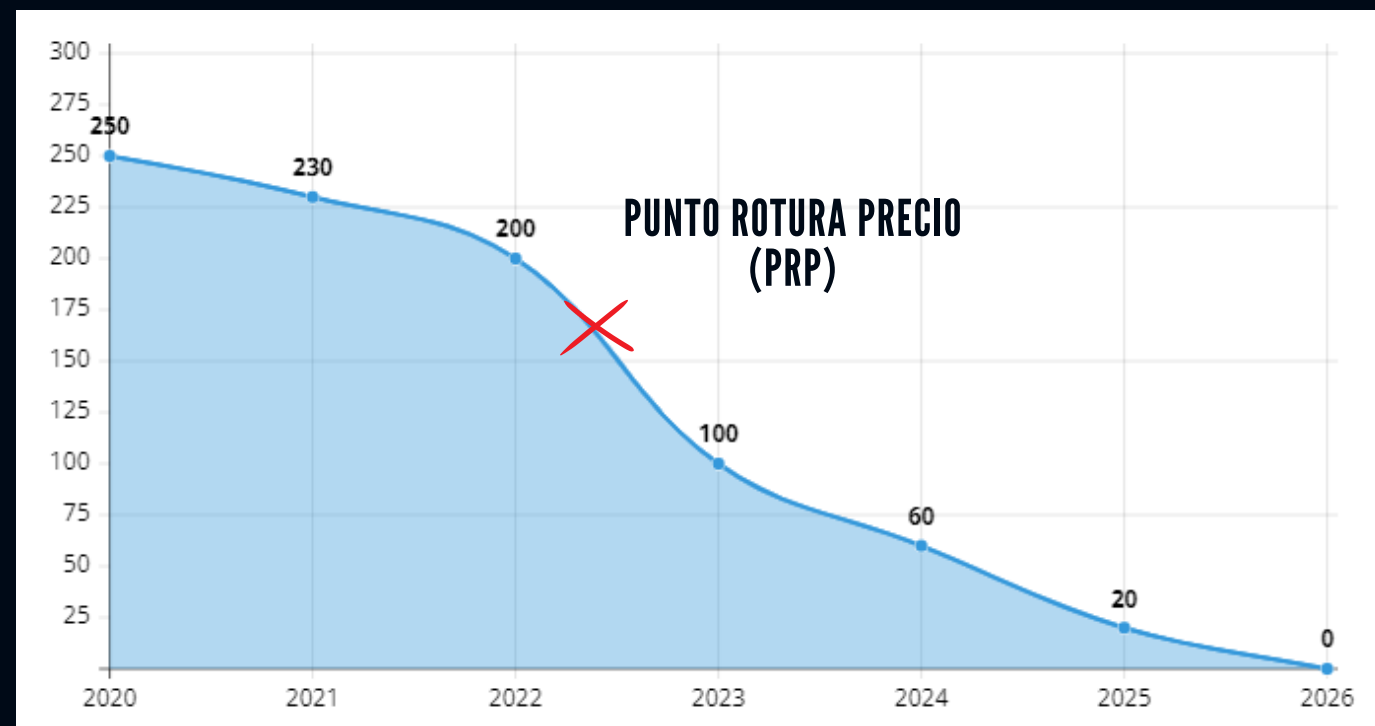
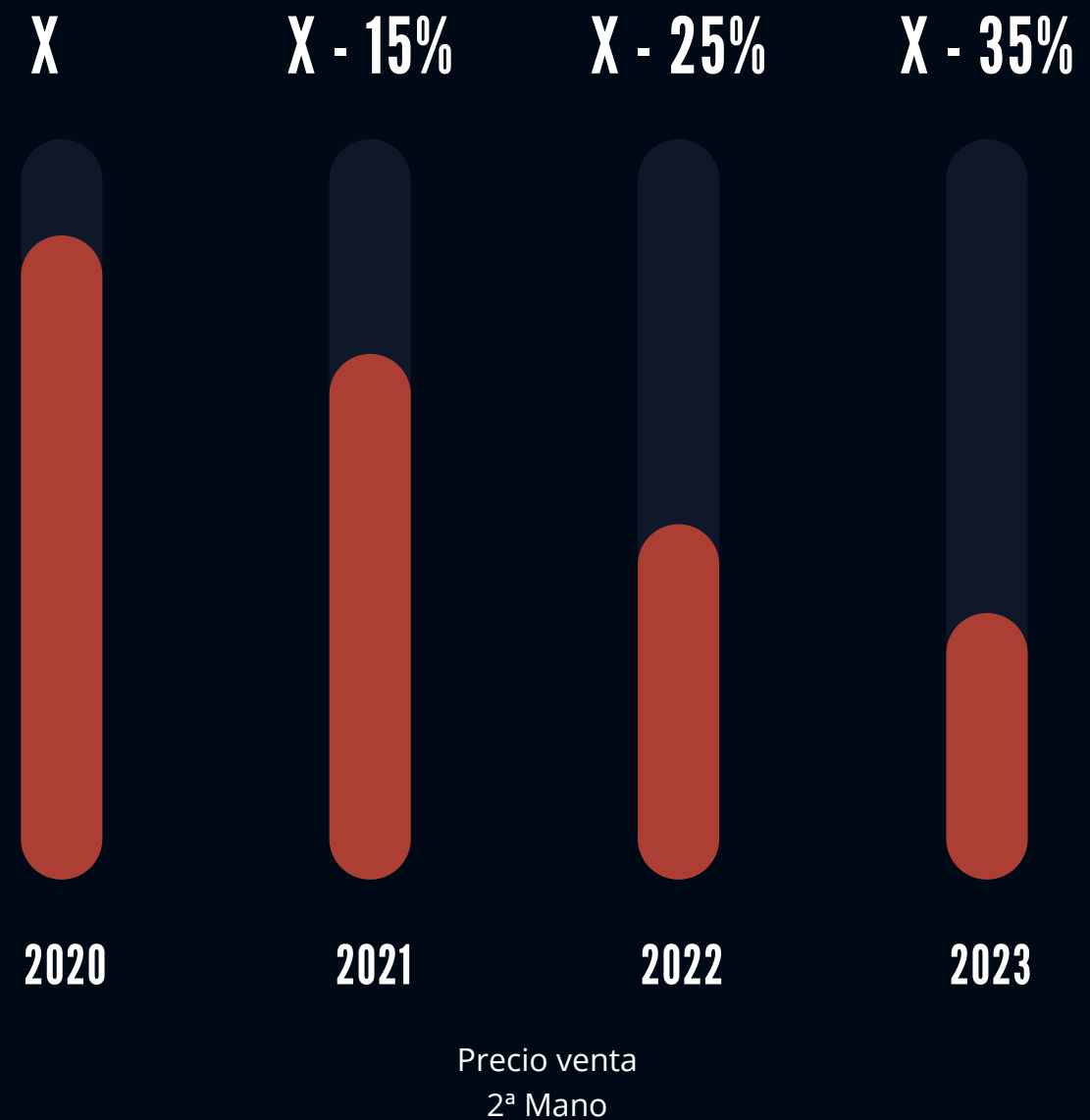
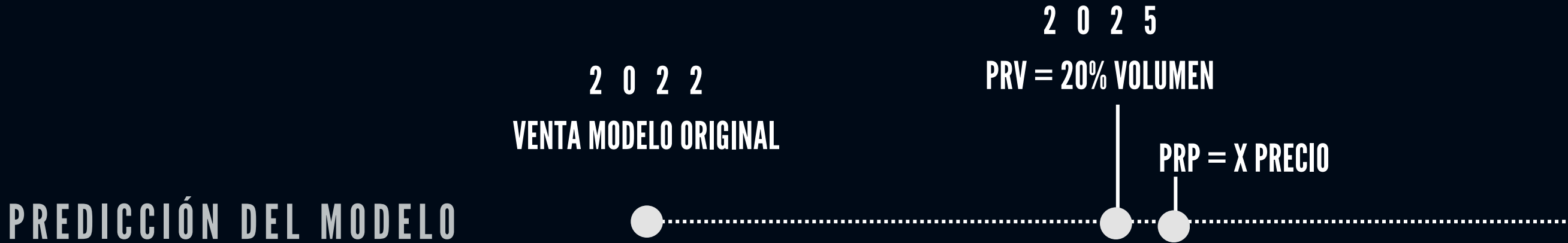


