

Te@m Am@ng US

×

ESiEE[it]
L'école de l'expertise numérique

I.A

×

Prédire un accord de prêt bancaire

Loïc **DUMAY**
Zhihao **CHEN**
Emmanuel **ROSE**

Maxence **HENNEKEIN**
Louis **LEMOINE**



08/12/2022

×



1/22

Préambule

Le besoin :

- Nous avons des données qui classent si le client peut avoir un prêt ou non selon les caractéristiques qu'il contient

La solution:

- créer un modèle qui essaie de prédire si un client a droit à ce prêt ou non.



De quoi va t-on parler ?

- 01 LES DONNÉES**
- 02 LA RÉPONSE AU CLIENT**
- 03 PRÉSENTATION DES MODÈLES**
- 04 DÉMONSTRATION**



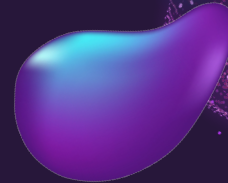
01 LES DONNÉES



08/12/2022



+



4/22

Les données de base

	age	job	marital	education	default	balance	housing	loan	contact	day	month	duration	campaign
0	58	management	married	tertiary	no	2143	yes	no	unknown	5	may	261	1
1	44	technician	single	secondary	no	29	yes	no	unknown	5	may	151	1
2	33	entrepreneur	married	secondary	no	2	yes	yes	unknown	5	may	76	1
3	47	blue-collar	married	unknown	no	1506	yes	no	unknown	5	may	92	1
4	33	unknown	single	unknown	no	1	no	no	unknown	5	may	198	1

Suppression des données

	age	job	marital	education	default	balance	housing	loan	contact	day	month	duration	campaign	pdays	previous	poutcome	Target
0	58	management	married	tertiary	no	2143	yes	no	unknown	5	may	261	1	-1	0	unknown	no
1	44	technician	single	secondary	no	29	yes	no	unknown	5	may	151	1	-1	0	unknown	no
2	33	entrepreneur	married	secondary	no	2	yes	yes	unknown	5	may	76	1	-1	0	unknown	no
3	47	blue-collar	married	unknown	no	1506	yes	no	unknown	5	may	92	1	-1	0	unknown	no
4	33	unknown	single	unknown	no	1	no	no	unknown	5	may	198	1	-1	0	unknown	no
...
45206	51	technician	married	tertiary	no	825	no	no	cellular	17	nov	977	3	-1	0	unknown	yes
45207	71	retired	divorced	primary	no	1729	no	no	cellular	17	nov	456	2	-1	0	unknown	yes
45208	72	retired	married	secondary	no	5715	no	no	cellular	17	nov	1127	5	184	3	success	yes
45209	57	blue-collar	married	secondary	no	668	no	no	telephone	17	nov	508	4	-1	0	unknown	no
45210	37	entrepreneur	married	secondary	no	2971	no	no	cellular	17	nov	361	2	188	11	other	no

Teđm Amđng US

Importance théorique

Importance en %	
age	5.795034
marital	2.615462
education	3.476249
default	0.223050
balance	14.564262
housing	4.518890
loan	1.312084
month	10.610606
duration	40.010065
campaign	5.309925
pdays	4.638283
previous	2.969819
poutcome	3.956271

L'importance de ces données est-elle pertinente ?

Cible : Compte à terme (Target)

Le client à accepter ?

