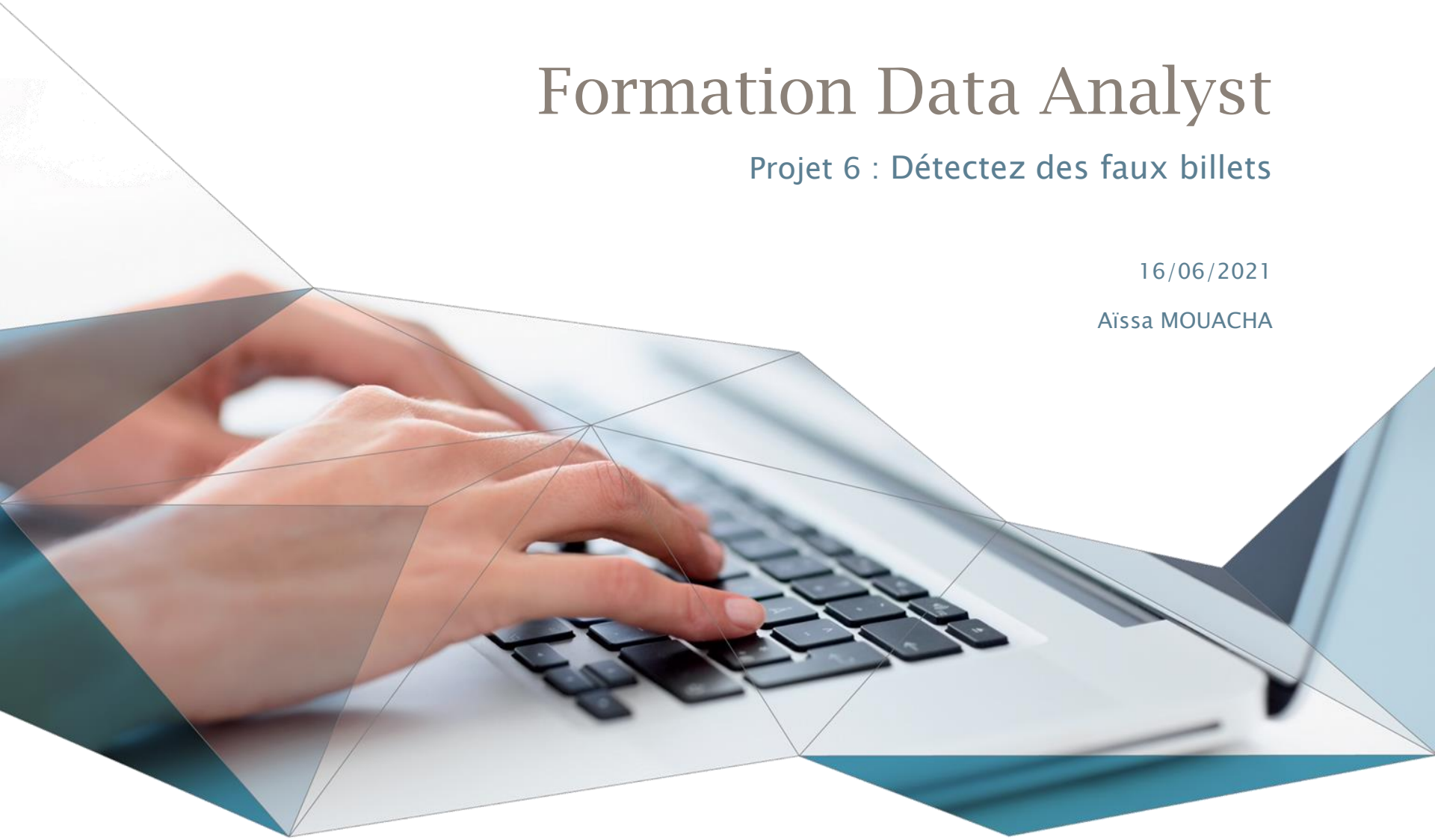
Formation Data Analyst

Projet 6 : Détectez des faux billets

16/06/2021

Aïssa MOUACHA



SOMMAIRE

- Introduction
- Statistique descriptive
- ACP
- Algorithme de classification (K-Means)
- Régression logistique
- Conclusion

Votre société de consulting informatique souhaite développer un algorithme de détection de faux billets.

Stratégie : Modélisez les données à l'aide d'une régression logistique.

Objectif : Créer un programme capable d'effectuer une prédiction sur un billet.

Source : Site OC

Statistique descriptive

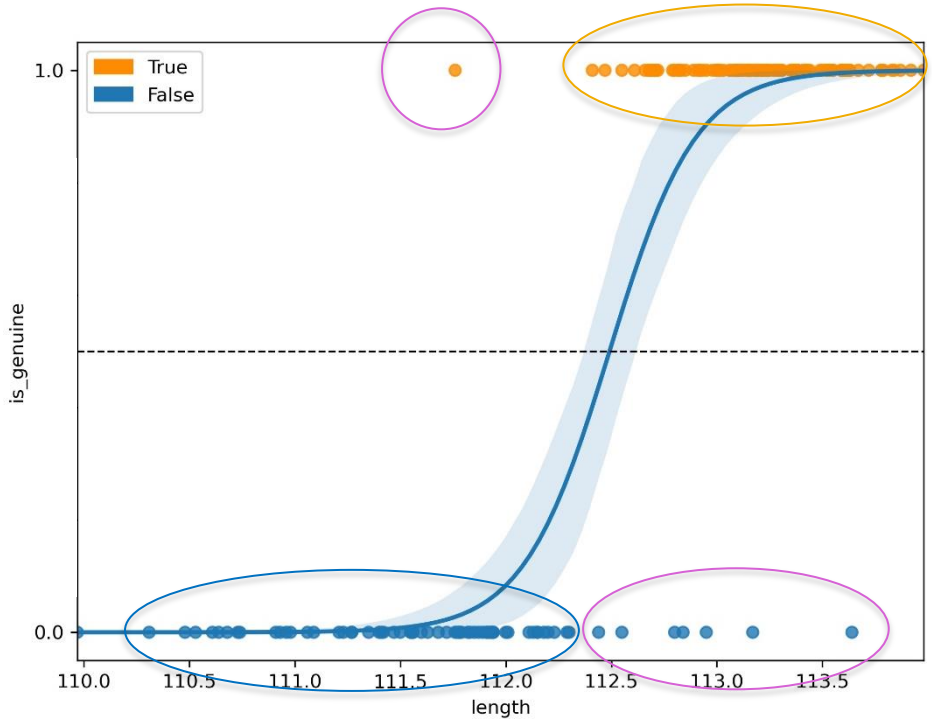
Décrire, résumer, représenter la donnée

Analyses univariées/bivariées

- Téléchargement
- Homogénéité des données

	is_genuine	diagonal	height_left	height_right	margin_low	margin_up	length
0	True	171.81	104.86	104.95	4.52	2.89	112.83
1	True	171.67	103.74	103.70	4.01	2.87	113.29
2	True	171.83	103.76	103.76	4.40	2.88	113.84
3	True	171.80	103.78	103.65	3.73	3.12	113.63
4	True	172.05	103.70	103.75	5.04	2.27	113.55

- A priori, plus un billet présente une longueur supérieure, plus il tend à être authentique.



