



27/02/2024

Réalisez un dashboard et assurez une veille technique

Armand FAUGERE [LinkedIn](#)

armand-faugere@live.fr

Sommaire

- I) Cadrage du projet et données d'entrée
- II) Dashboard interactif
- III) Veille technique
- IV) Conclusion



I) Cadrage du projet et données d'entrée

I) Cadrage du projet et données d'entrée



❑ Contexte :

- Projet de mise en œuvre d'un outil de «scoring credit»
- ➔ probabilité qu'un client rembourse son crédit
- ➔ classification de la demande en « accordée » ou « refusée ».
- Proposition de crédits à la consommation pour des personnes ayant peu ou pas du tout d'historique de prêt
- ➔ Risque financier important pour l'entreprise Prêt à dépenser

❑ But :

- Réaliser un dashboard interactif et le déployer sur le cloud, et réaliser une veille technique

❑ Objectifs :

- À partir du projet précédent élaborer un dashboard avec streamlit
- Respecter le principe **WCAG**
- Tester et déployer le dashboard avec streamlit io
- Réaliser une veille technique sur le vision transformer (vit)
- Réaliser un POC
- Réaliser une note technique

➔ application réalisée avec fastapi, déployée sur render.com

The screenshot shows the Render dashboard interface. At the top, there's a navigation bar with links for Dashboard, Blueprints, Env Groups, Docs, Community, and Help. A 'New +' button and a user profile icon are on the right. Below the navigation bar, the 'Overview' section is visible, featuring a search bar for services. A table lists the deployed services. The table has columns for Service Name, Status, Type, Runtime, Region, and Last Deployed. One service, 'aplafgha', is listed with a status of 'Deployed', type of 'Web Service', runtime of 'Python 3', region of 'Frankfurt', and last deployed '17 days ago'.

SERVICE NAME	STATUS	TYPE	RUNTIME	REGION	LAST DEPLOYED
aplafgha	Deployed	Web Service	Python 3	Frankfurt	17 days ago

II) Dashboard interactif

2) Dashboard interactif



Jupyter Notebook, Python, Pandas, Numpy, Matplotlib, Seaborn, sklearn, plotly, streamlit

Elaboration de l'architecture souhaitée

Développement de premiers POC

Développement de l'interface en local

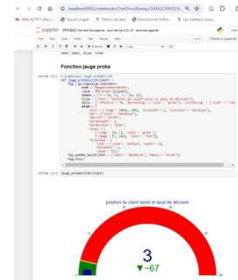
Déploiement sur le cloud

- ☐ draft notebook
- ☐ draft interface streamlit
- ☐ création d'un dépôt local git

- ☐ Développement du contenu de l'application sur un note book :
 - chargement des données
 - graphiques plotly

- ☐ Avec streamlit et plotly, élaboration et validation du code en local
 - mise en page
 - graphiques
 - interface utilisateur
 - ...
 - tests en local

- ☐ Mise à jour du dispo distant github
- ☐ connexion github et streamlit io
- ☐ déploiement avec streamlit io
 ➔ <https://projet8af-g2dxwn8gnbyjaz2ewdropb.streamlit.app/>



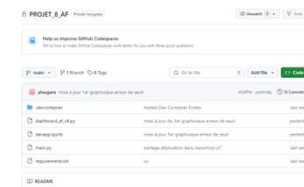
```

import streamlit as st
import pandas as pd
import plotly.express as px

# Chargement des données
df = pd.read_csv('data.csv')

# Création d'un graphique
fig = px.pie(df, names='category', values='value')

# Affichage du graphique
st.plotly_chart(fig)
    
```



1
Mise en place d'une interface utilisateur WCAG

2) Dashboard interactif



```
1 import numpy as np
2 import pandas as pd
3 import streamlit as st
4 import json
5 import requests
6 import plotly
7 import plotly.express as px
8 import plotly.graph_objects as go
9 from plotly.subplots import make_subplots
10
11 def main():
12     # liens vers application
13
14     URL_PRED = "https://apiafgha.onrender.com/predict"
15     URL_IMPORTANCE = "https://apiafgha.onrender.com/interpretation"
16
17     # Accueil
18     st.image("https://capirossi.org/wp/wp-content/uploads/2019/04/Fonctionnement-du-modèle-de-scoring.png")
19     st.title("Evaluation Solvabilité Client")
20     st.header("Analyse des résultats du modèle de scoring")
21     st.info("Basé sur un échantillon de 200 clients allant de 0 à 199,")
22     st.markdown("___")
23     st.markdown("___")
24
25     """
26     ### Objectifs
27     ##### Déterminer la solvabilité Client (Solvable/Non Solvable)
28     """
```

