

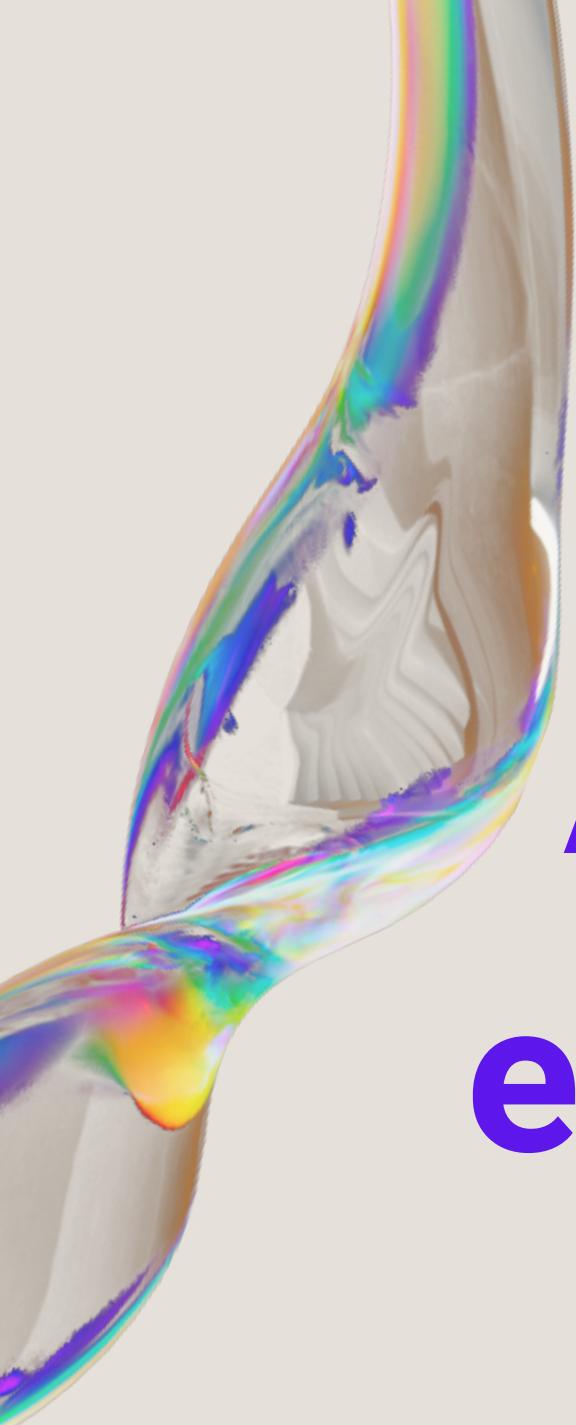
# ARTEFACT Project

## Car Price Prediction & Damage Detection