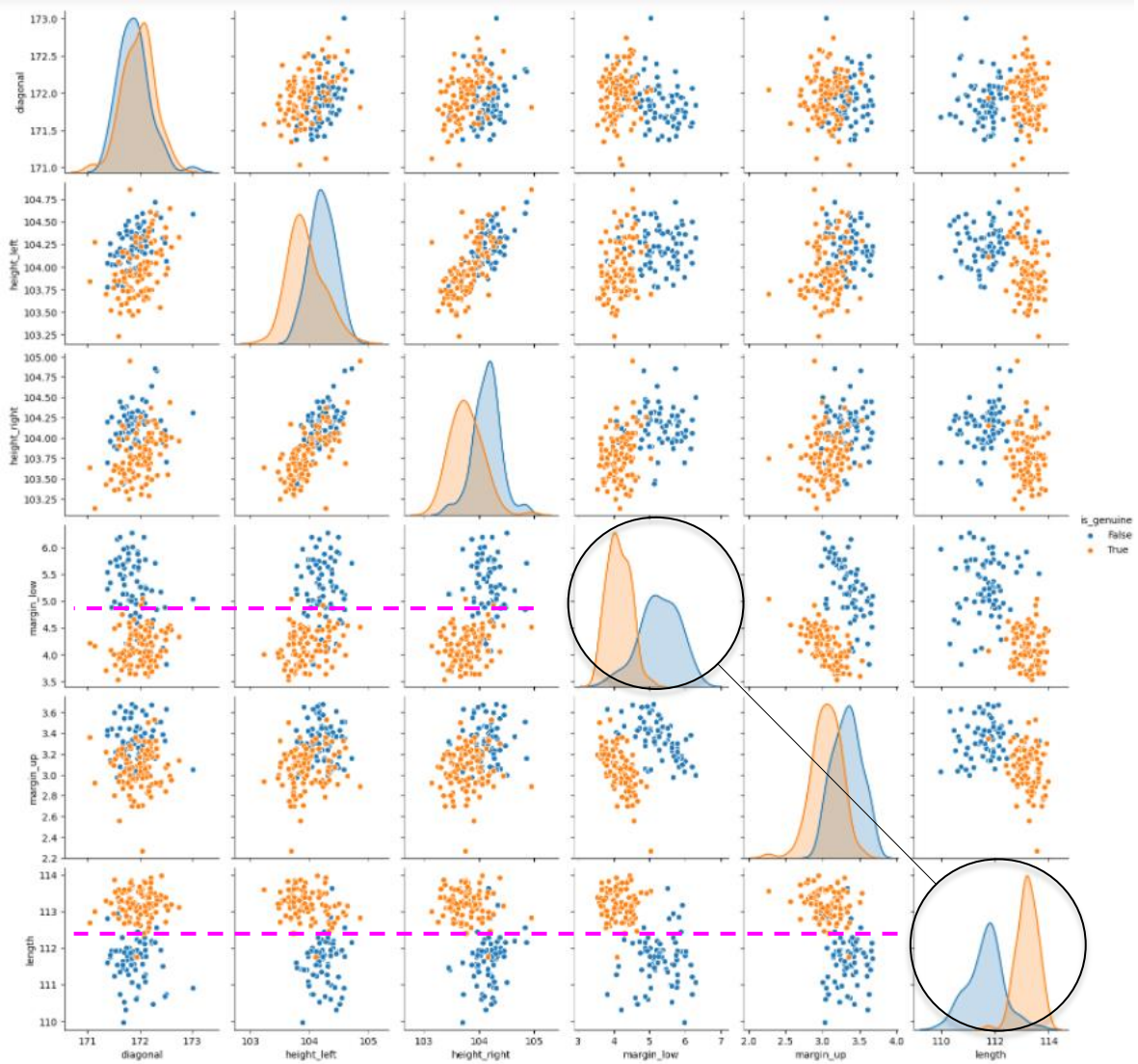
Analyses univariées/bivariées

- De manière générale :

2 voire 3 paramètres ont un impact réel sur l'authenticité d'un billet (distinction plus prononcée des populations).

A contrario, on peut deviner que *diagonal* aura très peu d'influence sur le fait qu'un billet soit vrai ou faux.

TBC avec la matrice de corrélation

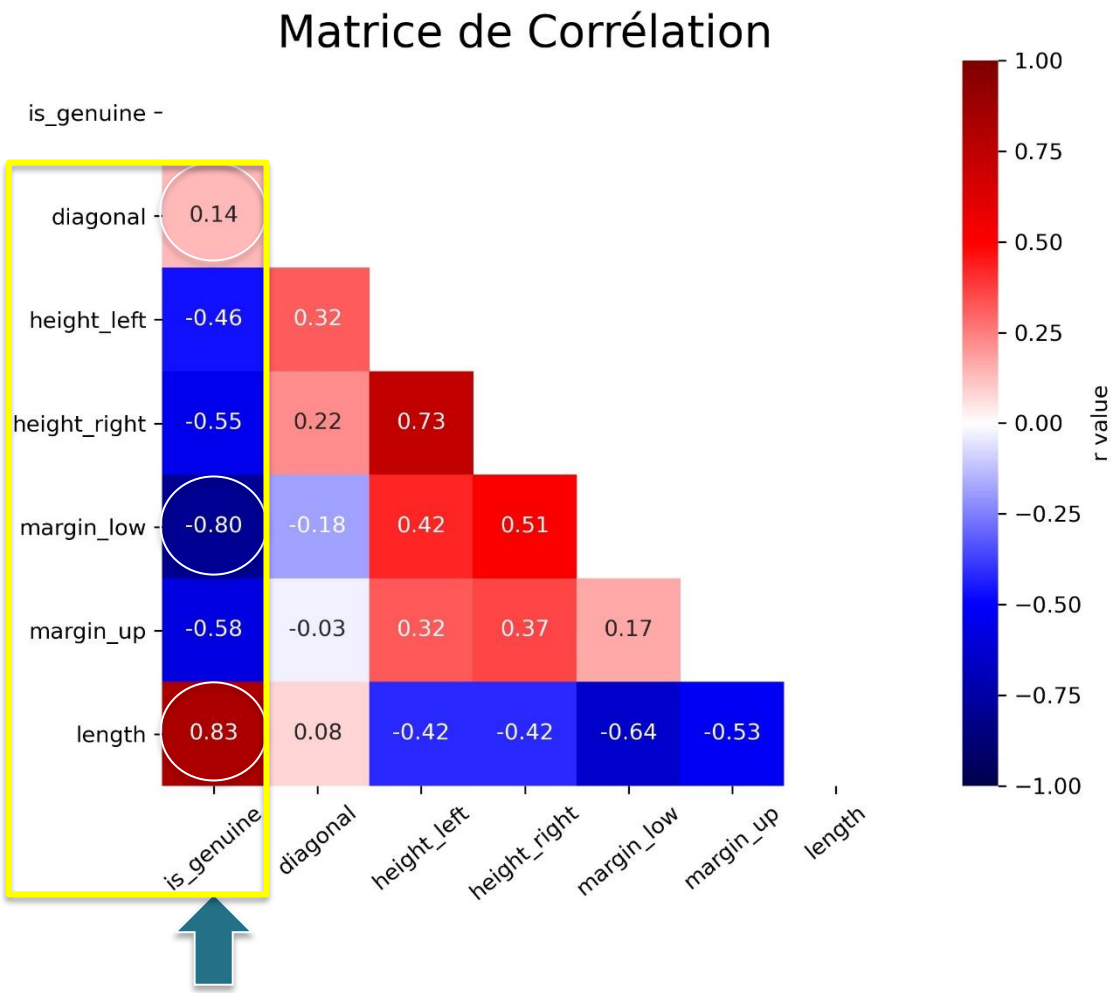


Analyses univariées/bivariées

- Par l'étude des covariances :

2 variables (*length & margin_low*) présentent une forte corrélation (respectivement positive et négative) avec la variable illustrative.

