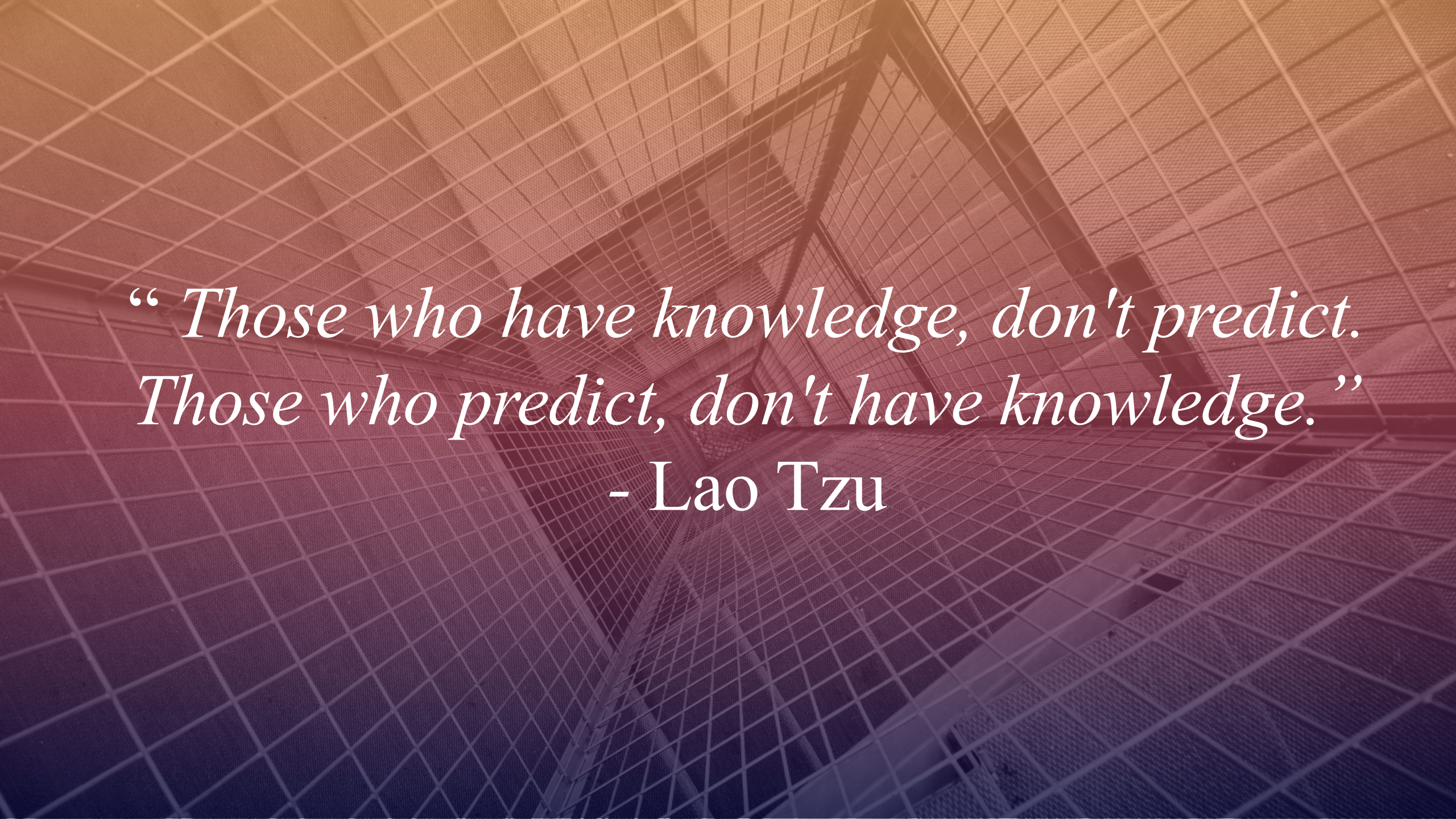
FINANCIAL DISTRESS ANALYSIS

AKHILESH BHAUGEERUTTY


“ Those who have knowledge, don't predict.
Those who predict, don't have knowledge.”
- Lao Tzu

INTRODUCTION

Maintenant plus que jamais, en ces temps de crises économique et financière, les entreprises ont besoin d'avoir une connaissance profonde et une vision du futur du marché afin de survivre.