Lien vers l'API
déployée sur render

Le code a été élaboré pour mettre à disposition une interface :

- ergonomique
- interactive
- explications claires
- WCAG
- informations générales et détaillées

2) Dashboard interactif



Sidebar pour interactivité

Mode d'emploi :

- 1) Choisir l'index client avec la sidebar
- 2) Choisir un critère (feature) pour l'analyser avec la position du Client
- 3) Choisir les axes pour analyser le comportement global des critères(features)

select index of client bellow 🎯

Index Client

0 107 199

select feature bellow 📊

select

EXT_SOURCE_1

select Axes bellow 📐

select axe X

EXT_SOURCE_1

select axe Y

EXT_SOURCE_1

Résultat : accepté ✅

Mode d'emploi

Sélection du client avec slide bar

Sélection de la feature à analyser

Sélection des features à comparer

Deuxième affichage du statut client

Affichage des informations (graphiques...)



Accueil avec rappel des objectifs

Evaluation Solvabilité Client

Analyse des résultats du modèle de scoring

Basé sur un échantillon de 200 clients allant de 0 à 199, le seuil de décision est à 7

Objectifs

- Determiner la solvabilité Client (Solvable/Non Solvable)
- Situer le Client vis à vis du seuil de décision et des autres Clients
- Situer le Client sur les critères qui ont abouti à la décision

Résultat : accepté ✅

Jauge de score client avec seuil de décision :

2) Dashboard interactif



Jauge de score client avec seuil de décision :

Seuil : 7

Score Client : 3.067

☐ Permet de visualiser le score du client et le seuil de décision



2) Dashboard interactif

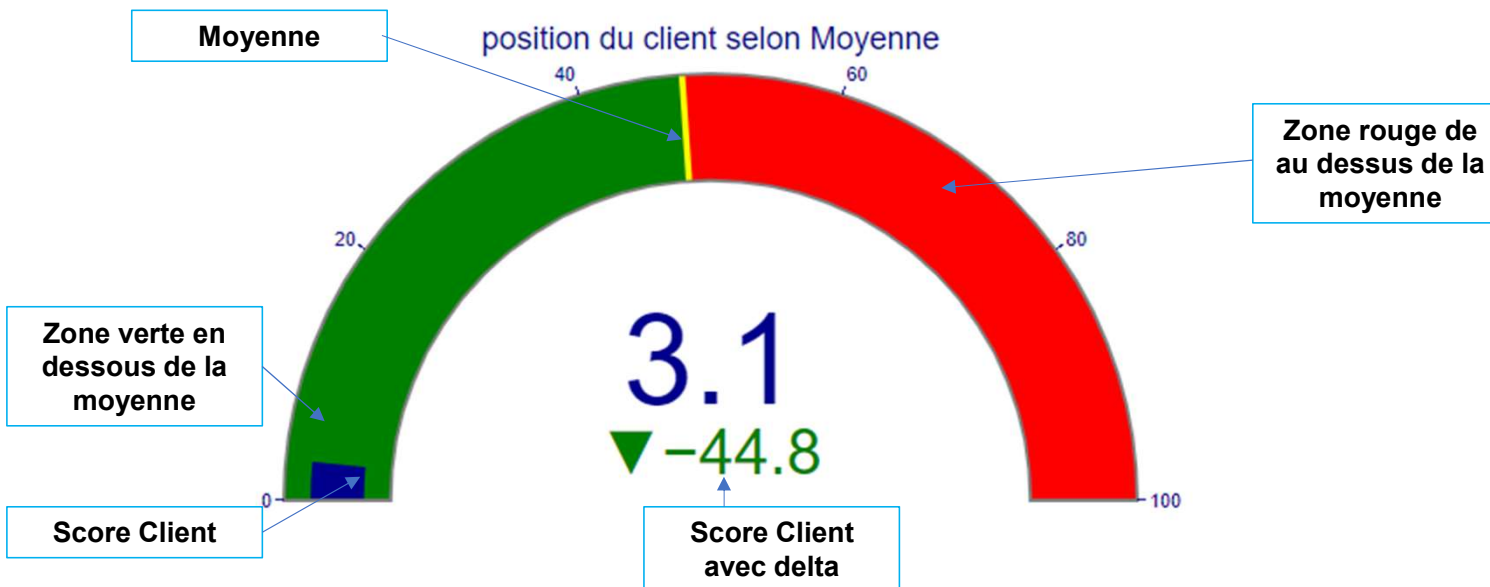


Jauge de score client avec Moyenne :