Conversion des données

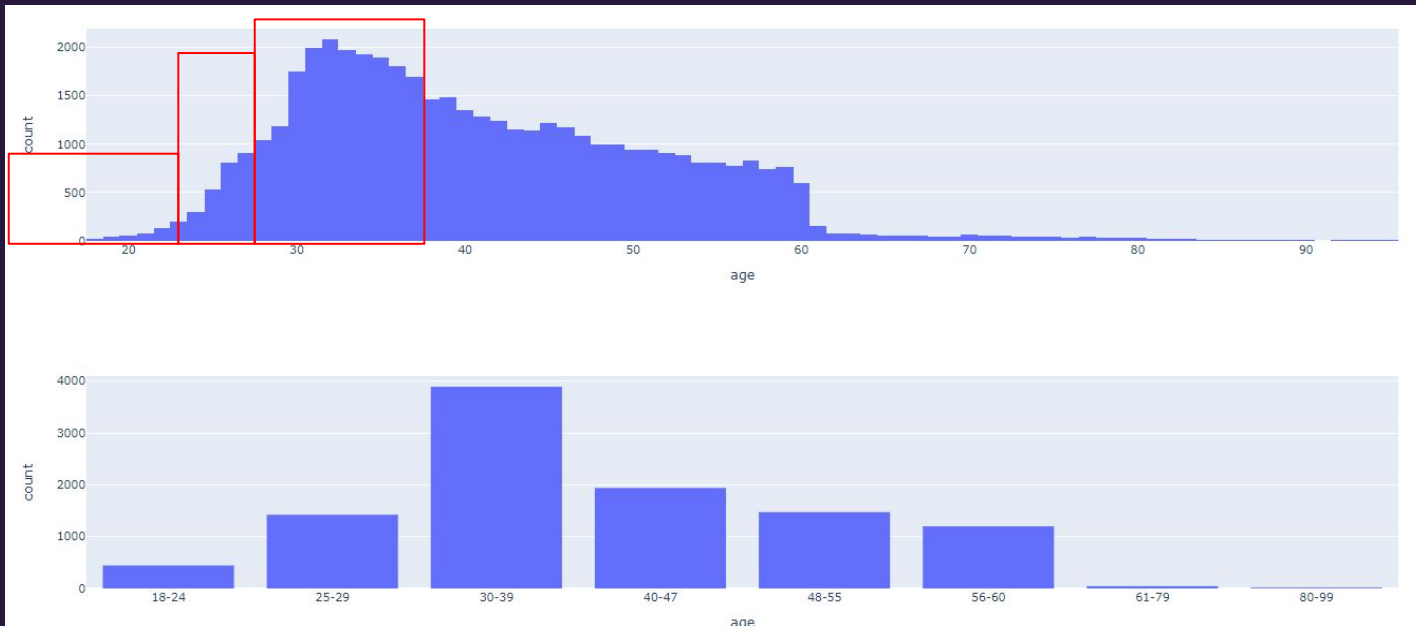
	age	marital	education	default	balance	housing	loan	month	duration	campaign	pdays	previous	poutcome	Target
41487	1	0	0	0	108	0	0	9	169	1	-1	0	0	1
40887	1	0	1	0	608	0	0	8	267	1	-1	0	0	1
43637	1	0	0	0	348	0	0	5	443	4	-1	0	0	1
42274	1	0	1	0	608	0	0	11	210	1	93	1	1	1
40744	1	0	0	0	108	0	0	8	167	1	-1	0	0	1
...
42727	8	1	0	0	775	0	0	1	164	4	96	3	1	1
44262	8	1	0	0	775	0	0	7	860	2	177	7	1	1
44420	8	1	0	0	775	0	0	8	476	2	13	9	1	1
33699	8	2	1	0	2282	0	0	4	207	17	-1	0	0	1
41663	8	1	2	0	0	0	0	10	215	1	-1	0	0	0



Teđm Amđng US

Simplification des données

Colonne age :