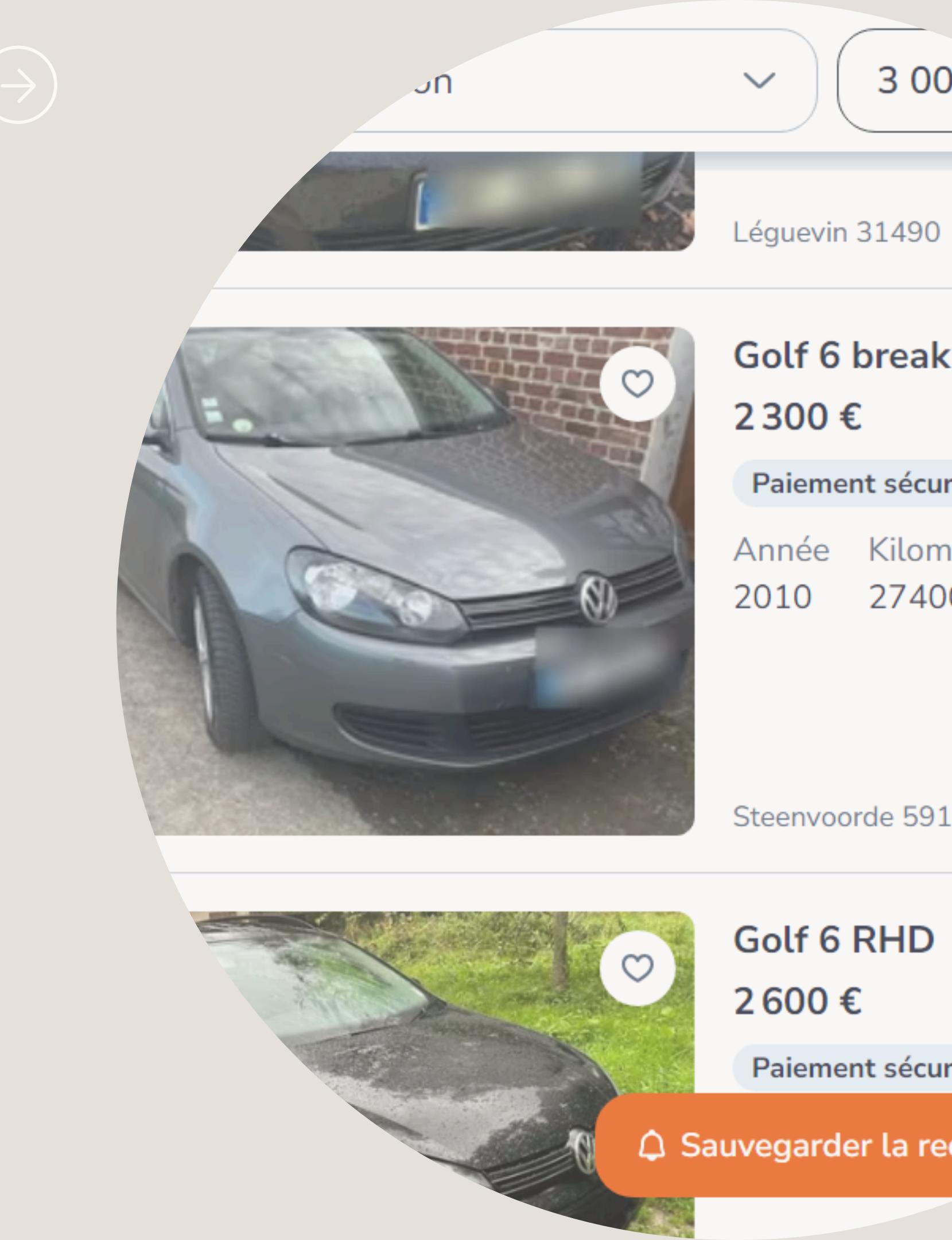
*Lissandre*

*Rova*

*Hevi*



Avez-vous déjà  
essayé de vendre  
votre voiture?