Moyenne : 47.86653560797283

Score Client : 3.067

☐ Permet de visualiser la position du client par rapport à la moyenne



2) Dashboard interactif

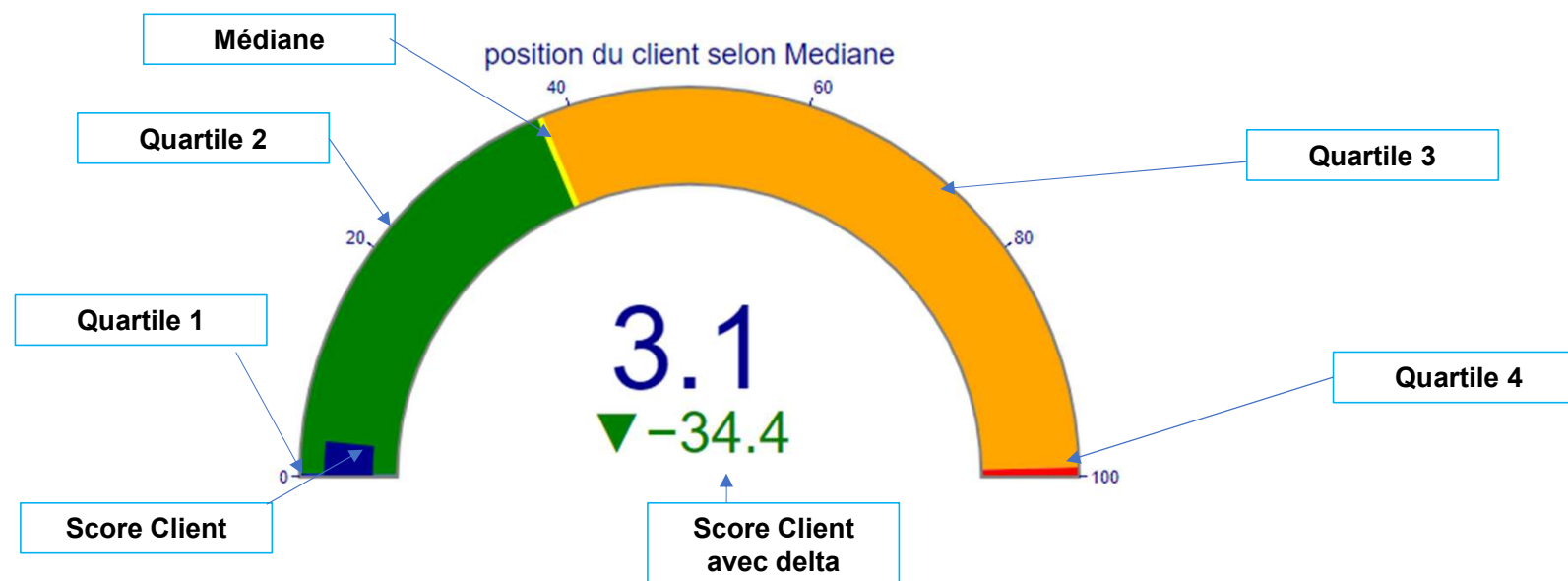


Jauge de score client avec Mediane :