On retrouve la faible corrélation entre *diagonal* et la variable *is_genuine*.



Analyse en Composantes Principales

Transformer variables liées entre elles en variables synthétiques, réduire leur nombre, rendre l'information moins redondante

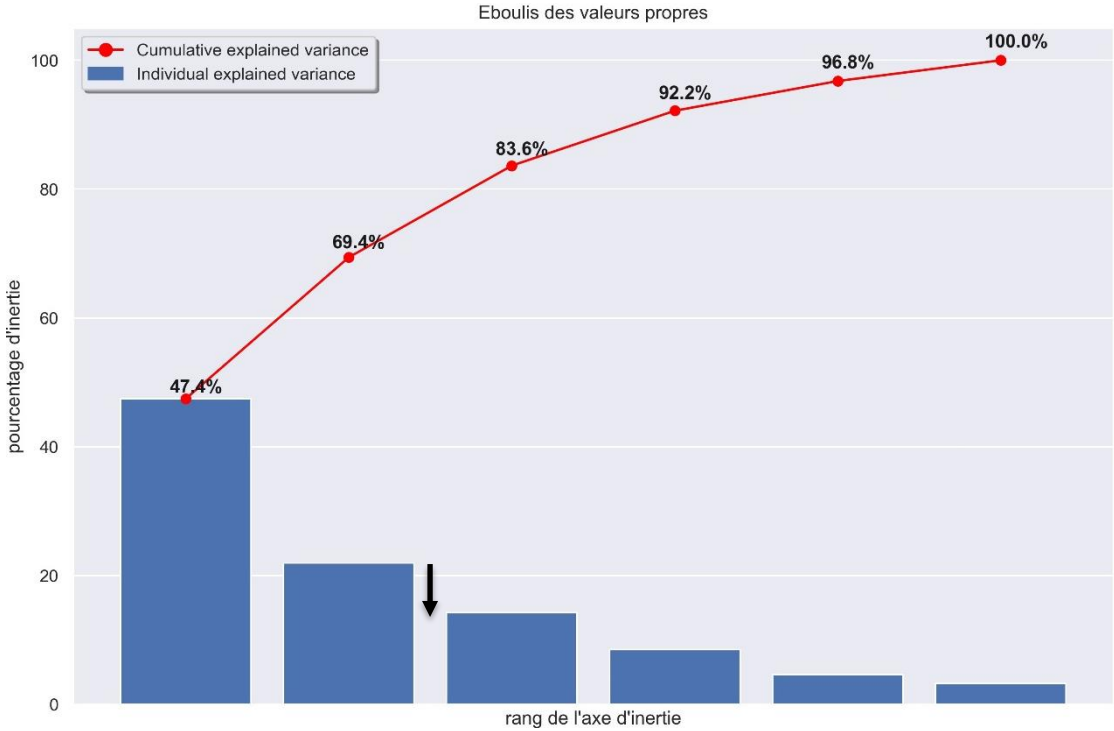
Eboulis des valeurs propres

- Par l'étude des inerties :

Les composantes F1 & F2 cumulent à elles-seules près de 70% de l'inertie totale.

(F3 suit avec 14%)

L'inflexion observée est nettement moins marquée après F2
➔ à priori 1 plan factoriel est suffisant



	Composantes	Valeur propre	% variance expliquée	% cum. var. expliquée
0	Comp1	2.863721	47.4	47.4
1	Comp2	1.325222	22.0	69.4
2	Comp3	0.859125	14.2	83.6
3	Comp4	0.514605	8.5	92.2
4	Comp5	0.278407	4.6	96.8
5	Comp6	0.194424	3.2	100.0

Critère de Kaiser : K=16.7% (100/p, inertie en dessous de laquelle les axes ne sont pas importants)

Cercle des corrélations

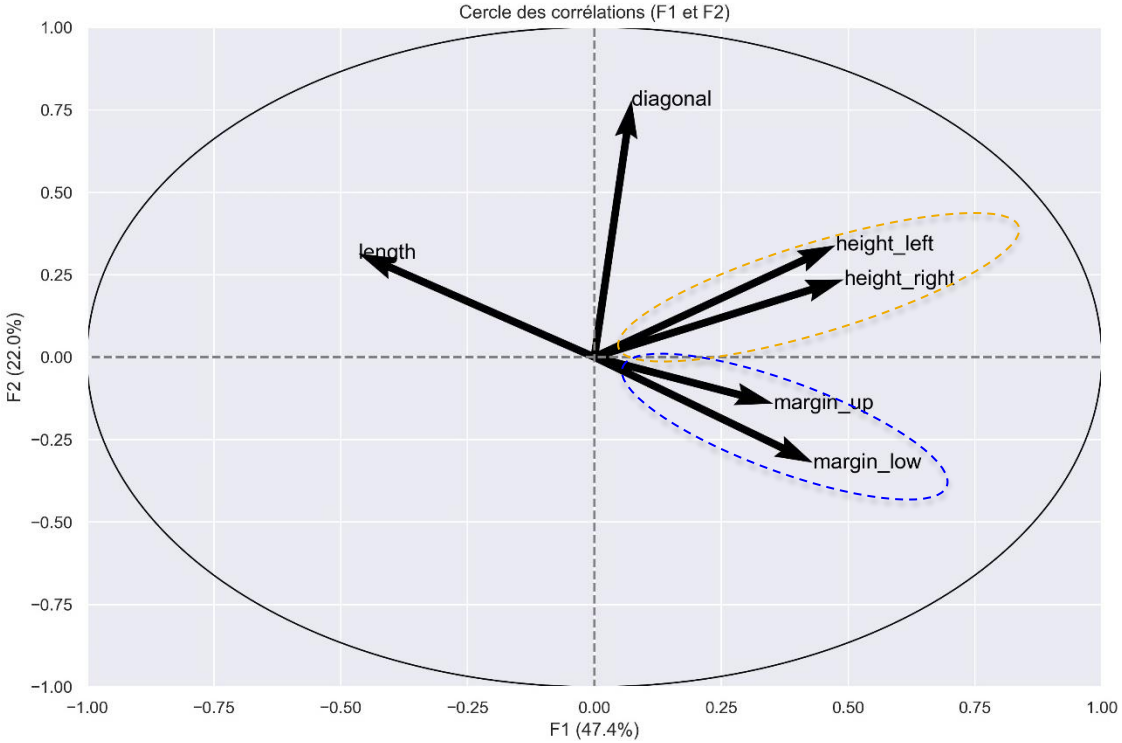
- Par projection des variables Q:

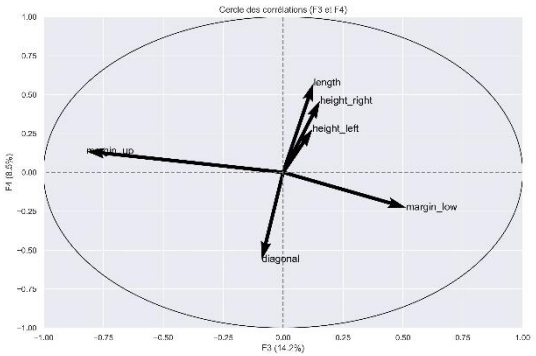
- *length* anticorrélée à F1, colinéarité avec *margin_low* ➔ simplifiable