3 00

Léguvin 31490

Golf 6 break

2 300 €

Paiement sécurisé

Année Kilométrage

2010 27400

Steenvoorde 591

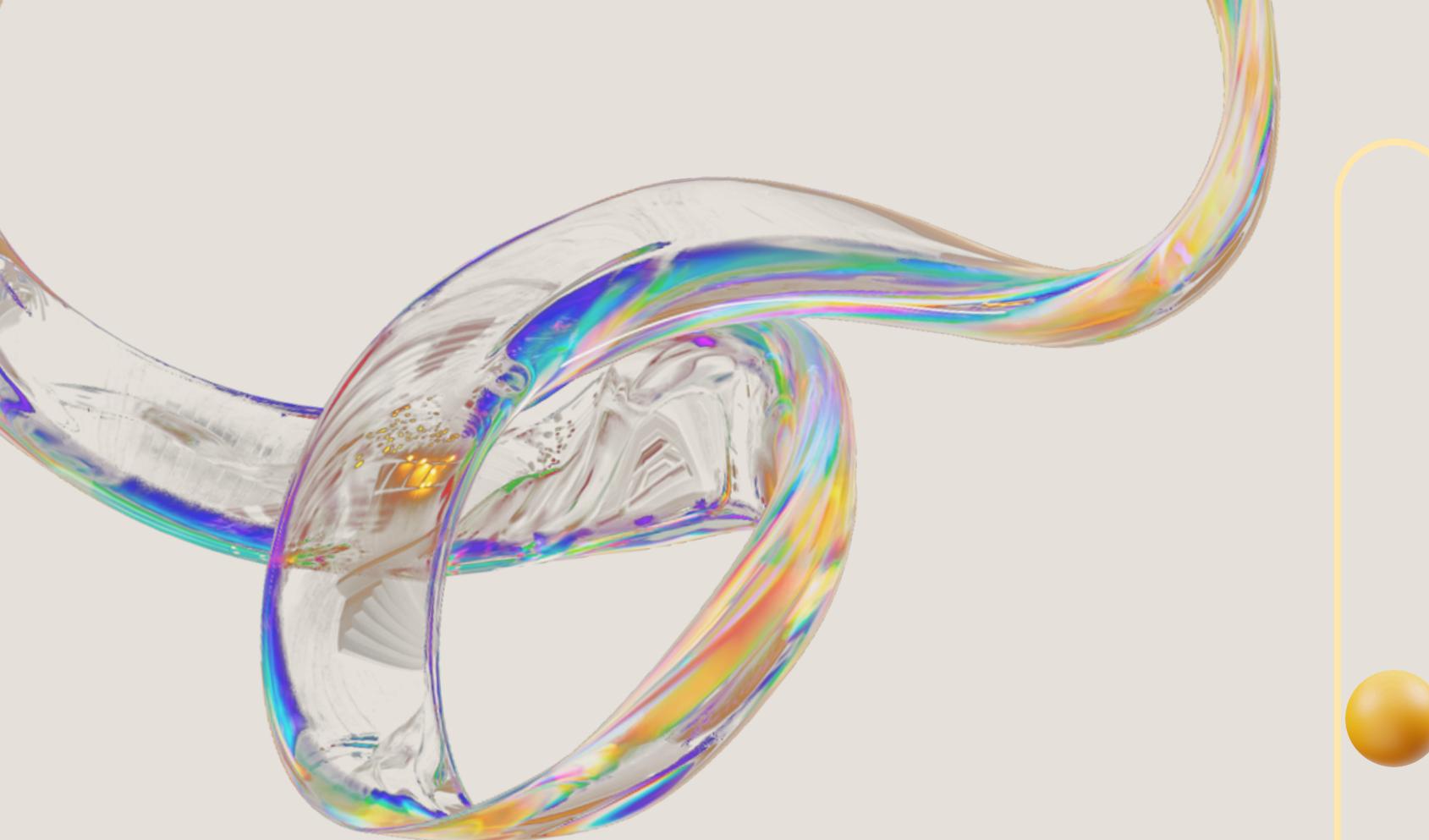
Golf 6 RHD

2 600 €

Paiement sécurisé

Sauvegarder la recherche

# DATASETS



## Données Tabulaires :

Un dataset de voitures d'occasion vendues aux enchères de 1970 à 2015 :

- Année
- Marque
- Modèle
- Couleur
- Transmission
- Kilométrage
- État du véhicule
- Intérieur
- ...

