



Desarrollo de un app para la estimación de ventas en la tienda de farmacia Rossmann

Integrantes:

- 1. Edwin Chirre (Data translator)
- 2. Angel Haro (Data Scientist)
- 3. Jaqueline Villamarín (ML Engineer)
- 4. William Chuquer (Developer)

Rossmann es la 2da cadena de farmacia más grande de Alemania

ROSSMANN



Fundado en 1972 por Dirk Rossmann



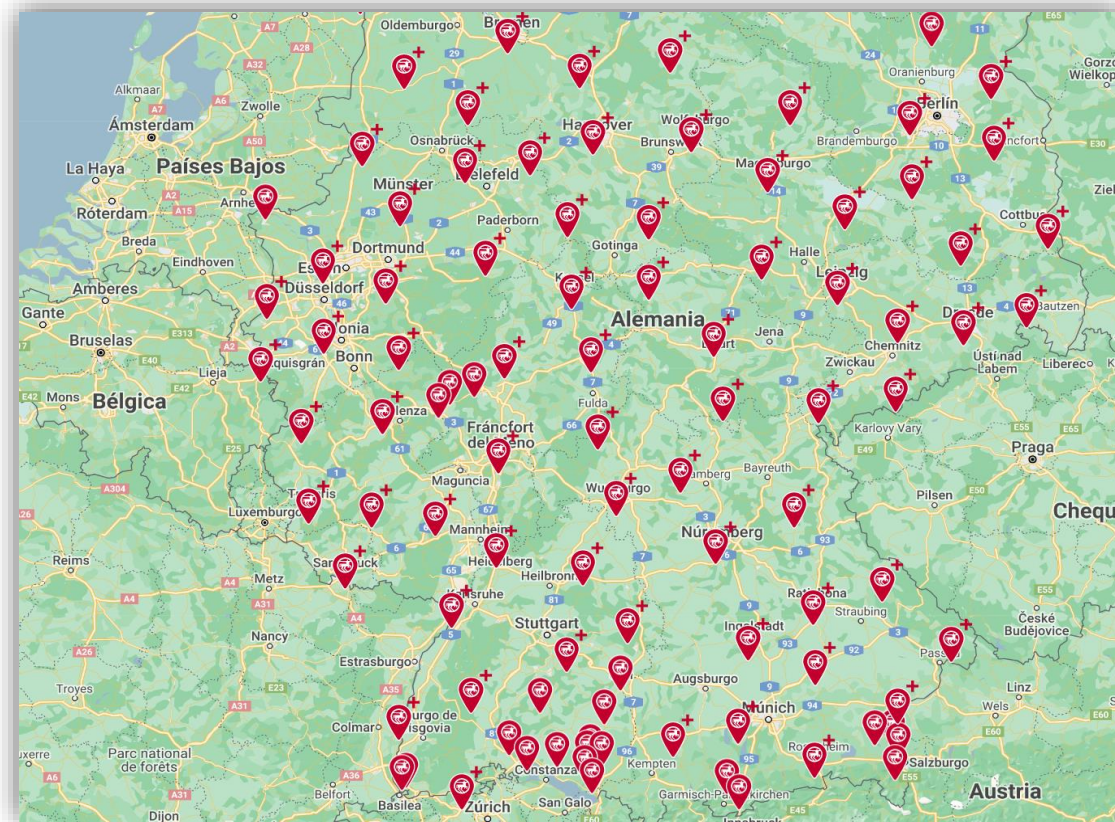
2da cadena de farmacia más grande de Alemania



Tiene más de 3,000 tiendas



Se ubica en 7 países de Europa



Desarrollar app para predecir las ventas de las tiendas



Situación actual

Los gerentes de las tienda predican las ventas diarias bajo su propia experiencia, por lo que no hay uniformidad en dicho cálculo



Objetivo

Predecir de manera confiable y estandarizada las ventas diarias de cada tienda mediante un algoritmo de machine learning



Propuesta

Desarrollar un app donde cada gerente pueda predecir la venta de sus tiendas

Pasos para la creación del formulario de predicción de ventas

Formulario de Predicción de Ventas Tiendas ROSSMANN

Llene el siguiente formulario y compruebe cual será el valor en Dólares probable en la tienda seleccionada



Id Tienda:

1

☒ ¿Tiene Promoción?