- *diagonal* fortement corrélée à F2

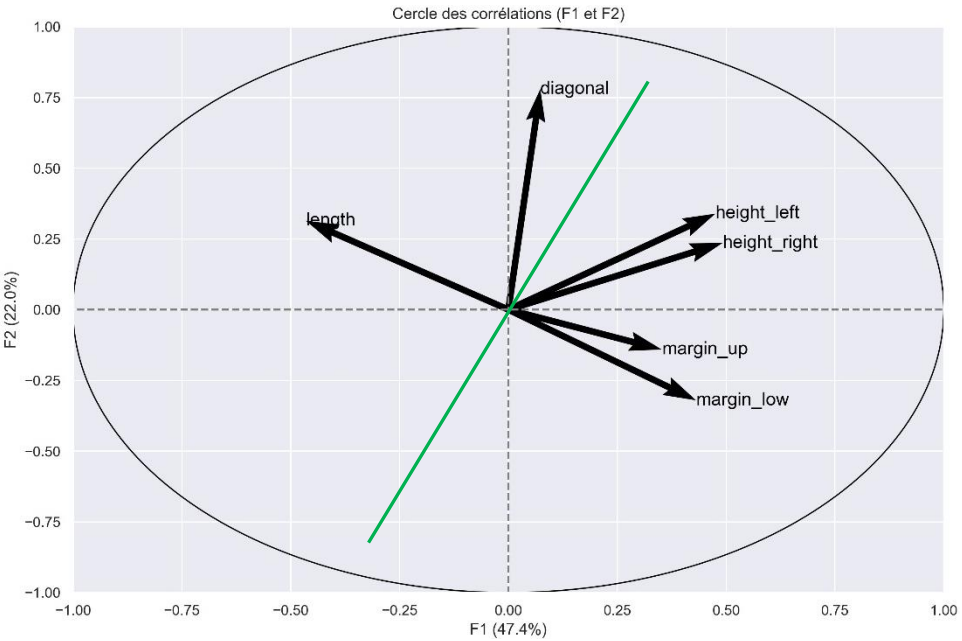
- corrélation *height_right & _left* à priori synthétisable (⚠ qualité projection)

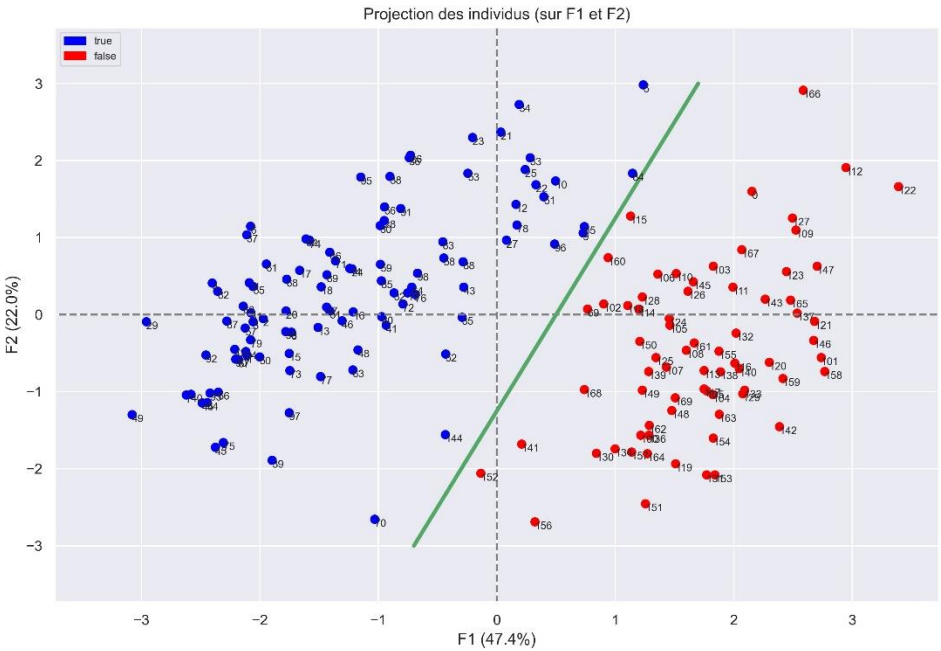
- pas le cas entre *margin_up & _low (forte contribution négative de margin_up à F3)*



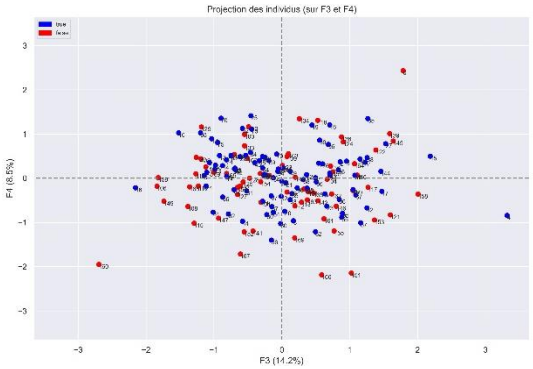


Projection des individus

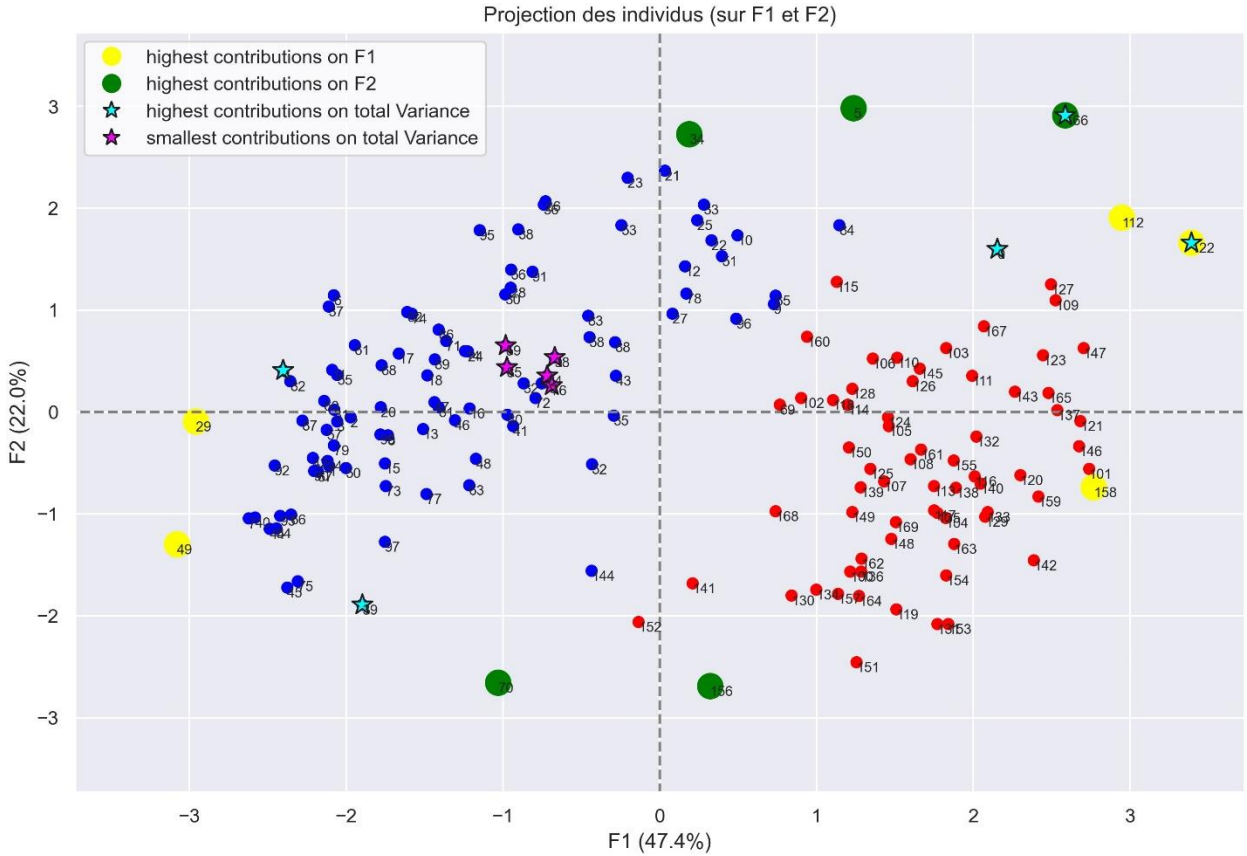




- Connaissant les valeurs de notre variable illustrative (True/False), nous avons 2 ensembles
- A priori, caractérisé par les variables *length* et/ou *margin_low* ➔ *TBC*