En utilisant des outils de Machine Learning, on veut analyser et pouvoir prédire la viabilité financière des entreprises. Celles-ci sont en effet soit en '**bonne santé**' ou alors en **'danger financier**' (Financial Distress). Ce problème est donc très intéressant, car grâce à certains signaux envoyés au préalable, une entreprise émettant des signes de danger pourrait survivre.

LES OUTILS ET SOLUTIONS

DATAFRAME

- **DataFrame** utilisé obtenu de Kaggle (Financial Distress.csv)
- 3672 observations de **86 variables**
- **Variable cible**: *Financial Distress* (indice < 0.5 indique une compagnie en danger)
- Autres variables: différents indicateurs financiers (pas de nom de variables)
- - Train/Test : 0.7/0.3, échantillonnage **stratifié** (pour conserver les proportions)

ALGORITHMES

- Problème de **Régression** (variable cible continue)
- Peut-être aussi vu comme une **Classification** (Bonne santé/Danger)
- Ici, on va tester Classification et Régression → Classification
- **Modèles Utilisés**:
  - Random Forests
  - Gradient Boosting Trees
  - Support Vector Machines

→ Modèles choisis car réputés pour être les plus performants sur peu de données

EVALUATION

- **Skewed Data**: Il y a beaucoup plus d'entreprises en bonne santé --> *Accuracy* sera biaisée.
- On utilise donc le ***F-Score*** pour tester la performance et comparer les modèles
- **Cross Validation** pour trouver les meilleurs paramètres pour les algorithmes

DATAFRAME

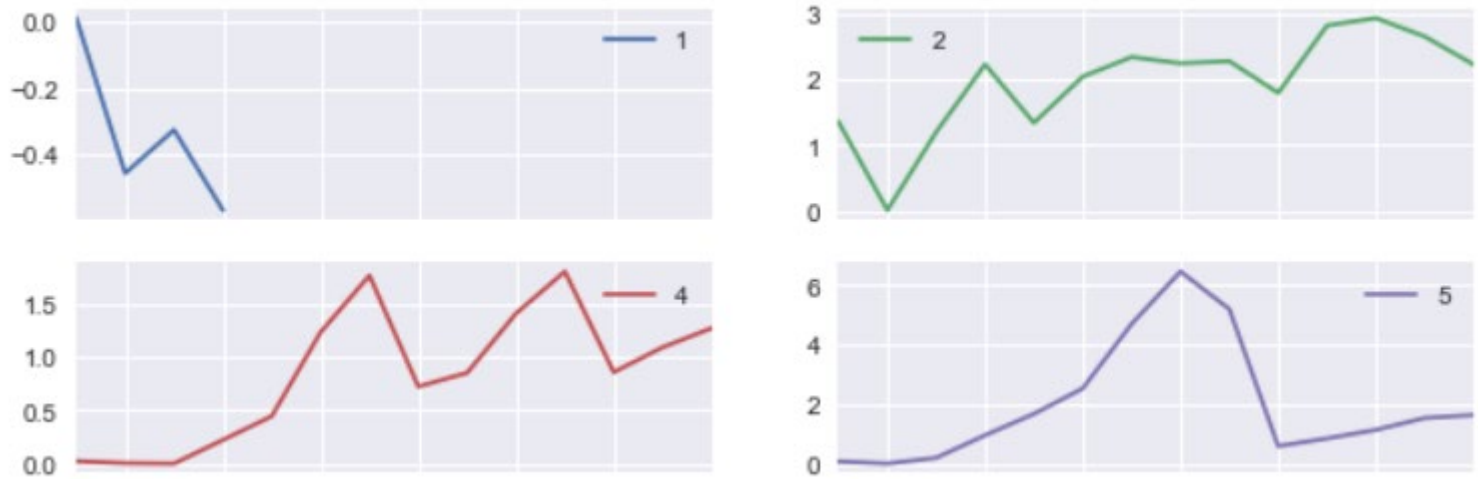
	Company	Time	Financial Distress	x1	x2	x3	x4	x5	x6	x7	...	x74	x75	x76	x77	x78	x79	x80	x81
0	1	1	0.010636	1.2810	0.022934	0.87454	1.21640	0.060940	0.188270	0.52510	...	85.437	27.07	26.102	16.000	16.0	0.2	22	0.060390
1	1	2	-0.455970	1.2700	0.006454	0.82067	1.00490	-0.014080	0.181040	0.62288	...	107.090	31.31	30.194	17.000	16.0	0.4	22	0.010636
2	1	3	-0.325390	1.0529	-0.059379	0.92242	0.72926	0.020476	0.044865	0.43292	...	120.870	36.07	35.273	17.000	15.0	-0.2	22	-0.455970
3	1	4	-0.566570	1.1131	-0.015229	0.85888	0.80974	0.076037	0.091033	0.67546	...	54.806	39.80	38.377	17.167	16.0	5.6	22	-0.325390
4	2	1	1.357300	1.0623	0.107020	0.81460	0.83593	0.199960	0.047800	0.74200	...	85.437	27.07	26.102	16.000	16.0	0.2	29	1.251000