☐ ¿Es Feriado Estatal?

Seleccionar Fecha Estimación

2022/11/27

Tipo de feriado estatal:

Ninguno

Predecir Venta de Tienda

La venta estimada para la tienda 1 es \$5025.17 USD

Pasos para crear el formulario analítico

Entendimiento de la data



Creación del modelo



Generación del API



Publicación de la imagen Docker y subir a AWS



Interfaz web (front end)

Entendimiento de la data

Store.csv

Base de datos de 1115 Tiendas

Train.csv

Base de datos de Ventas de cada tienda. Histórico de 01ENE2013 hasta el 31JUL2015, más de 1 millón de registros

Variables	
store.csv	train.csv
Store	Store
StoreType	DayOfWeek
Assortment	Date
CompetitionDistance	Sales
CompetitionOpenSinceMonth	Customers
CompetitionOpenSinceYear	Open
Promo2	Promo
Promo2SinceWeek	StateHoliday
Promo2SinceYear	SchoolHoliday
PromoInterval	

Variable de Predicción

Creación del modelo

Ficha técnica



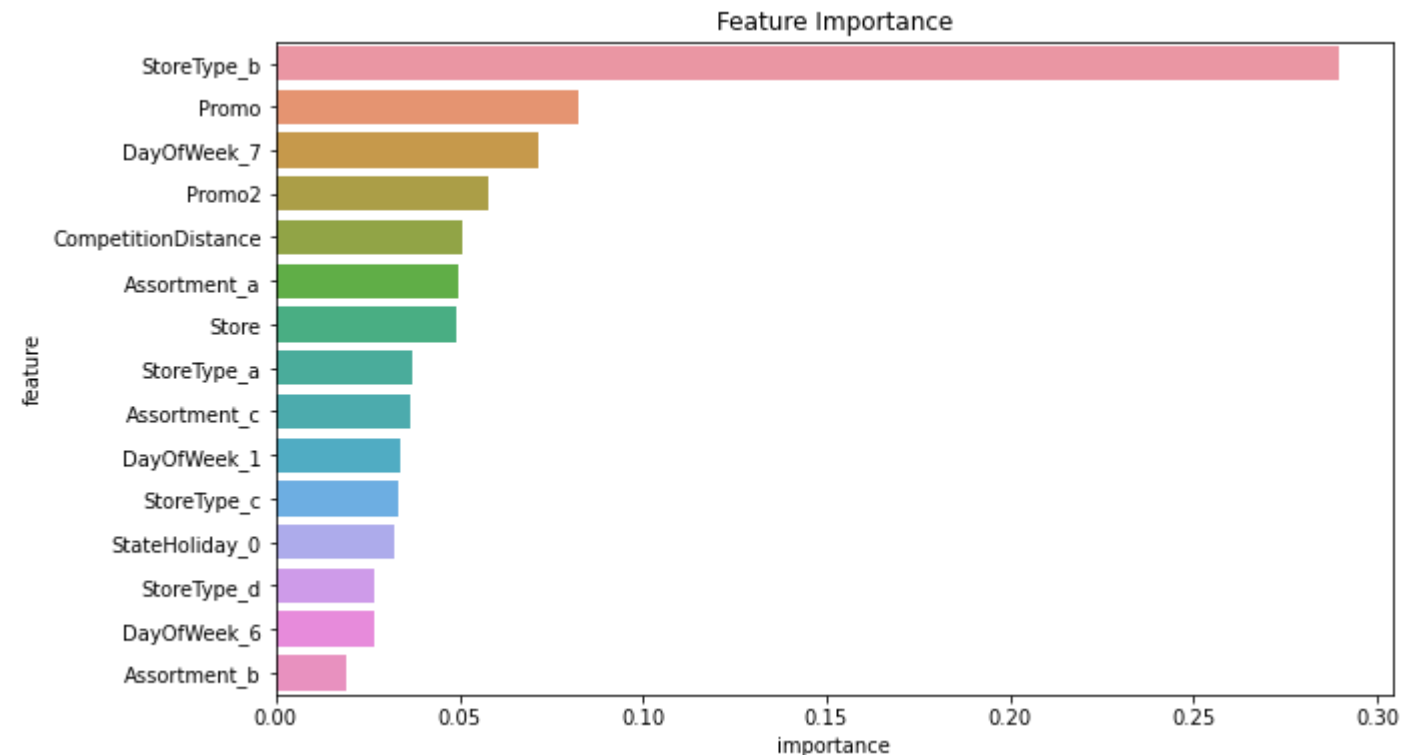
Se realizó un modelo Xgboost regressor



Se consideraron 30 variables



Tiene un RMSE de 370



Generación del API con FastAPI (Métodos Rest)