# DATASETS

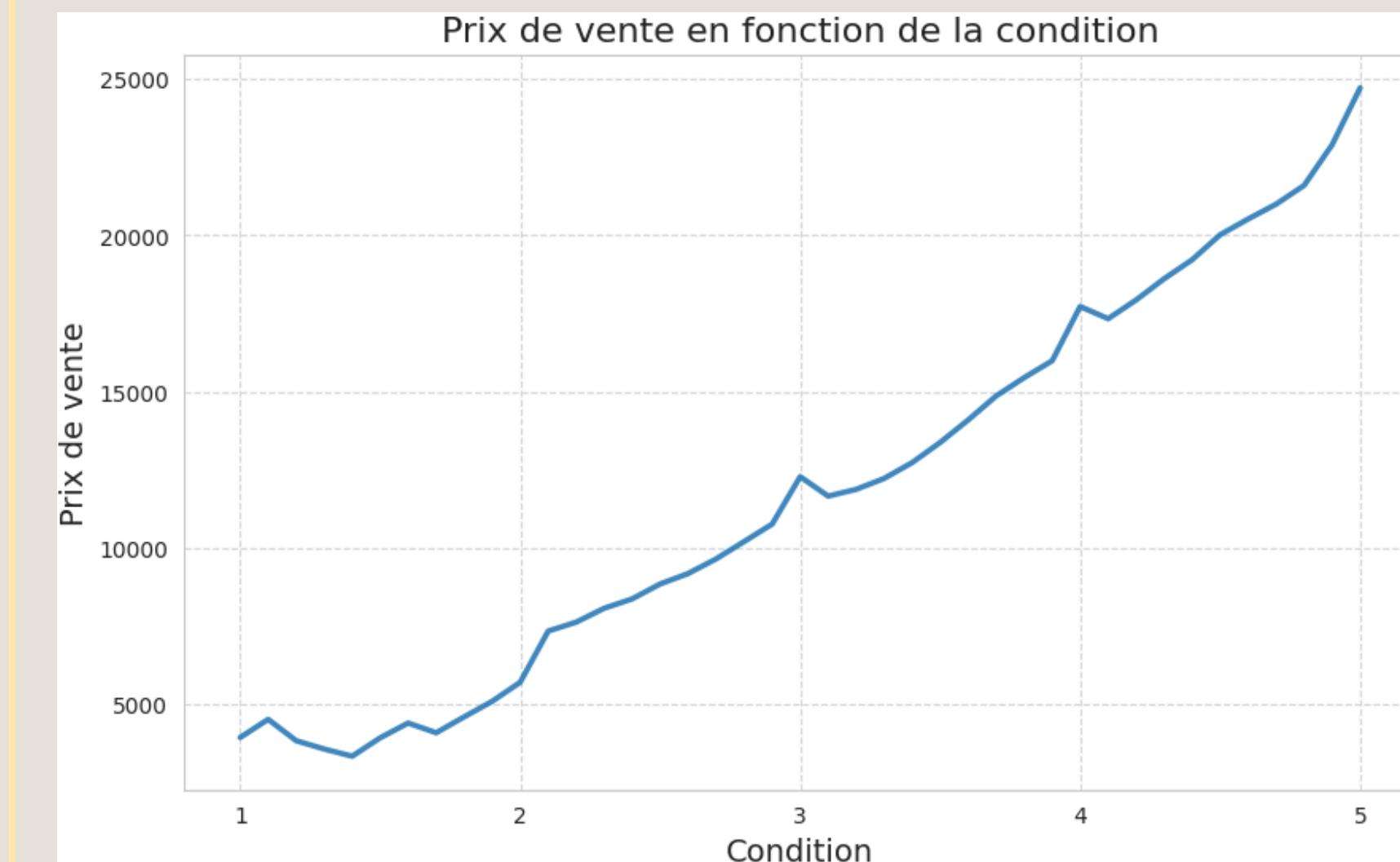


## Données Tabulaires :

Un dataset de voitures d'occasion vendues aux enchères de 1970 à 2015 :

- Année de production
- Marque
- Modèle
- Couleur
- Transmission
- Kilométrage
- État du véhicule
- Intérieur
- ...