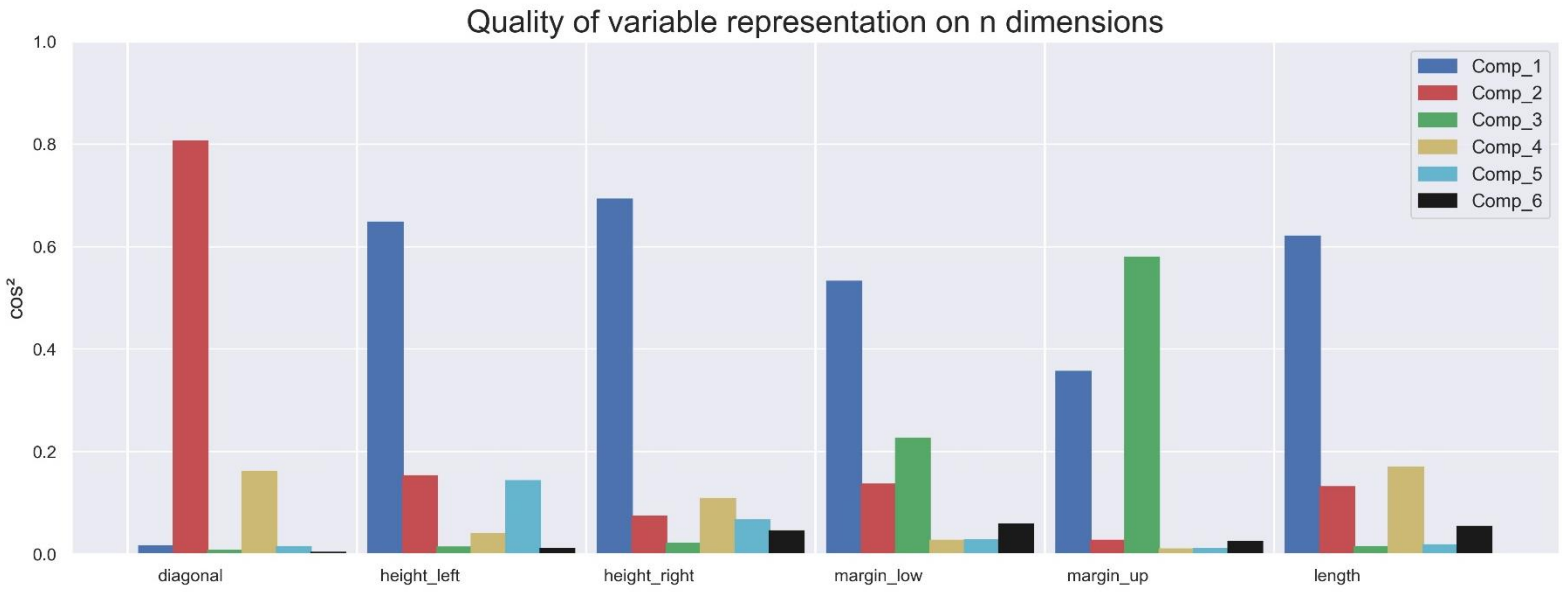


Qualité de représentation et contribution des individus



- Identification des individus aux fortes contributions aux axes
- Contribution des individus à l'inertie totale

Qualité de représentation et contribution des variables

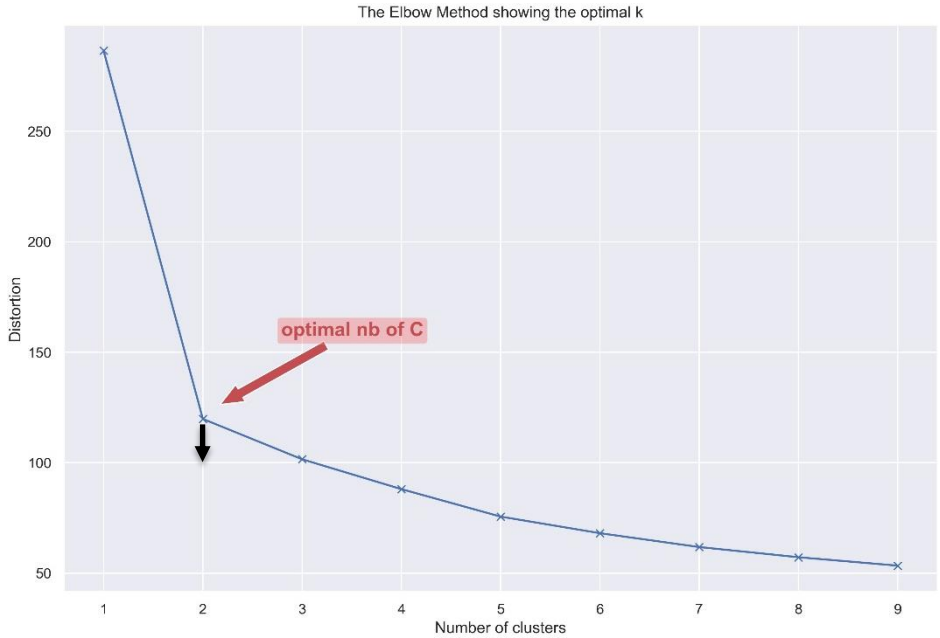


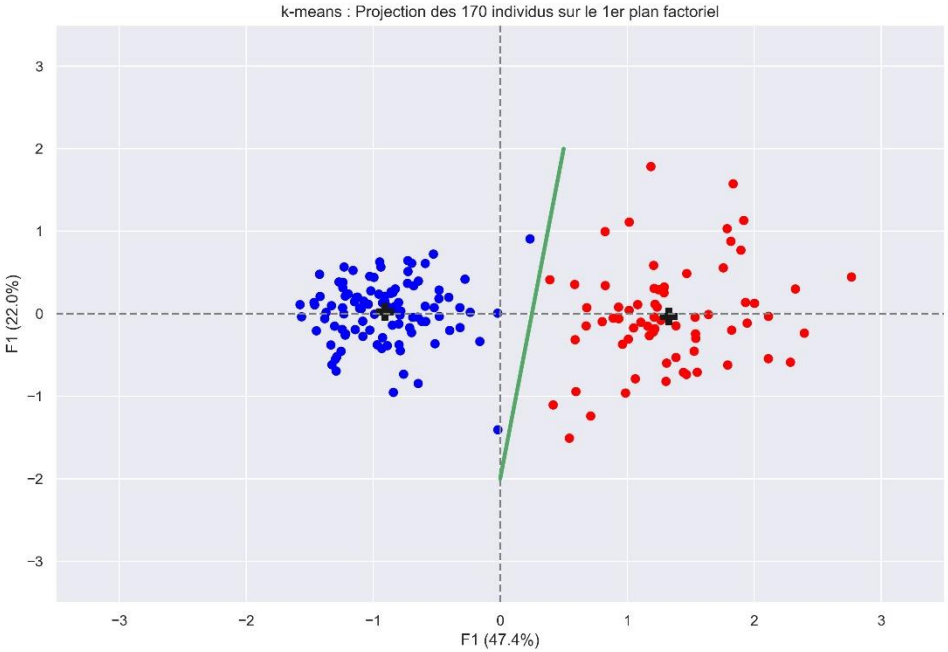
- Plus de la moitié de la qualité représentative des variables visible sur F1 & F2 (F3 importe aussi dans une moindre mesure pour deux variables)
- Confirmation du 1er plan factoriel retenu