Mediane : 37.452154210785025

Score Client : 3.067

☐ Permet de visualiser la position du client par rapport à la médiane

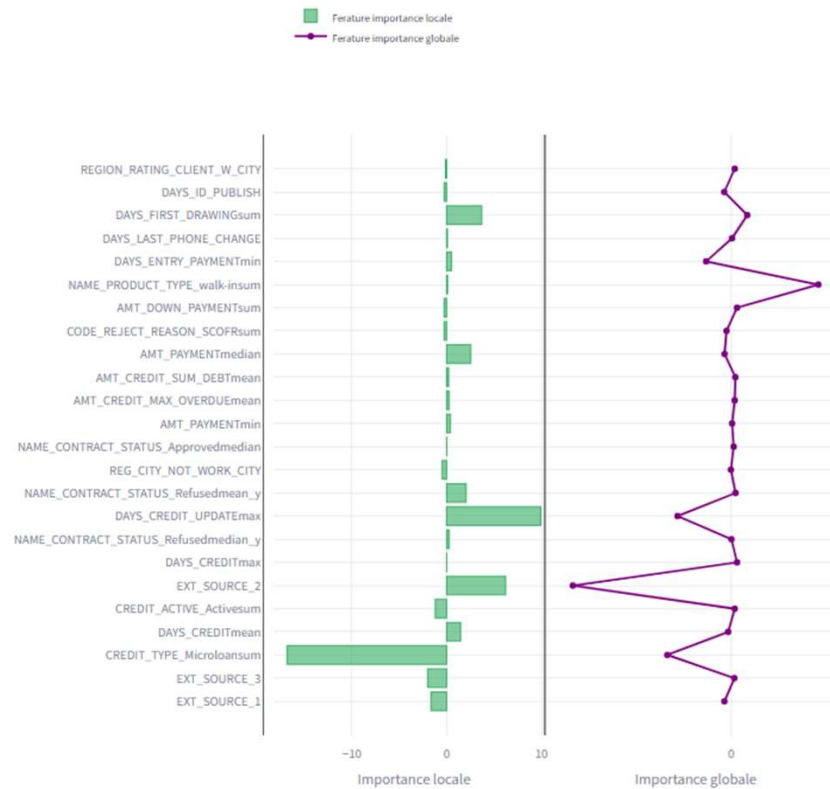


2) Dashboard interactif



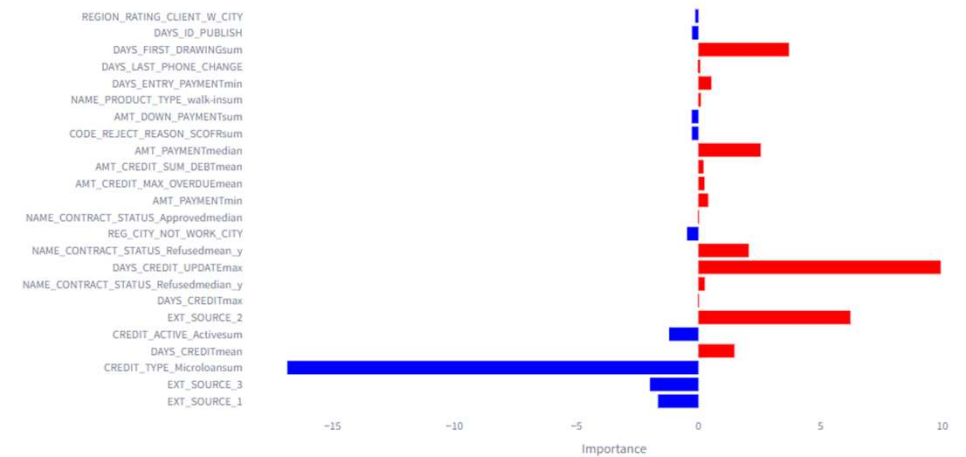
Comparaison feature importance globale et locale :

Comparaison feature importance locale et globale



Evaluation de l'importance des critères dans la notation du Client :