## L'influence de l'état du véhicule sur son prix :



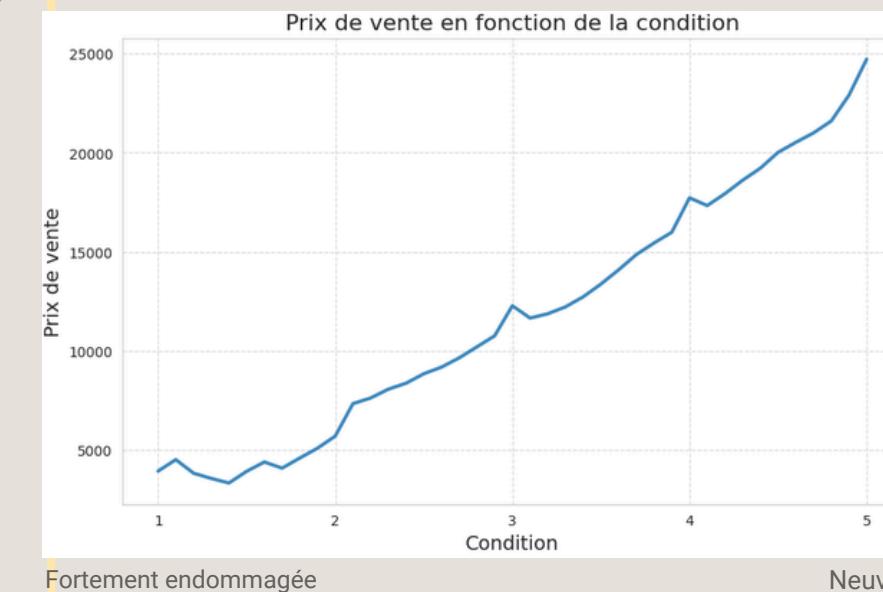
Fortement endommagée

Neuve

# DATASETS



L'influence de l'état du véhicule sur son prix :



Données visuelles :

Un ensemble de 997 photos de voitures, de l'état neuf à très endommagé, avec des annotations précisant les dommages :



Voiture Neuve



Voiture fortement endommagée

# STRUCTURE



## Input Utilisateur

### Caractéristiques du véhicule

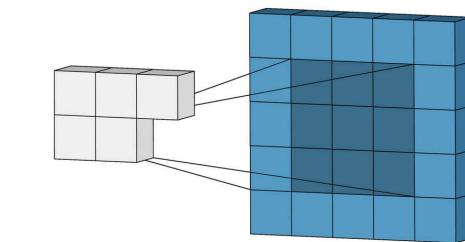
- Année
- Kilométrage
- Marque
- Modèle
- ...

### Photo du véhicule



## Classification

Analyse d'image pour identifier la condition du véhicule



Réseau de Neurones Convolutif (CNN)

# STRUCTURE

## Input Utilisateur

### Caractéristiques du véhicule

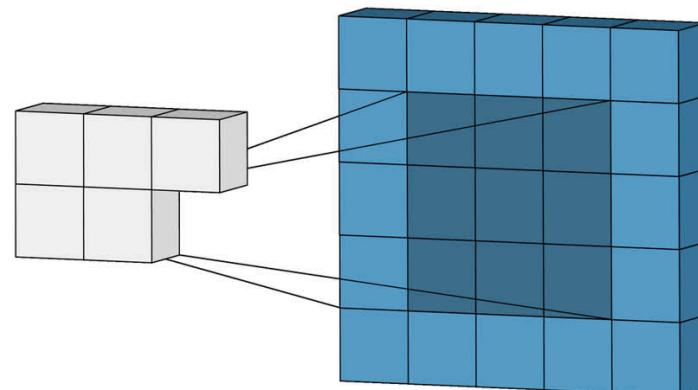
- Année
- Kilométrage
- Marque
- Modèle
- ...

### Photo du véhicule



## Classification

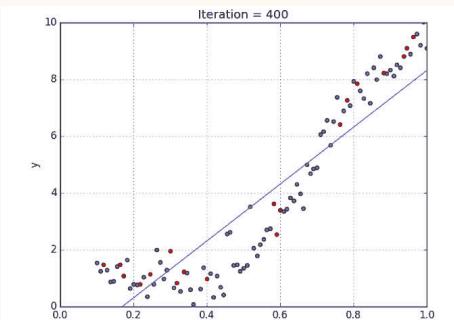
Analyse d'image pour identifier la condition du véhicule



Réseau de Neurones Convolutif (CNN)

## Régression

### Prédiction du prix basée sur les caractéristiques



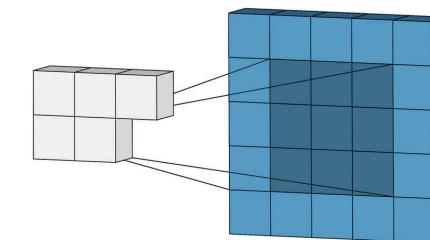
## XGBoost

Algorithme combinant des arbres de décision pour corriger progressivement les erreurs