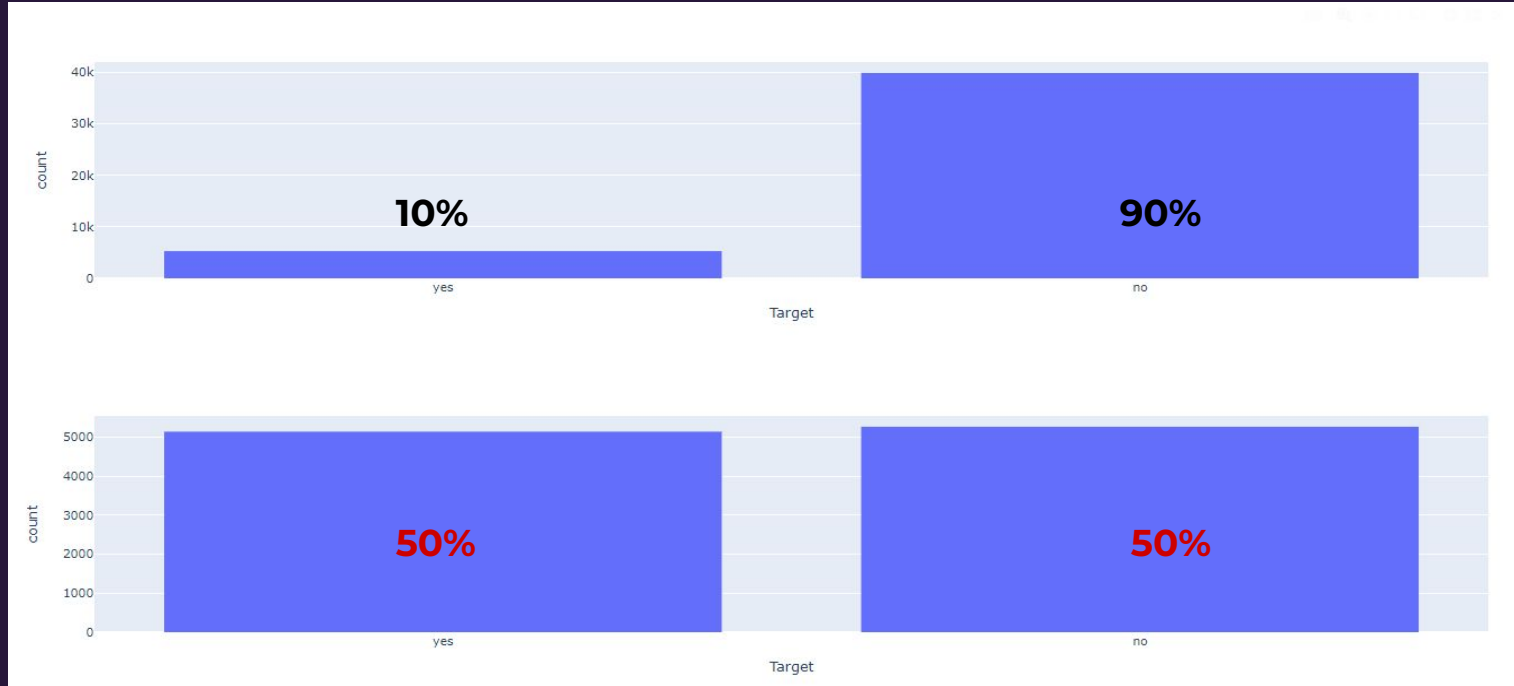


Teđm Amđng US

Colonne durée de communication avec le client en seconde :



Equilibrage des données (simplification) :



02

LA RÉPONSE AUX SPÉCIFICATIONS CLIENT



08/12/2022



+

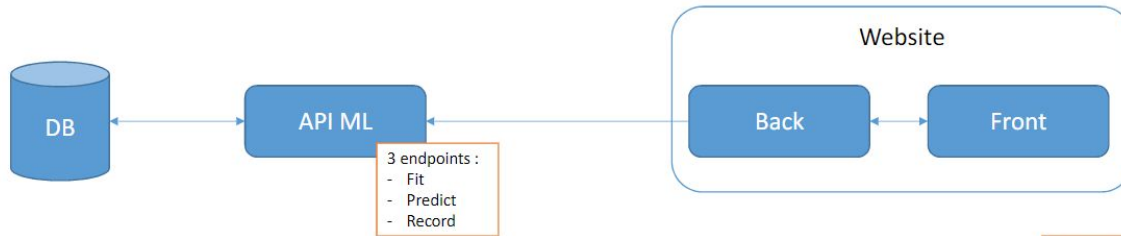


12/22

02

Applicatif

Architecture proposée :