TABLA EJEMPLO  
PRECIO COCHES EN VENTA 2º MANO / AÑOS



MODELO DE PREDICCIÓN

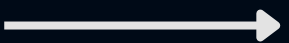
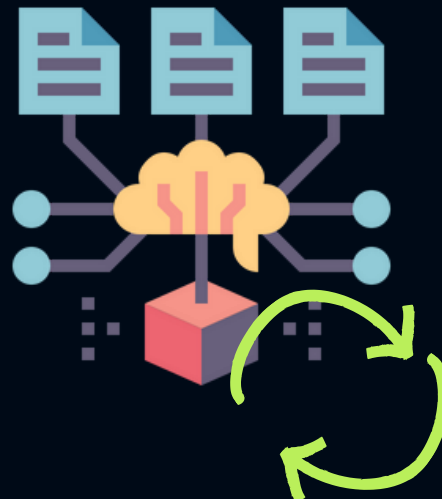


# SOLUCIÓN TECNOLÓGICA

## OPTIMIZACIÓN DATASET



## ENTRENAMIENTO MODELO PREDICTIVO



## COMPARATIVA KPI (PRV / PRP)

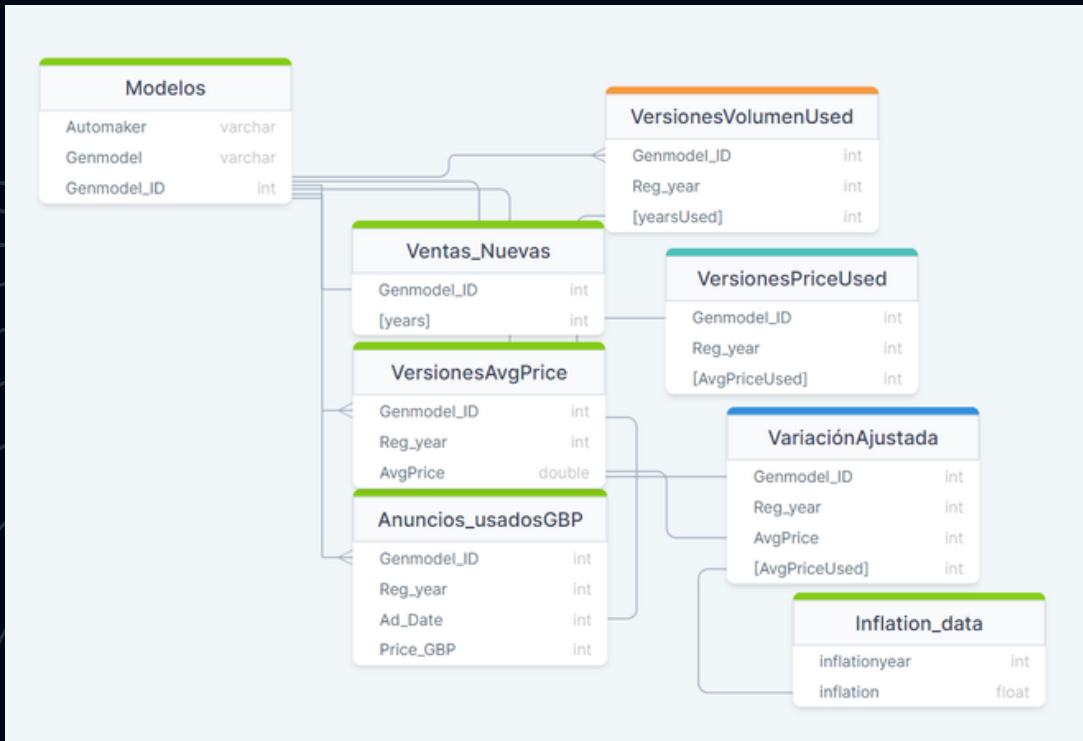


DATASET EN BRUTO

VS



PREDICCIÓN DEL MODELO



```
[ ] #Carga del dataset
anuncios_usados = pd.read_csv("/content/VersiónesVolumenUsed.csv", skipinitialspace=True)
pd.set_option('display.max_rows', 200)

#Establecer los modelos de coche concretos
a = anuncios_usados['Genmodel_ID'].unique()

#Cribamos por el año 2018
prueba = anuncios_usados.loc[anuncios_usados['Ad_Date'] == 2018]

#Creamos una nueva columna para la métrica
prueba['PRV'] = None

#Iteramos para cada modelo de coche y establecemos el valor KAK
for valor in a:
    try:
        val = prueba.loc[prueba['Genmodel_ID'] == valor]