→ ↺ 🏠

🔒 https://ventaproductos-container.mrcbe3o1f4emc.us-east-2.cs.amazonlightsail.com/docs

80% ☆

📁 ⬇️ ☰ 📄

FastAPI

0.1.0 OAS3

/openapi.json

default

POST

/sales-predictions

Generar Prediccion Sales

GET

/

Model Info

GET

/health

Service Health

security-admin

POST

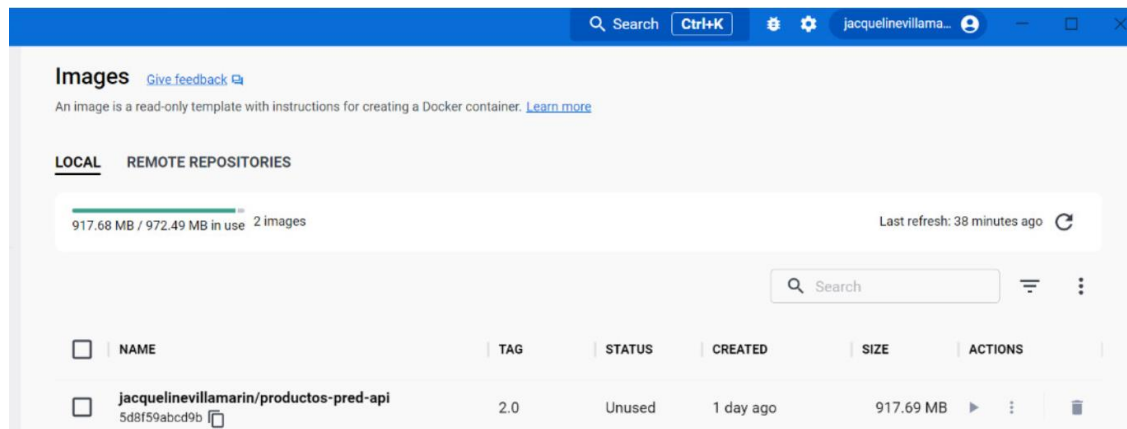
/admin/api-keys

Generar Nuevo Api Key

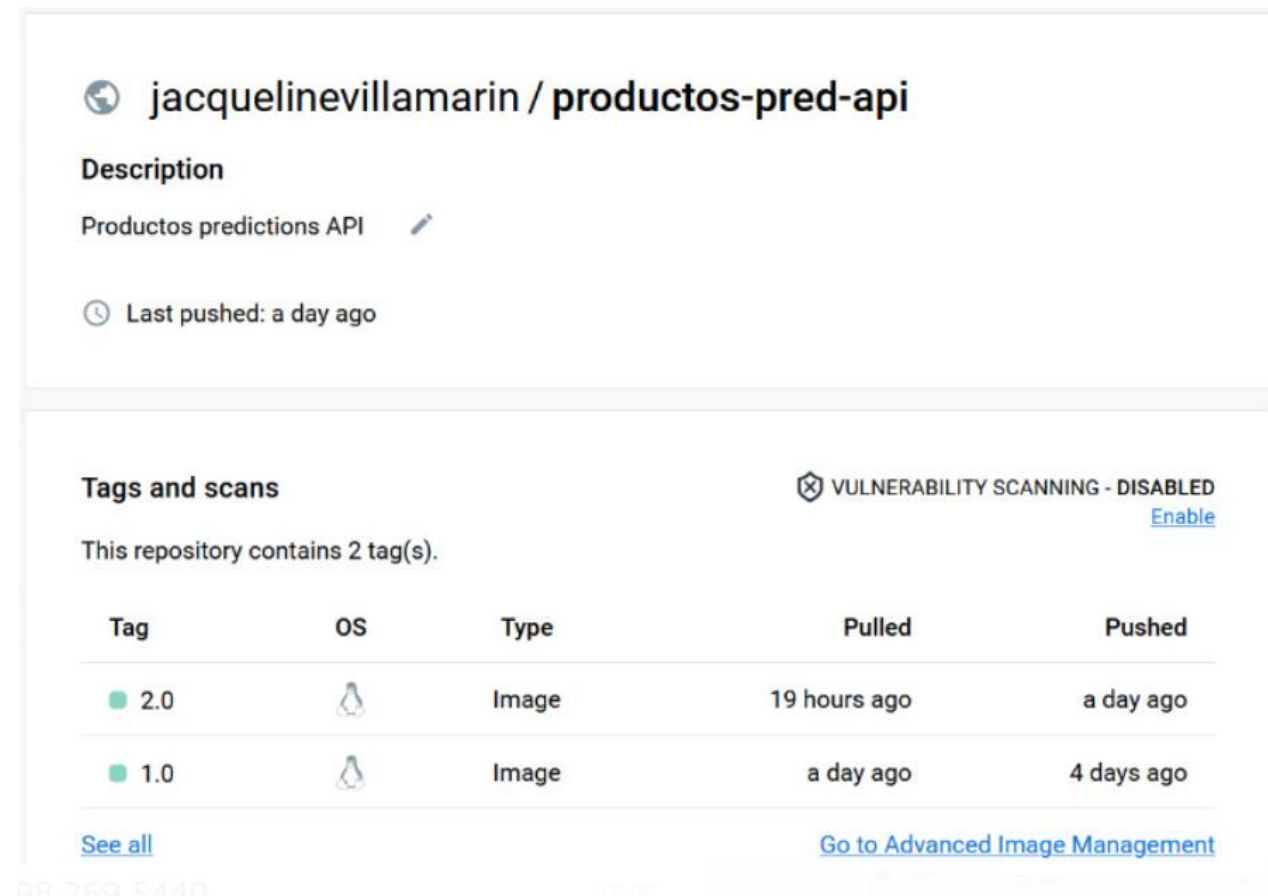
```
class Input(BaseModel):
    Store:int #variable ingresa usuario
    Promo:int #variable ingresa usuario
    SchoolHoliday:int #variable ingresa usuario
    CompetitionDistance:float #Campo que se obtiene de los datos del store
    CompetitionOpen:float # Campo Calculado en función de datos del Store
    Promo2:int #Campo que se obtiene de los datos del store
    Promo2Open:float # Campo Calculado en función de datos del Store
    IsPromo2Month:int # Campo Calculado en función de datos del Store
    Day:int #variable ingresa usuario (calculada en función de la fecha ingresada)
    Month:int #variable ingresa usuario (calculada en función de la fecha ingresada)
    Year:int #variable ingresa usuario (calculada en función de la fecha ingresada)
    WeekOfYear:int #variable ingresa usuario (calculada en función de la fecha ingresada)