# STRUCTURE

## Classification

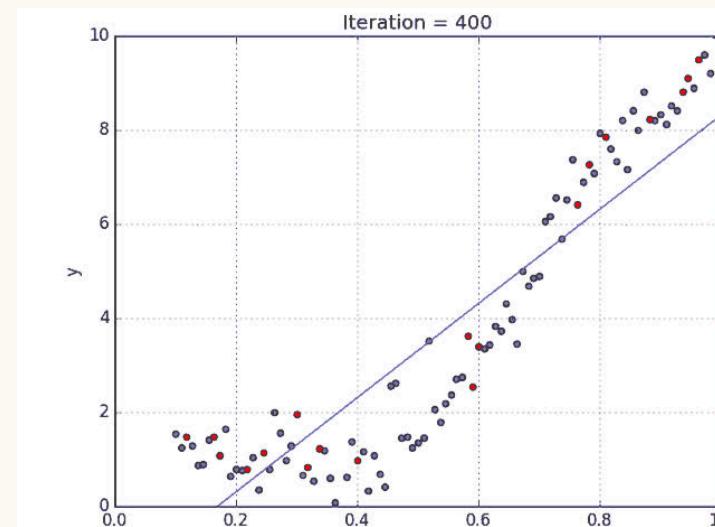
Analyse d'image pour identifier la condition du véhicule



Réseau de Neurones Convolutif (CNN)

## Régression

Prédiction du prix basée sur les caractéristiques



## *XGBoost*

Algorithme combinant des arbres de décision et corigeant progressivement les erreurs

## Segmentation

Déttection des dommages par analyse d'image

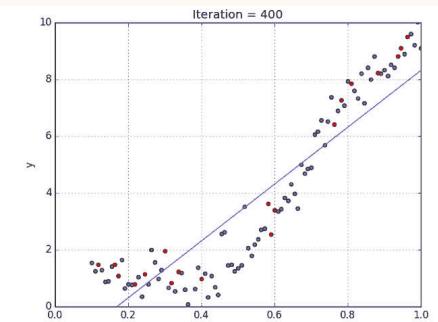


Entraînement d'un algorithme YOLO apprenant à partir d'images annotées

# STRUCTURE

## Régression

Prédiction du prix basée sur les caractéristiques



### XGBoost

Algorithme combinant des arbres de décision pour corriger progressivement les erreurs

## Segmentation

Détection des dommages par segmentation



Entraînement d'un algorithme YOLO apprenant à partir d'images annotées

## Output

Prix de revente estimé



Photo annotée des dommages



# STRUCTURE

## Segmentation

Détection des dommages par analyse d'image



Entraînement d'un algorithme YOLO apprenant à partir d'images annotées

## Output

Prix de revente estimé



Photo annotée des dommages



# STRUCTURE

## Input Utilisateur

### Caractéristiques du véhicule

- Année
- Kilométrage
- Marque
- Modèle
- ...