5 rows × 86 columns

- 83 Indicateurs financier continus (sauf x80).
- **Financial Distress**: Variable Cible
- Variables retirées: **Company** (Ajoutera un biais inutile), **x80** (catégorique) et **Time** (voir ci-dessous)

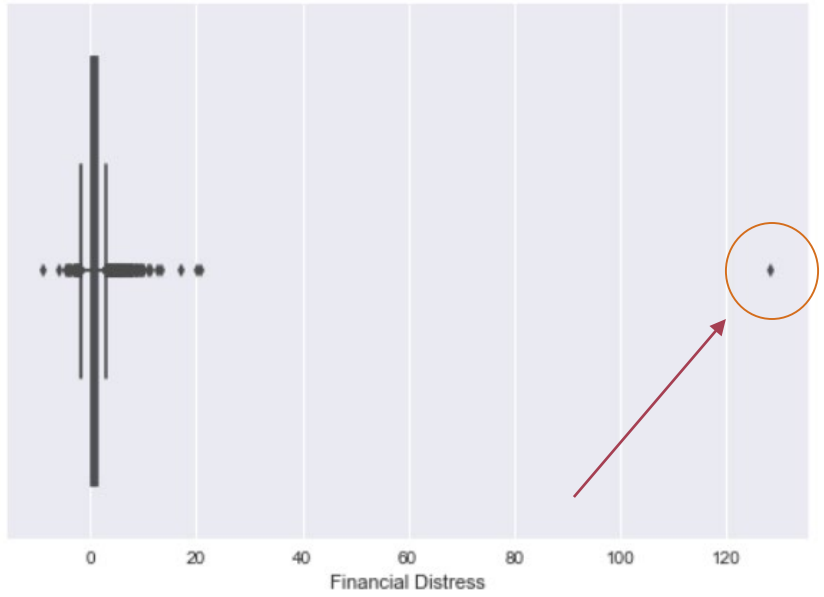
Si le temps avait un effet: le problème peut être une *série temporelle à multivariables*.
Ici, il ne semble pas avoir de vraie corrélation (0.14), ni de tendance similaire donc on supprime la variable.