Classification binaire

Répartir les membres d'un ensemble dans deux groupes disjoints

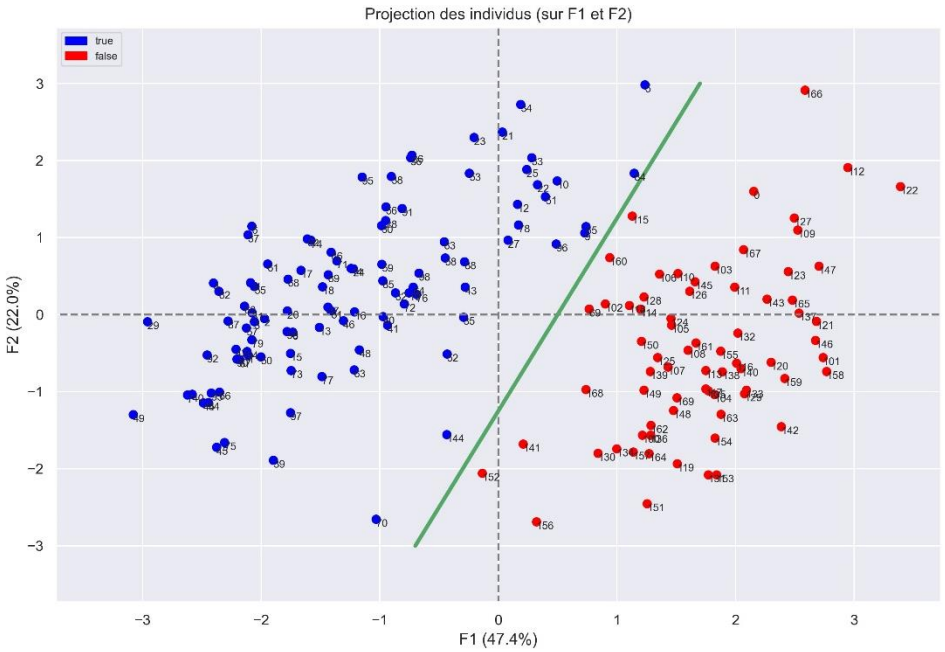
Algorithme K-Means

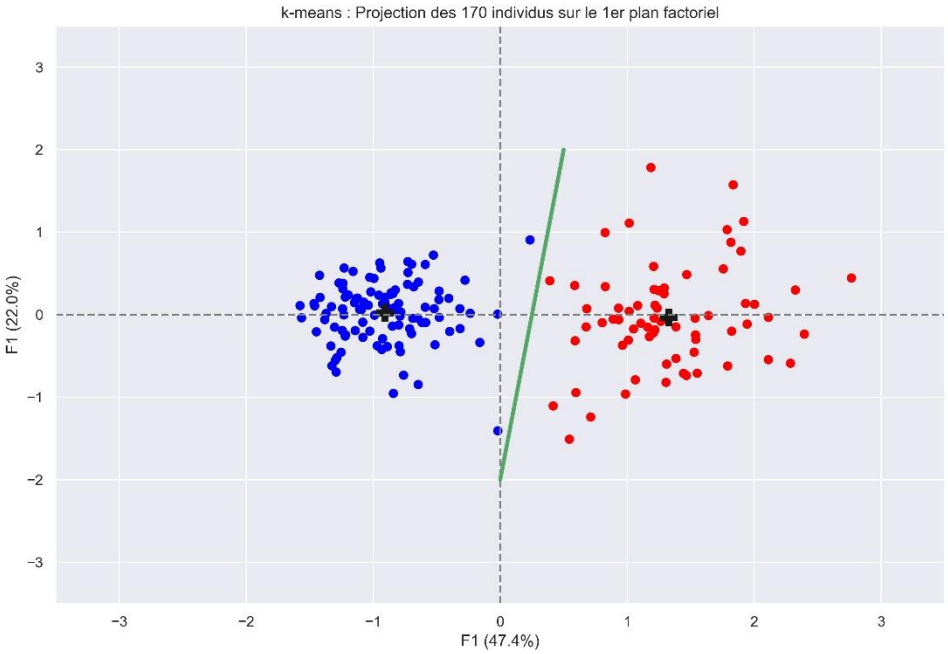




- Application Elbow Method ➔ Nb de clusters = 2
- Partition obtenue visualisée dans le premier plan factoriel de l'ACP
- 2 clusters distincts

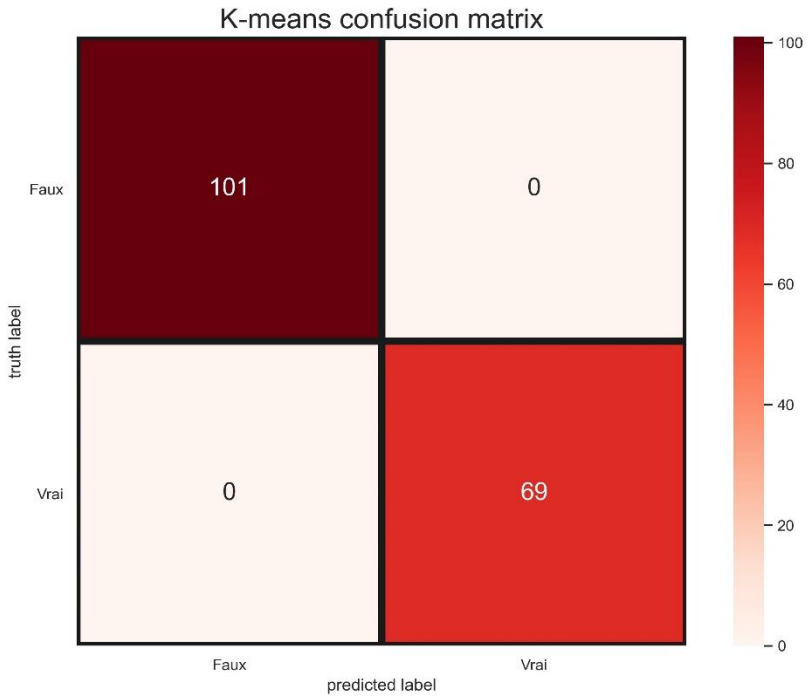
Algorithme K-Means : comparaison à la projection sur 1er plan





- Clustering donne 2 groupes distincts semblables à ceux obtenus lors de l'ACP
- Qu'en est-il de la précision de classification ...?