Importance des features dans la décision

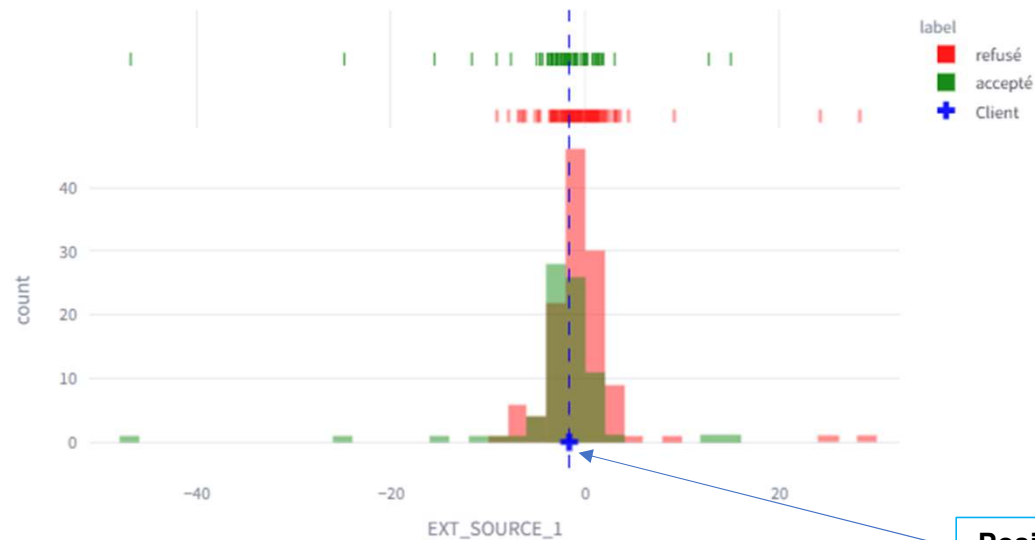


2) Dashboard interactif



Position du client vs autres clients sur critère sélectionné :

Importance feature EXT_SOURCE_1 clients



- ☐ Permet de visualiser la distribution d'une feature sélectionnée, les clients refusés et acceptés, et de positionner le client actuel

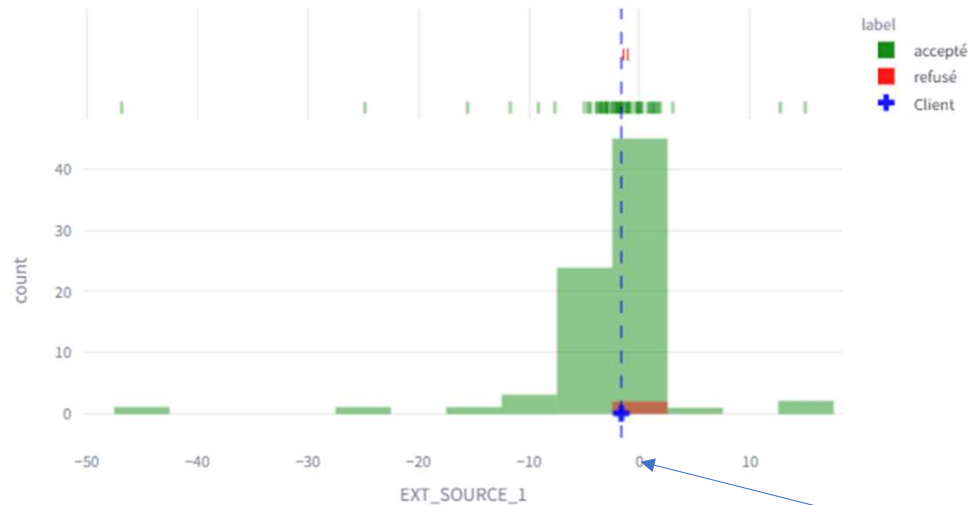
Position du client sélectionné vs autres clients

2) Dashboard interactif



Position du client vs clients similaires sur critère sélectionné :

Importance feature EXT_SOURCE_1 clients similaires



- ☐ Permet de visualiser la distribution d'une feature sélectionnée par rapport à des clients similaires (score +5 et -5), et de positionner le client

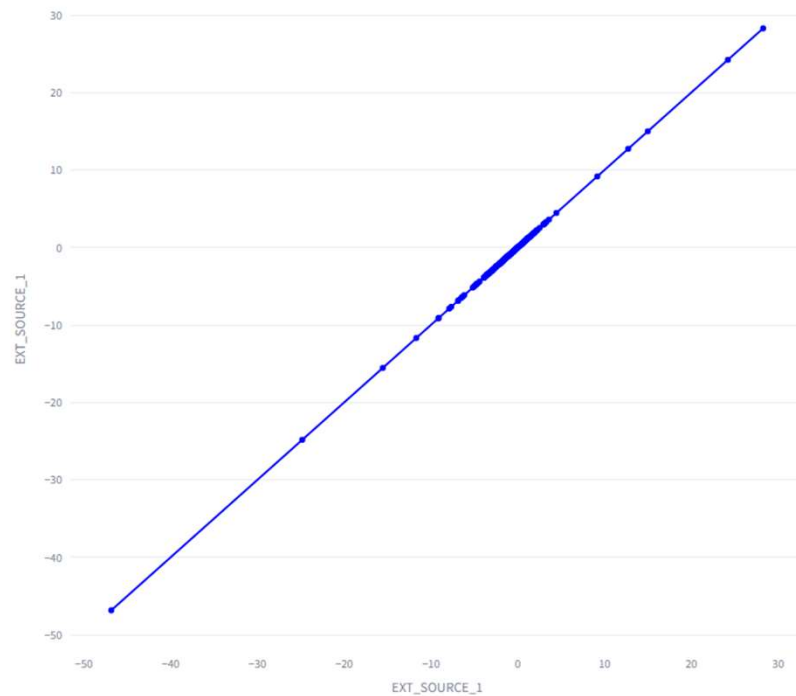
Position du client sélectionnée
vs autres clients similaires