	Time	Financial Distress
Time	1.0000	0.1442
Financial Distress	0.1442	1.0000



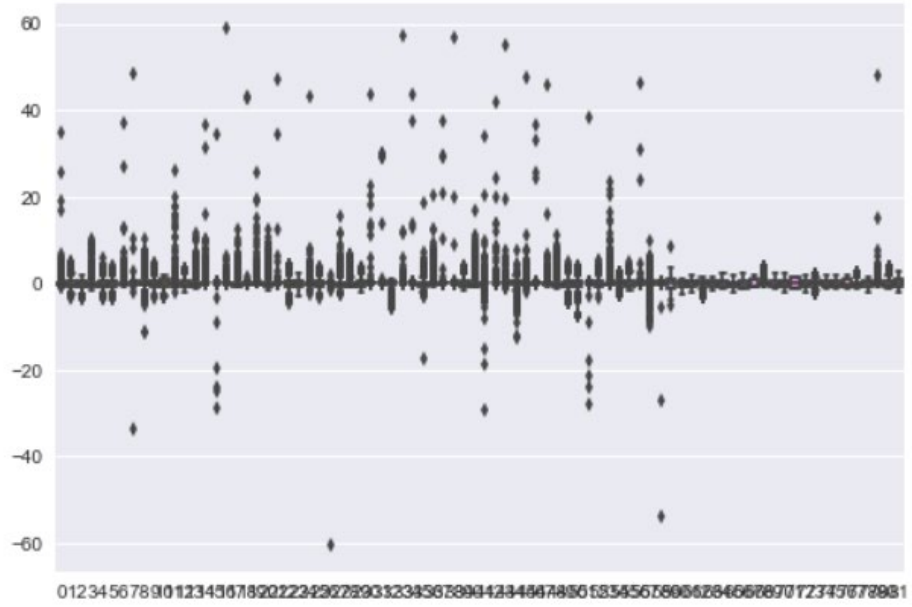
Financial Distress over Time for 4 random companies

PREPARATION DES DONNEES



OUTLIER

- Retiré par précaution



SCALING

- Important car ils accélère les calculs pour Gradient Boosting (algorithme converge + rapidement)

ALGORITHMES

MODELES

- Modèles ayant des algorithmes de **Classification & Régression**:
  - Forêts (Gradient Boosting & Random Forests)
  - Support Vector Machine

- Après la Régression, on utilise la prédiction obtenue pour classifier si en bonne santé ou pas. **On teste plusieurs seuils (thresholds)** (-0.5 n'est peut-être pas le meilleur seuil prédit pour dire si en bonne santé)

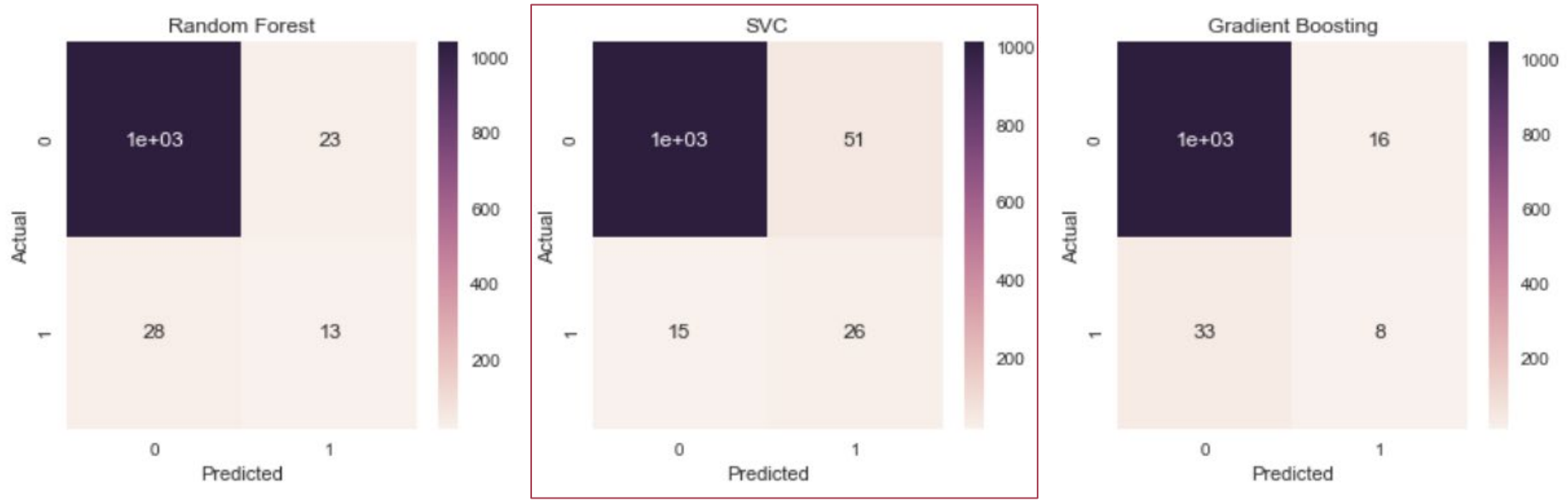
- **Cross Validation** immédiatement effectuée pour avoir les meilleurs paramètres.