    DayOfWeek_1:float #variable ingresa usuario (calculada en función de la fecha ingresada)
    DayOfWeek_2:float #variable ingresa usuario (calculada en función de la fecha ingresada)
    DayOfWeek_3:float #variable ingresa usuario (calculada en función de la fecha ingresada)
    DayOfWeek_4:float #variable ingresa usuario (calculada en función de la fecha ingresada)
    DayOfWeek_5:float #variable ingresa usuario (calculada en función de la fecha ingresada)
    DayOfWeek_6:float #variable ingresa usuario (calculada en función de la fecha ingresada)
    DayOfWeek_7:float #variable ingresa usuario (calculada en función de la fecha ingresada)
    StateHoliday_0:float #variable ingresa usuario
    StateHoliday_a:float #variable ingresa usuario
    StateHoliday_b:float #variable ingresa usuario
    StateHoliday_c:float #variable ingresa usuario
    StoreType_a:float #Campo que se obtiene de los datos del store
    StoreType_b:float #Campo que se obtiene de los datos del store
    StoreType_c:float #Campo que se obtiene de los datos del store
    StoreType_d:float #Campo que se obtiene de los datos del store
    Assortment_a:float #Campo que se obtiene de los datos del store
    Assortment_b:float #Campo que se obtiene de los datos del store
    Assortment_c:float #Campo que se obtiene de los datos del store
```