Algorithme K-Means : matrice de confusion



- L'algorithme réussit à parfaitement classer les individus (jeu de données parfait)

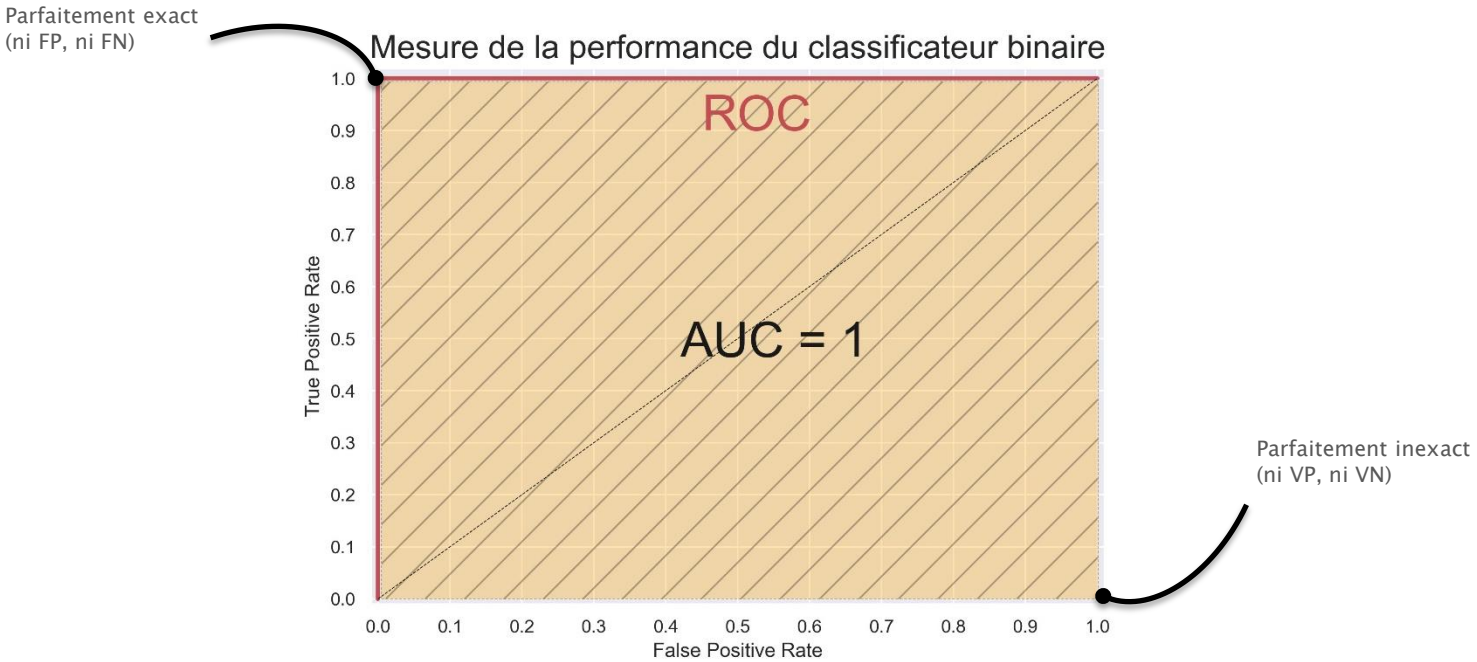
ARI (Adjusted Rand Index)

```
# A score close to 0.0 indicates random assignments
# A score close to 1 indicates perfectly labeled clusters
from sklearn.metrics import adjusted_rand_score
ARI = adjusted_rand_score(km.labels_, km.predict(X)) # adjusted_rand_score(labels_true, labels_pred)
print('\x1b[6;31;40m','Adjusted Rand Index :',np.round(ARI,2),'\x1b[0m')
```

```
Adjusted Rand Index : 1.0
```

- ARI = 1

Algorithme K-Means : AUC (Area Under The Curve)
ROC (Receiver Operating Characteristics)



- Le classificateur n'a aucun FP ni aucun FN, il est parfaitement exact, ne se trompant jamais

Régression logistique

Montrer une relation de dépendance entre une variable à expliquer et une série de variables explicatives
Modéliser cette association par un modèle mathématique

Régression logistique

- Lib Statmodels
- Split des données : jeu d'entraînement + jeu de test (20%)
- Ajout manuel de la constante (ou Intercept)
- Choix des variables à retenir :
  - cas divergence (nb maxi itération exceeded/singular matrix)

```
# x_trainbis=x_train[['const','diagonal','height_left','height_right','margin_low','margin_up','length']] => Max nb_ite exceeded
# x_trainbis=x_train[['const',           'height_left','height_right','margin_low','margin_up','length']] => Max nb_ite exceeded
# x_trainbis=x_train[['const',                         'height_right','margin_low','margin_up','length']] => Max nb_ite exceeded
```

  - cas convergence modèle en 9 itérations

Jeu 1

```
                           Logit Regression Results
==============================================================================
Dep. Variable:             is_genuine   No. Observations:                  136
Model:                          Logit   Df Residuals:                      132
Method:                           MLE   Df Model:                            3
Date:                Mon, 14 Jun 2021   Pseudo R-squ.:                  0.7928
Time:                        10:12:28   Log-Likelihood:                -19.088
converged:                       True   LL-Null:                       -92.139
Covariance Type:            nonrobust   LLR p-value:                 1.826e-31
===============================================================================
                  coef    std err          z      P>|z|      [0.025      0.975]
-------------------------------------------------------------------------------
const          81.9744    174.407      0.470      0.638    -259.857     423.806
height_right   -4.4641      1.601     -2.788      0.005      -7.603      -1.325
margin_up      -6.6483      2.923     -2.274      0.023     -12.377      -0.919
length          3.5903      0.781      4.598      0.000       2.060       5.121
===============================================================================
 each p-value (P>|z|) below 0.05 => ok
```

Régression logistique

- Test des variables à retenir :
  - cas convergence modèle en 9 itérations

Jeu 2

```
                           Logit Regression Results
==============================================================================
Dep. Variable:             is_genuine   No. Observations:                  136
Model:                          Logit   Df Residuals:                      133
Method:                           MLE   Df Model:                            2
Date:                Mon, 14 Jun 2021   Pseudo R-squ.:                  0.7391
Time:                        10:12:28   Log-Likelihood:                -24.038
converged:                       True   LL-Null:                       -92.139
Covariance Type:            nonrobust   LLR p-value:                 2.654e-30
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const       -414.6860     89.433     -4.637      0.000    -589.971    -239.401
margin_up     -6.1841      2.627     -2.354      0.019     -11.333      -1.035
length         3.8634      0.787      4.909      0.000       2.321       5.406
==============================================================================
 each p-value (P>|z|) below 0.05 => ok
```

Jeu 3

```
                           Logit Regression Results
==============================================================================
Dep. Variable:             is_genuine   No. Observations:                  136
Model:                          Logit   Df Residuals:                      134
Method:                           MLE   Df Model:                            1
Date:                Mon, 14 Jun 2021   Pseudo R-squ.:                  0.7021
Time:                        10:12:28   Log-Likelihood:                -27.452
converged:                       True   LL-Null:                       -92.139
Covariance Type:            nonrobust   LLR p-value:                 5.618e-30
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const       -518.1723     90.389     -5.733      0.000    -695.332    -341.013
length         4.6067      0.803      5.738      0.000       3.033       6.180
==============================================================================
 each p-value (P>|z|) below 0.05 => ok
```