REGRESSION
- Random Forest Regressor
- Gradient Boosting Regressor
- Support Vector Regressor

CLASSIFICATION (pas d'algorithme)
(< seuil ou pas)

CLASSIFICATION
- Random Forest Classifier
- Gradient Boosting Classifier
- Support Vector Classifier

PERFORMANCE REVIEW (CLASSIFICATION)



- Accuracy Elevée:

```
RF Accuracy: 0.9537
GB Accuracy: 0.9555
SVC Accuracy: 0.940
```

- F-Score est un meilleur indicateur (*skewed data*)

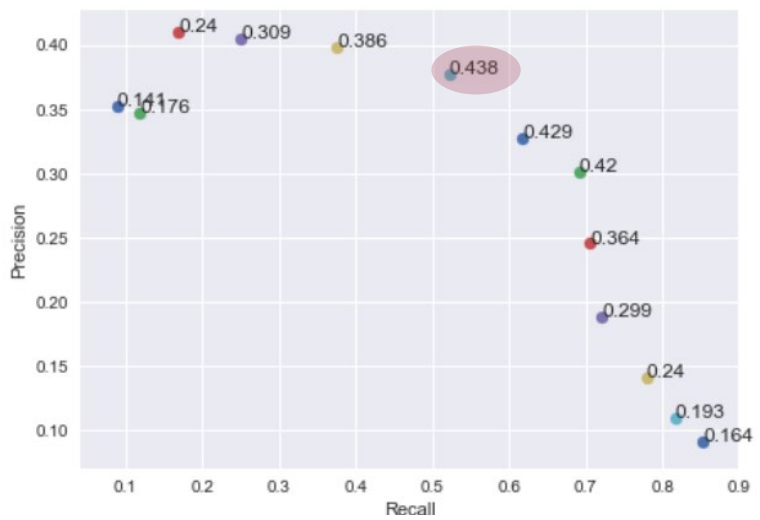
```
RF F-score: 0.33766
GB F-score: 0.24615
SVC F-score: 0.4406
```

- **Support Vector Classifier** est le meilleur algorithme: il réussit le mieux à identifier les entreprises en danger
- **Gradient Boosting**: pas efficace (long temps de calcul) et pas précis.