        kl = Kneelocator(val['Reg_year'], val['count'], direction="decreasing")
        prueba.loc[prueba['Genmodel_ID'] == valor, 'PRV'] = kl.x
    except Exception as e:
        prueba.loc[prueba['Genmodel_ID'] == valor, 'PRV'] = None

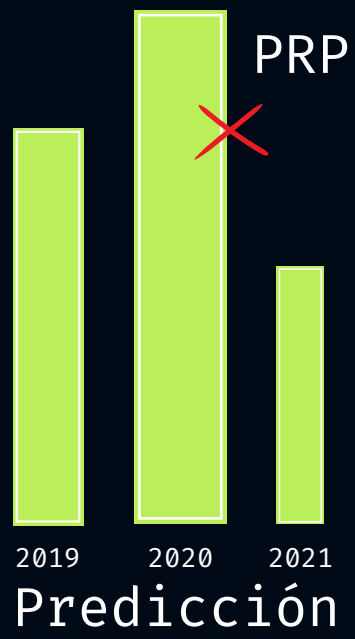
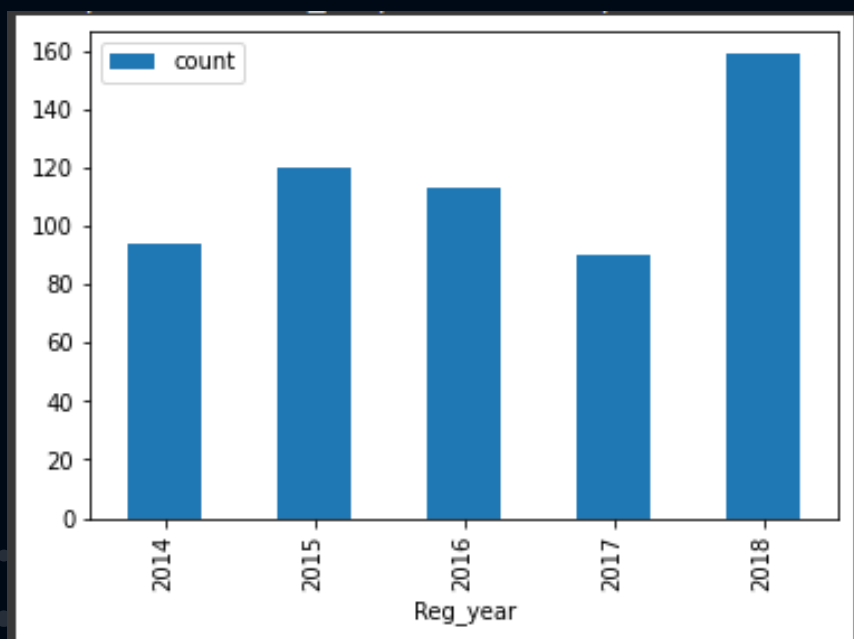
#Eliminamos los valores con Nan
prueba = prueba.dropna(how = 'any', axis=0)

#Establecemos los datos de entrenamiento
datos_entrenamiento = prueba.sample(frac=0.8, random_state=0)

#Establecemos los datos de test
datos_test = prueba.drop(datos_entrenamiento.index)

#Cribamos la etiqueta
etiquetas_entrenamiento = datos_entrenamiento.pop('PRV');
etiquetas_test = datos_test.pop('PRV')

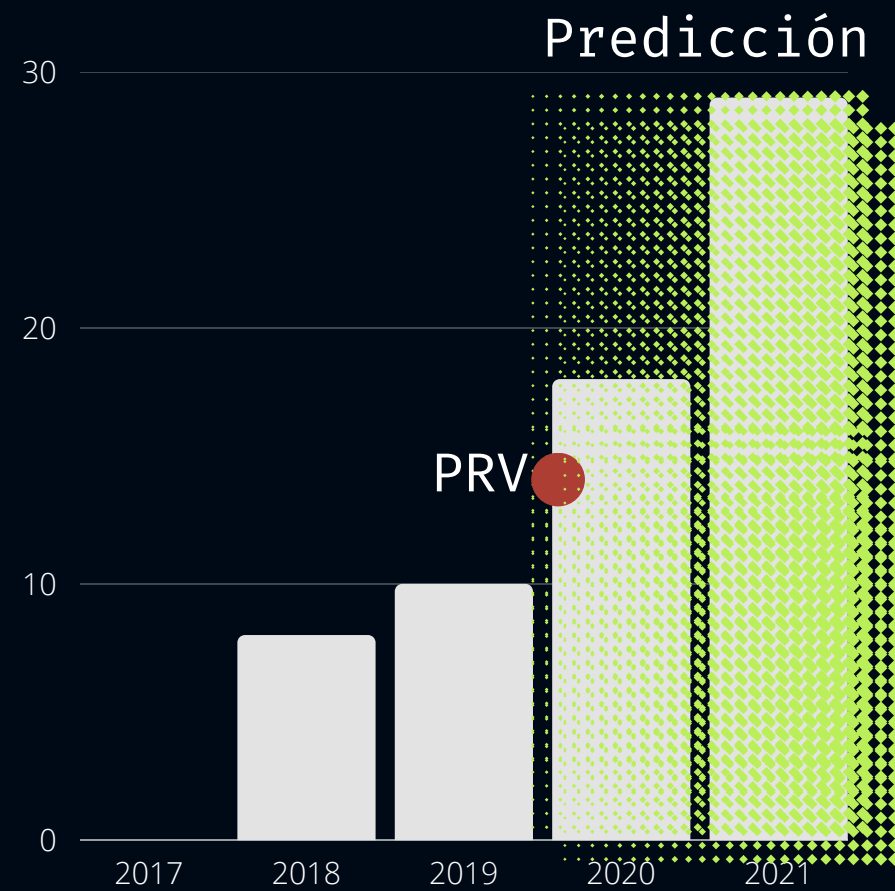
#Entrenamos el modelo, creamos las predicciones y las mostramos
modelo = LinearRegression()
modelo.fit(datos_entrenamiento,etiquetas_entrenamiento);
predicciones = modelo.predict(datos_test)
```