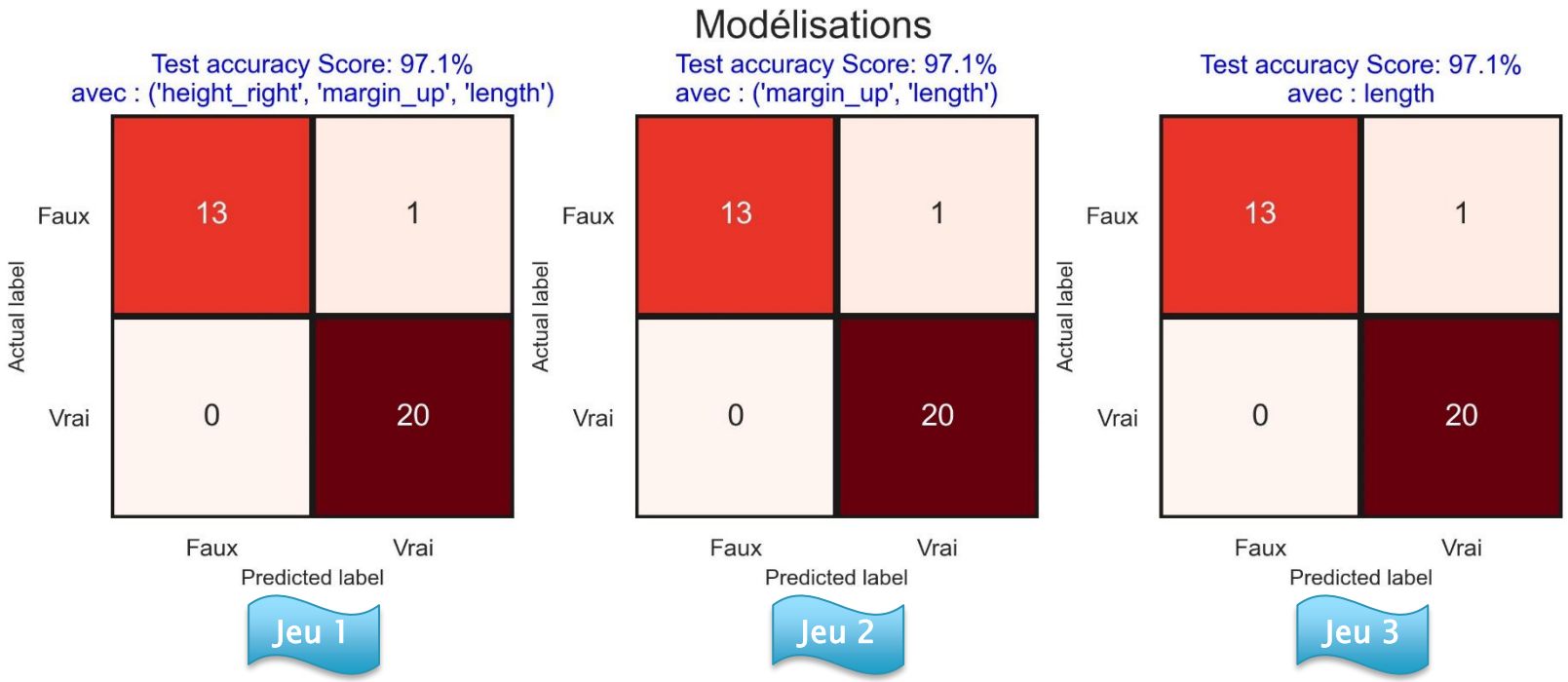
- Bilan :

package	Setting with	McFadden's R^2	pvalue	train accuracy(%)	test accuracy(%)
statsmodels	[const, height_right, margin_up, length]	0.793	1.8e-31	95.6	97.1
statsmodels	[const, margin_up, length]	0.739	2.7e-30	94.1	97.1
statsmodels	[const, length]	0.702	5.6e-30	94.1	97.1

- Avec 1 seule variable explicative *length*, la régression présente la même précision de test qu'avec 3 variables

Régression logistique

- Score et matrice de confusion :



- ∀ le jeu, score inchangé, même répartition/prédiction (1 FP)
- ➔ choix porté sur la 3ème modélisation la plus simple.

Régression logistique

- Prédiction en utilisant la modélisation du 3ème jeu:

```
url_test = 'example.csv' # test_P6.csv example.csv
test_prog = pd.read_csv(url_test, sep=',',encoding='utf-8')
```

Prédire l'authenticité ou non d'un billet et donner la probabilité que le billet soit vrai

```
# Choix du modèle
x_test              = test_prog_[x_trainbis_jet3.columns]          # adaptation du colonnage du dataset de test
predProbaSm         = result_jet3.predict(x_test)                  # prédiction sur l'échantillon test
predSm              = np.where(predProbaSm > 0.5, True, False)     # en prédiction brute
test_prog_['Authenticite_billet'] = predSm                         # ajout nv colonne statut authenticité
test_prog_['Probabilite(%)']      = np.round(100*predProbaSm.values,0)  # ajout nv colonne probabilité
```

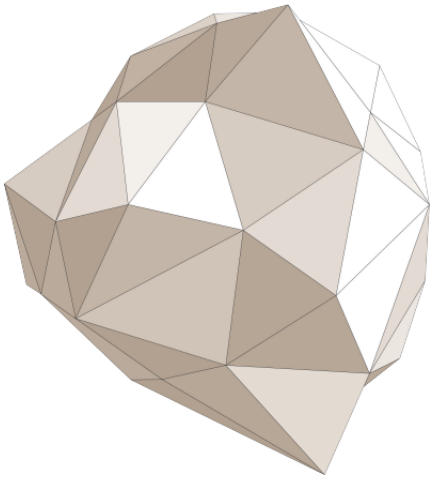
input

example.csv

```
diagonal,height_left,height_right,margin_low,margin_up,length,id
171.76,104.01,103.54,5.21,3.3,111.42,A_1
171.87,104.17,104.13,6.0,3.31,112.09,A_2
172.0,104.58,104.29,4.99,3.39,111.57,A_3
172.49,104.55,104.34,4.44,3.03,113.2,A_4
171.65,103.63,103.56,3.77,3.16,113.33,A_5
171.76,104.01,103.54,5.21,3.3,112.47,A_6
171.87,104.17,104.13,6.0,3.31,112.48,A_7
172.0,104.58,104.29,4.99,3.39,112.49,A_8
172.49,104.55,104.34,4.44,3.03,112.50,A_9
171.65,103.63,103.56,3.77,3.16,112.51,A_10
```

Dans model

id	const	diagonal	height_left	height_right	margin_low	margin_up	length	Authenticite_billet	Probabilite(%)
A_1	1.00	171.76	104.01	103.54	5.21	3.30	111.42	False	1
A_2	1.00	171.87	104.17	104.13	6.00	3.31	112.09	False	14
A_3	1.00	172.00	104.58	104.29	4.99	3.39	111.57	False	1
A_4	1.00	172.49	104.55	104.34	4.44	3.03	113.20	True	96
A_5	1.00	171.65	103.63	103.56	3.77	3.16	113.33	True	98
A_6	1.00	171.76	104.01	103.54	5.21	3.30	112.47	False	49
A_7	1.00	171.87	104.17	104.13	6.00	3.31	112.48	False	50
A_8	1.00	172.00	104.58	104.29	4.99	3.39	112.49	True	51
A_9	1.00	172.49	104.55	104.34	4.44	3.03	112.50	True	52
A_10	1.00	171.65	103.63	103.56	3.77	3.16	112.51	True	53

- Statistique descriptive + ACP + Classification
- Algorithme de détection de faux billets par régression logistique
- Avec l'utilisation seule de la variable *length*, nous pouvons prédire avec une précision acceptable l'authenticité d'un billet

Merci