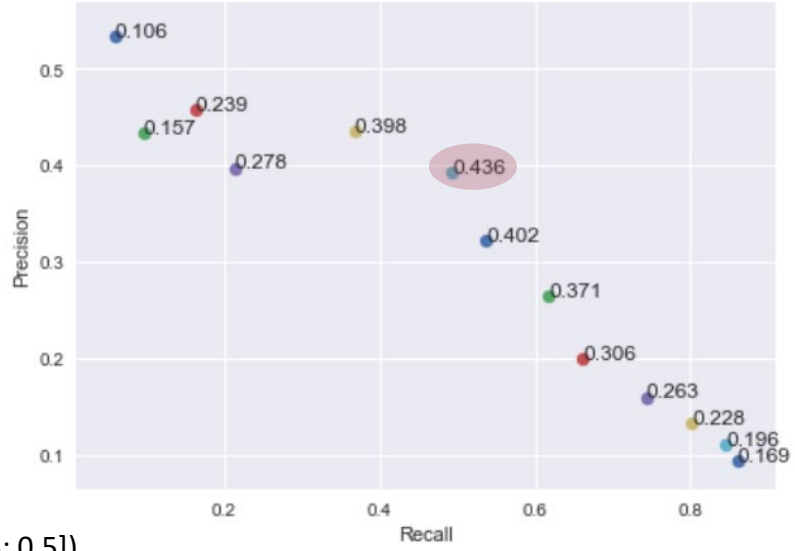
PERFORMANCE REVIEW (REGRESSION)

Random Forest

Gradient Boosting



SVR



F-Score, Precision & Recall en fonction de différents seuils (entre [-0.75; 0.5])

- Régression plus stable que la Classification, le F-Score max est en général plus élevé:

```
RF: 0.45762
GBR: 0.4382
SVR: 0.4364
```

- Meilleur seuil pour les 3 algorithmes: -0.25 → le F-Score est maximal lorsqu'après la régression, on classifie une entreprise comme en bonne santé si son Financial Distress est > -0.25

- **Random Forest**: plus efficace (rapidité + précision)
- **SVR** et **Gradient Boosting** équivalents en précision mais GBR plus rapide

(On privilégie un meilleur Recall car données *unbalanced*)

FEATURE SELECTION

80 features: trop élevé pour peu d'observations.
Etape très importante:

- Limiter le temps de calcul
- Améliorer la précision des modèles
- Réduire le sur-apprentissage

Deux méthodes utilisées:

- **Recursive Feature Elimination** (RFE): Les entités sont classées en fonction de leur importance et en éliminant récursivement les moins importantes, RFE élimine les dépendances et colinéarités (Cross Validation pour obtenir le nombre de features à conserver) → 15 variables conservées

- **Méthode par Forêt Aléatoire**: mesure l'impureté et l'impact de chaque feature sur le modèle. En général robuste et précis. → 19 variables conservées

		Random Forest	Gradient Boosting	Support Vector
Forest	Accuracy	0.961887	0.962795	0.933757
	F1	0.560000	0.327869	0.486504
RFE	Accuracy	0.942831	0.962795	0.961887
	F1	0.463636	0.280702	0.375862

- **Classification**: le F-Score a augmenté pour tous les modèles et avec les deux méthodes de sélection
  → succès

		Random Forest	Gradient Boosting	Support Vector
Forest	F1	0.564000	0.430233	0.443800
	Seuil	-0.250000	-0.250000	-0.050000
RFE	F1	0.452368	0.459333	0.484615
	seuil	-0.250000	-0.250000	-0.050000

- **Régression**: le F-Score a augmenté pour tous les modèles et avec les deux méthodes de sélection (moins marqué que pour classification mais beaucoup plus rapide qu'avant)
  → succès

CONCLUSION

Après avoir établi que la Régression et la Classification étaient toutes deux adaptées pour répondre à notre problème, nous avons choisi nos modèles (Random Forest, Support Vector Machine & Gradient Boosting), qui ont tous la particularité d'avoir des algorithmes de Régression & Classification.

La performance des algorithmes a été mitigée, en majorité due à la qualité des données recensées: les classes (bonne santé/danger) n'étant pas équilibrées. On note tout de fois que la Régression a donnée des résultats plus concluants que la Classification.

La sélection de variables fut ensuite une étape réussie. En effet, même si les résultats ne sont pas nettement meilleurs, nous avons considérablement réduit le nombre de features, (respectivement 15 & 19 au lieu de 83), ce qui a grandement accéléré les calculs. En général, la Random Forest a été le meilleur algorithme, puis les Support Vector Machines, alors que les Gradient Boosting Trees ont rarement impressionné.


MERCI!