### Photo du véhicule



**XGBoost**

## Output

### Prix de revente estimé



### Photo annotée des dommages



# TRAITEMENT D'IMAGE



# TRAITEMENT D'IMAGE

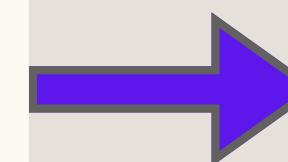


# TRAITEMENT D'IMAGE



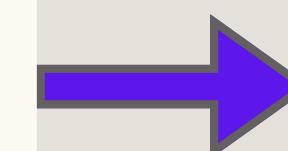
## ÉVALUATION & RÉSULTATS

*MODÈLE DE CLASSIFICATION  
(CNN)*



Prédictions correctes :  
**85 %**

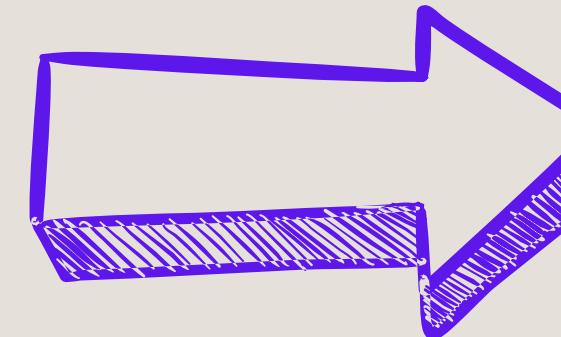
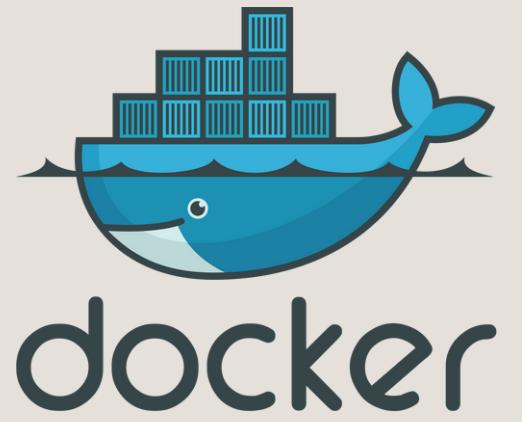
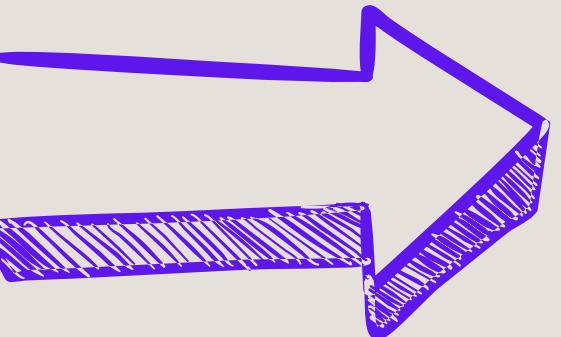
*MODÈLE DE RÉGRESSION  
(XGBOOST)*



Ecart moyen entre la prédition et la  
valeur réelle:  
**724 \$**

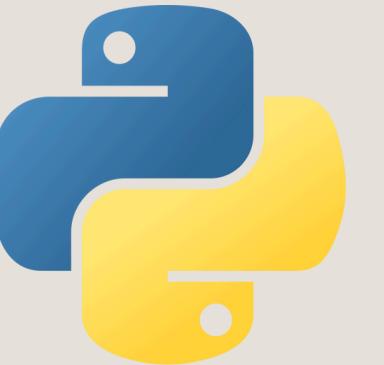
# DÉPLOIEMENT

A screenshot of a web application interface titled "Estimez votre véhicule". The form includes fields for "Année" (Year), "Kilométrage (km)" (Mileage), "Marque" (Brand), "Modèle" (Model), "Type de carrosserie" (Body type), "Transmission", "État (US)" (State), "Couleur" (Color), and "Intérieur" (Interior). There is also a file upload field for a vehicle photo and a "Obtenir une estimation" (Get an estimate) button.



A code editor interface showing a snippet of Python code. The code uses the requests library to check a response status code and BeautifulSoup to parse HTML content, specifically looking for images with the alt attribute "Post image".

```
1 # checking response.status_code (if you get 502, try rerunning the code)
2 if response.status_code != 200:
3     print(f"Status: {response.status_code} - Try rerunning the code")
4 else:
5     print(f"Status: {response.status_code}")
6
7     # using BeautifulSoup to parse the response object
8     soup = BeautifulSoup(response.content, "html.parser")
9
10    # finding Post images in the soup
11    images = soup.find_all("img", attrs={"alt": "Post image"})
12
13    # saving the images to a folder named 'post_images'
14    for image in images:
15        image_url = image['src']
16        response = requests.get(image_url)
17        with open(os.path.join('post_images', os.path.basename(image_url)), 'wb') as f:
18            f.write(response.content)
```



# DÉMONSTRATION

<https://car-app-867908798687.europe-central2.run.app/>

# OBSTACLES RENCONTRÉS



**Traitement des images :**  
Processus long et manuel

**Entraînement du modèle YOLO :**  
Temps de calcul élevé et coûteux en ressource

**Déploiement sur le cloud :**  
Complexité d'intégration et d'optimisation

# AXES D'AMÉLIORATION



## Enrichissement des données :

- Compléter avec des données plus récentes, hors US et incluant l'état moteur du véhicule

## Optimisation des modèles :

- Développer d'autres modèles de reconnaissance (couleur, marque, ..)

## Amélioration UX / UI :

- Réduire le nombre d'inputs requis
- Optimiser le temps de chargement

Thank  
you!