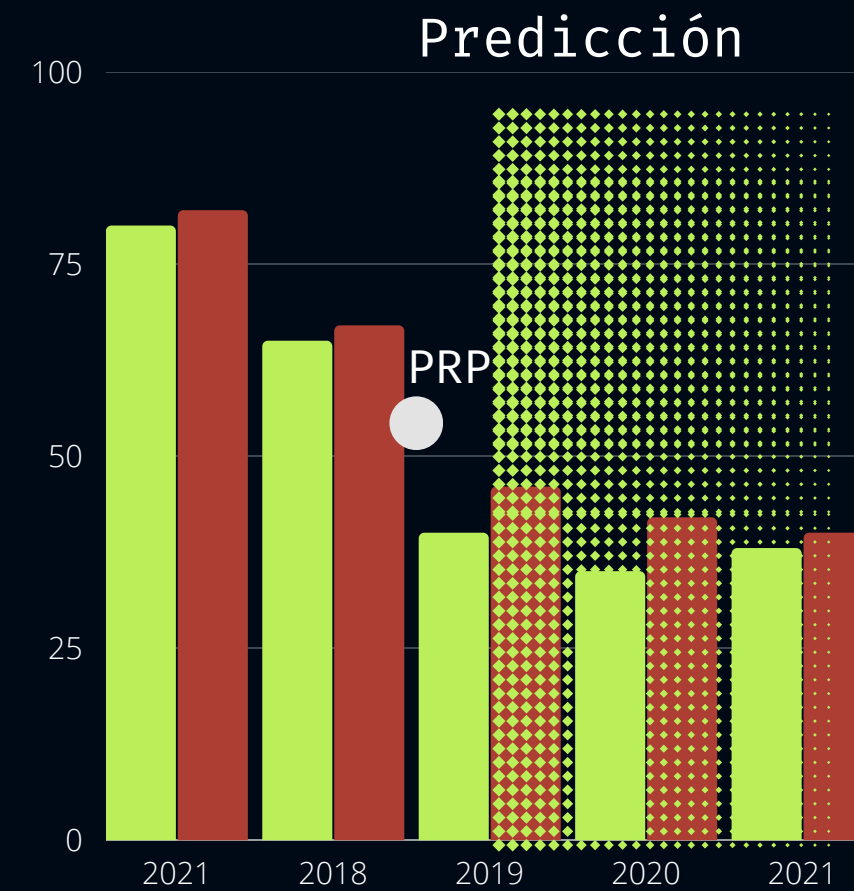


# EJEMPLO DASHBOARD

## VOLUMEN VENTA



## PRECIO



MARCA

MERCEDES BENZ



MODELO

CLASE A



AÑO

2016



T A L E N T   H I G H W A Y

GRACIAS

MOWATAVE - CHECKPOINT 1