Architecture :
Microservice

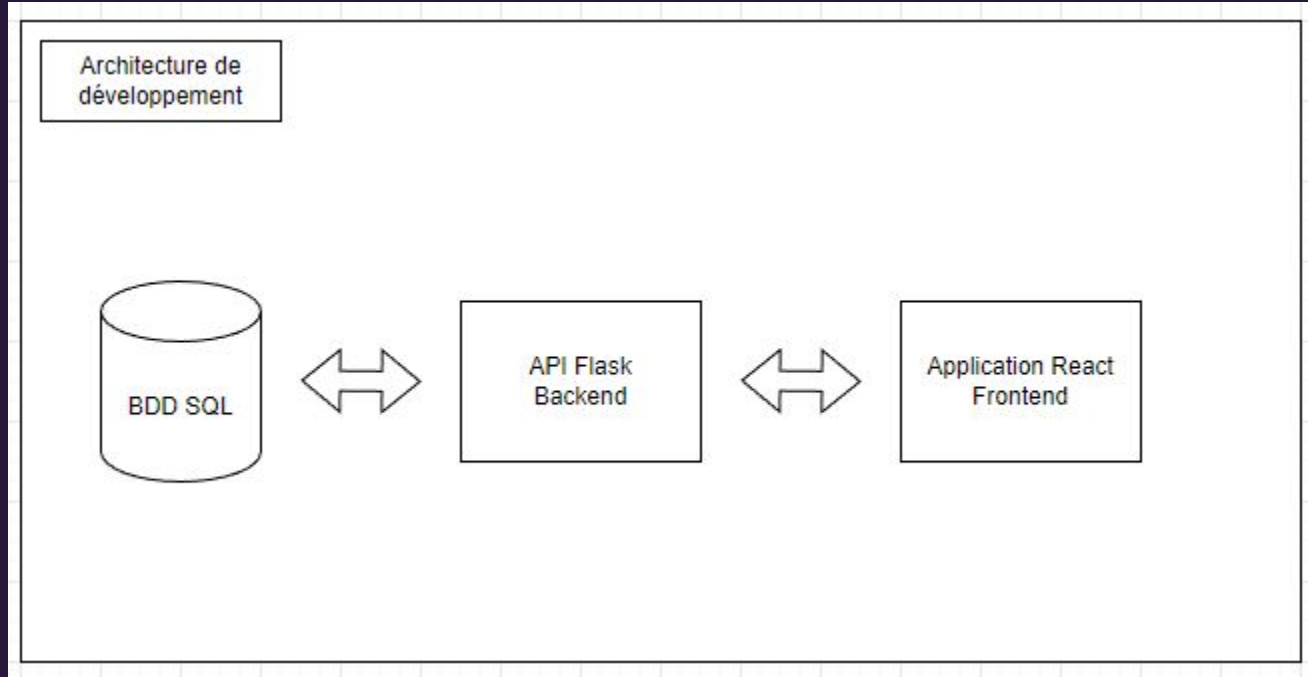
Language :
Python

Stack :

- SQLite (BDD)
- API : fastapi
- Web : Dash

02

Applicatif



03

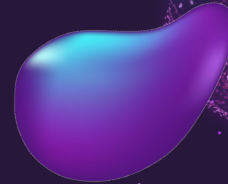
Présentation des modèles



08/12/2022



+



15/22

03

Présentation des modèles

RandomForestClassifier

```
rf = RandomForestClassifier(n_estimators = 100, random_state = 23, max_features = 2)
rf.fit(X=x_train, y=y_train)
print_score(rf)
```