Publicación de la imagen en Docker

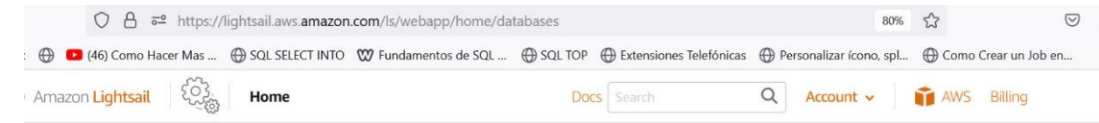
Docker Desktop



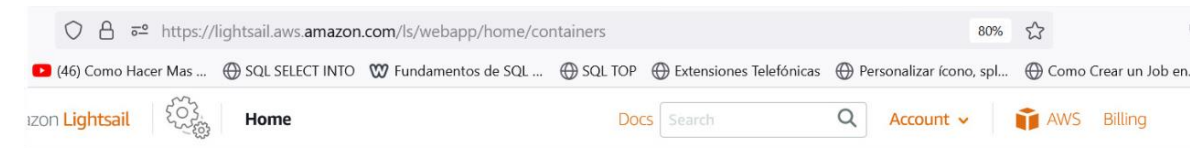
Docker HUB



Servicios en la Nube (AWS)



Base de Datos



Contenedor



Current deployment

Your deployment is the set of containers currently running on your container service.

[Learn more about deployments](#)

[Modify your deployment](#)



Variables de configuración Contenedor

Deployment versions

All containers deployed to your container service are recorded as versions. You can roll back your current deployment to a previous version.

[Learn more about deployment versions](#)

Date	Version	Status
December 5, 2022 - 3:12 PM	3	Active
December 5, 2022 - 9:05 AM	2	Inactive
December 5, 2022 - 8:45 AM	1	Failed

3 of 3 items

Interfaz Web para consumir el modelo y disponibilizar a los usuarios (Streamlit)

Formulario de Predicción de Ventas Tiendas ROSSMANN

Llene el siguiente formulario y compruebe cual será el valor en Dólares probable en la tienda seleccionada



Id Tienda:	<input checked="" type="checkbox"/> ¿Tiene Promoción?
<input type="text" value="1"/>	<input type="checkbox"/> ¿Es Feriado Estatal?
Seleccionar Fecha Estimación	Tipo de feriado estatal:
<input type="text" value="2022/11/27"/>	<input type="text" value="Ninguno"/>
<input type="button" value="Predecir Venta de Tienda"/>	
La venta estimada para la tienda 1 es \$5025.17 USD	

```
import pandas as pd
import streamlit as st
import datetime as dt
import ParametrosPrediccion as pp
import requests
import json
import pickle

API_URLBASE = "https://ventaproductos-container.mrcbe3o1f4emc.us-east-2.cs.amazonlightsail.com/"
API_KEY = "3N3YtdacZPUfke8fI5L98FcQk1QFqz10AeiNMkABHxaLkteNuWUvvsf3I53R6mls411dZZdt7PnmBJ2VKYyA5CARoR2Eca_G20kdA"

def execute_prediction_request(Store: int, Promo: int, SchoolHoliday: int, CompetitionDistance: float,
                               CompetitionOpen: float, Promo2: int, Promo2Open: float, IsPromo2Month: int,
                               Day: int, Month: int, Year: int, WeekOfYear: int, DayOfWeek_1: float,
                               DayOfWeek_2: float, DayOfWeek_3: float, DayOfWeek_4: float,
                               DayOfWeek_5: float, DayOfWeek_6: float, DayOfWeek_7: float,
                               StateHoliday_0: float, StateHoliday_a: float, StateHoliday_b: float, StateHoliday_c: float,
                               StoreType_a: float, StoreType_b: float, StoreType_c: float, StoreType_d: float,
                               Assortment_a: float, Assortment_b: float, Assortment_c: float) -> float:

    payload = {
        'Store': int(Store),
        'Promo': int(Promo),
        'SchoolHoliday': int(SchoolHoliday),
        'CompetitionDistance': CompetitionDistance,
        'CompetitionOpen': CompetitionOpen,
        'Promo2': int(Promo2),
        'Promo2Open': Promo2Open,
        'IsPromo2Month': int(IsPromo2Month),
        'Day': int(Day),
        'Month': int(Month),
        'Year': int(Year),
```

QR's de la solución analítica



Enlace del API



Enlace del interfaz web



*Github de la solución
(click en el ícono)*