2) Dashboard interactif



Analyse globale entre les features :

Analyse bivarié feature importance



☐ Permet d'analyser les features entre elles

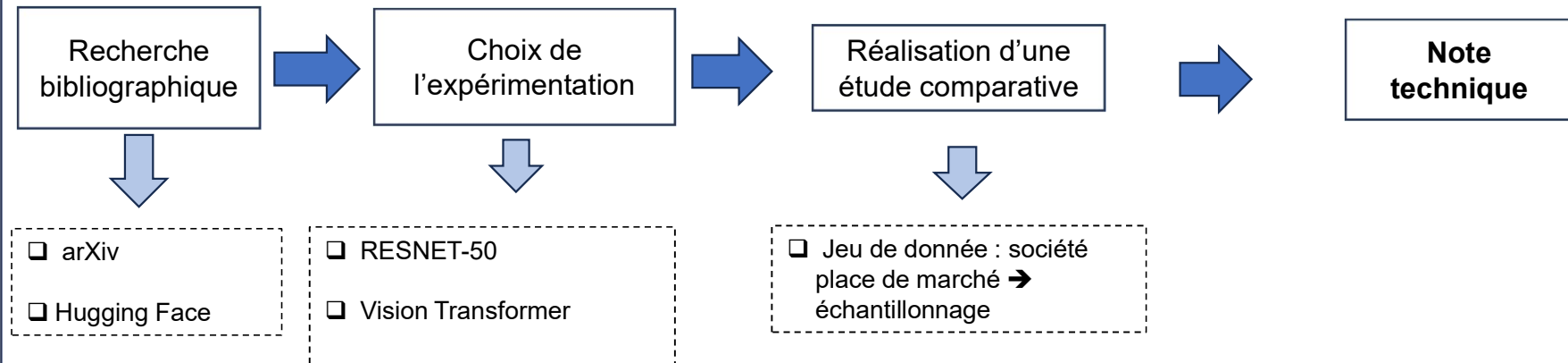
2) Dashboard interactif



☐ **Démonstration en ligne**

III) Veille technique

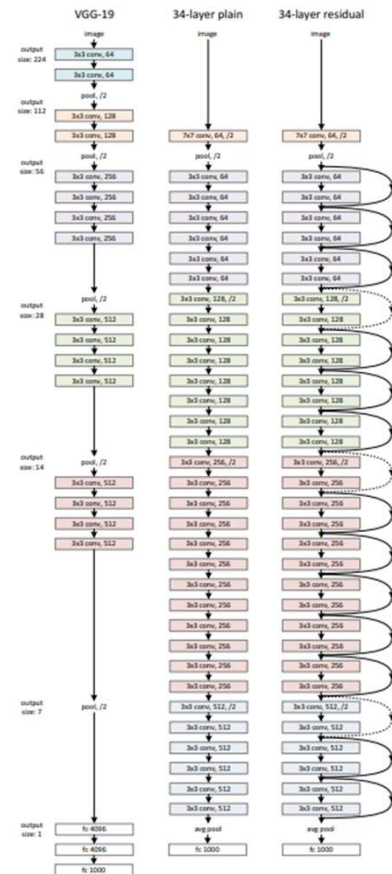
3) Veille technique



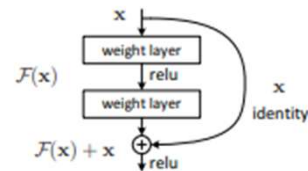
2

**Veille
Technique**

3) Veille technique



❑ **Le RESNET-50**
- un réseau convolutif profond avec des shortcuts