Score train : 100.0 %

Score test : 84.59692898272553 %

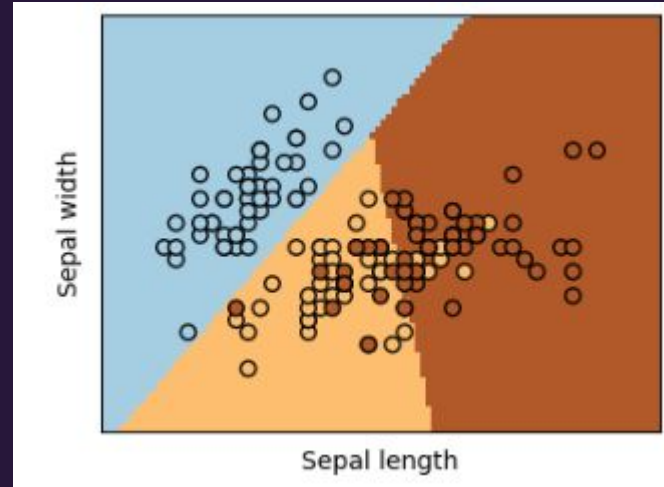


03

Présentation des modèles

Logistic Regression 3-class Classifier

```
Score train : 49.00407967362611 %  
Score test : 51.77543186180422 %
```



03

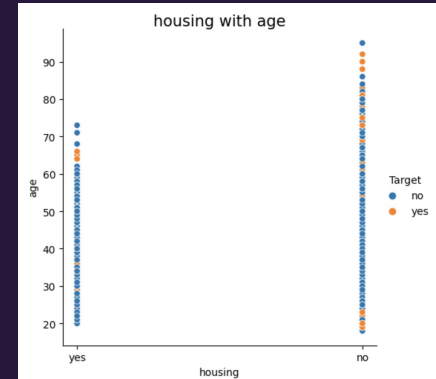
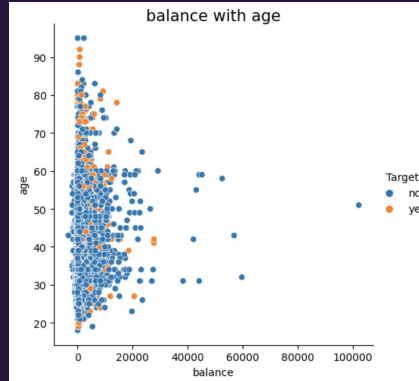
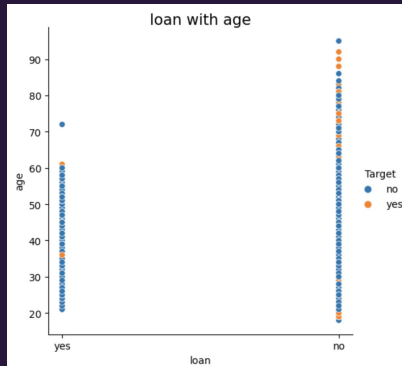
Présentation des modèles

GradientBoostingClassifier

```
In [50]: clf = GradientBoostingClassifier(n_estimators = 100, random_state = 23, max_features = 2)
         clf.fit(X=x_train, y=y_train)
         print_score(clf)
```

Score train : 83.95728341732661 %

Score test : 83.92514395393475 %



04

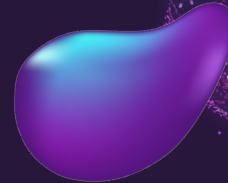
DÉMONSTRATION



08/12/2022



+



19/22

04

Démonstration



04 Démonstration



Le projet est actuellement open-source :

Grâce à l'open-source, le projet est disponible
sur GitHub.com à l'adresse suivant :

https://github.com/ldumay/esiee_2022_ia_tp_banque

*Étant soutenu, le dépôt du projet peut évoluer.
Cependant, les versions sont enregistrées
sous la forme de **release**.*



Te@m Am@ng US

×

ESiEE[it]
L'école de l'expertise numérique

Merci !

×



Loic **DUMAY**
Zhihao **CHEN**

Maxence **HENNEKEIN**
Louis **LEMOINE**
Emmanuel **ROSE**



08/12/2022

×



22/22