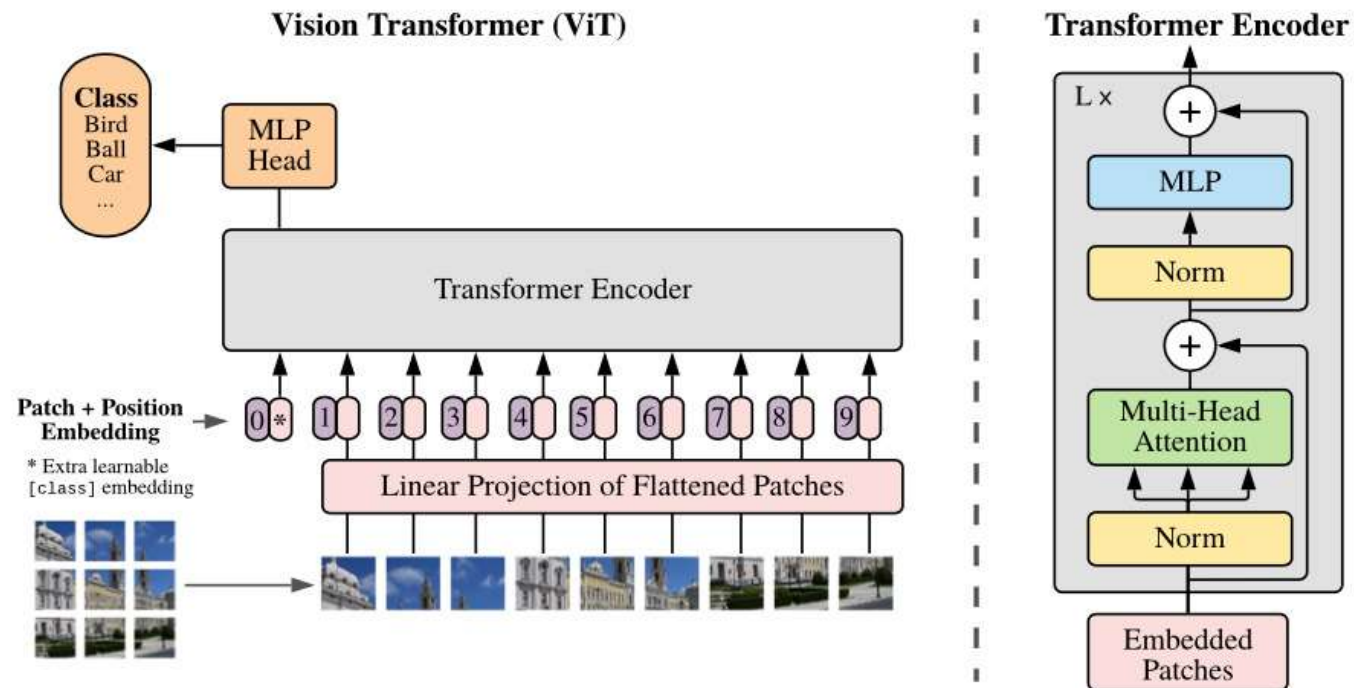
layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112 × 112			7 × 7, 64, stride 2		
				3 × 3 max pool, stride 2		
conv2,x	56 × 56	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3,x	28 × 28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4,x	14 × 14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5,x	7 × 7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1 × 1			average pool, 1000-d fc, softmax		
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

3) Veille technique



❑ Le Vision Transformer

- une approche avec un transformer encoder



3) Veille technique

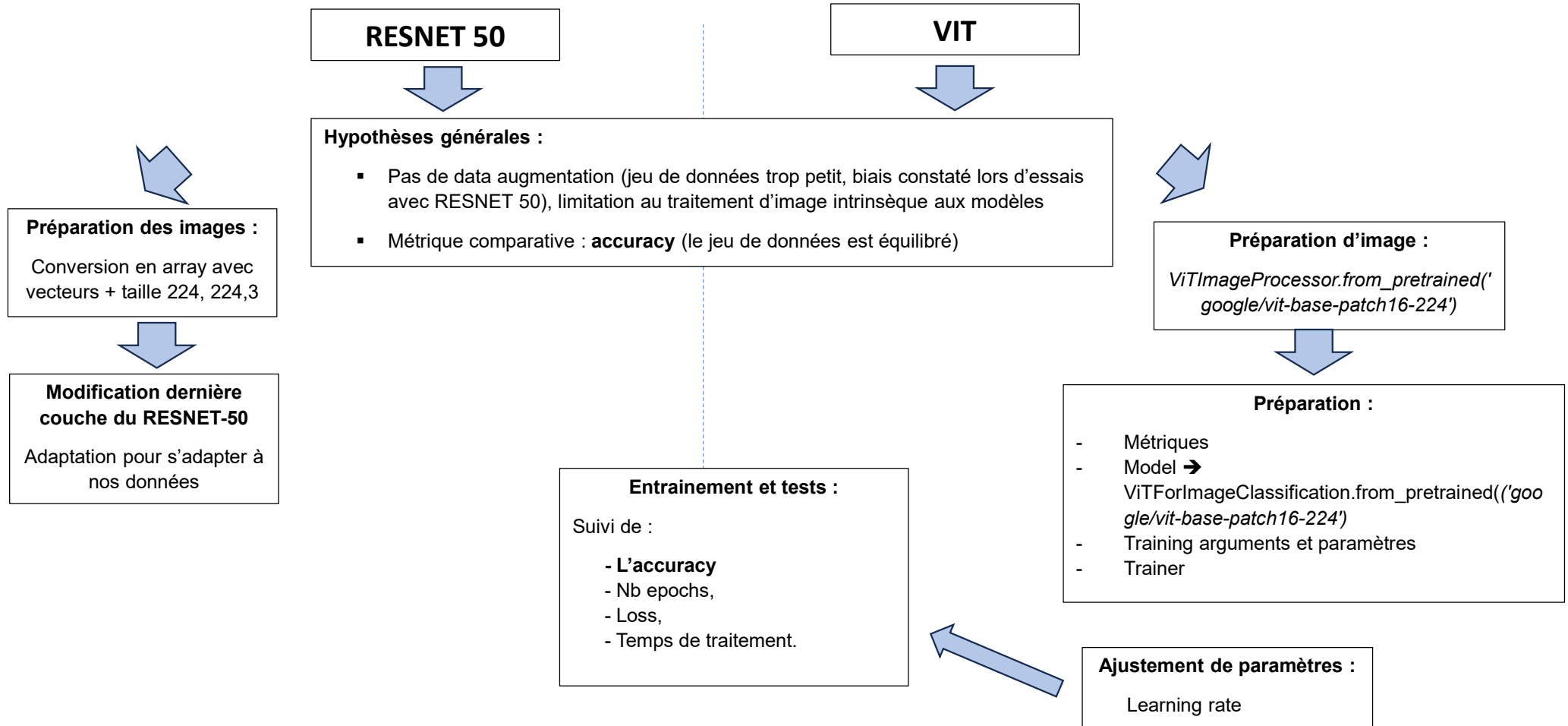


Résultats

	RESNET-50		VIT	
EPOCH	Accuraccy val	Loss val	Accuraccy val	Loss val
1	0.539	1.374	0.714	1.583
2	0.666	1.048	0.785	1.344
3	0.714	1.006	0.809	1.153
4	0.698	1.069	0.809	1.046
5	0.682	1.059	0.785	1.013

	RESNET-50	VIT
TEMPS TOTAL	00 : 01 : 29	00 : 12 : 32

3) Veille technique



3) Veille technique



Algorithmes très performants et leur utilisation dépendra de l'usage que l'on veut en faire

- ☐ Réactivité et économique → le RESNET-50
- ☐ Précision → VIT



☐ **Limites VIT :**

- Le temps de traitement est très long, et cela a forcément un impact financier.

☐ **Améliorations possibles VIT :**

- Elargir le jeu de données
- Affiner le Learning rate
- Réaliser le traitement en parallèle sur plusieurs machines

IV) Conclusion

6) Conclusion



Réaliser un dashboard interactif et le déployer sur le cloud, et réaliser une veille technique		
À partir du projet précédent élaborer un dashboard avec streamlit	Un dashboard interactif a été réalisé avec streamlit à partir de l'application déployée sur render.com	
Respecter le principe WCAG	<ul style="list-style-type: none"> - Graphiques interactifs, - Interface utilisateur activable avec la souris et avec le clavier, - Affichages multiples et sur de multiples supports... 	
Tester et déployer le dashboard avec strealmit io	Le dashboard a été testé en local et déployé sur le cloud	
Réalisez une veille technique sur le vision transformer	Une veille technique a été réalisée les sources → l'Arxiv	
Réalisez un POC	Un poc a été réalisé sur un projet précédent et sur un échantillon de jeu de donnée	
Réalisez une note technique	Une note technique a été réalisée	

Merci

- Armand FAUGERE
- armand-faugere@live.fr

