FONDATION D'ENTREPRISE FEREC

Démonstrateur d'utilisation de l'Intelligence Artificielle pour la gestion opérationnelle des risques naturels gravitaires géologiques
Projet RINA

RAPPORT DE RECHERCHE / LIVRABLE

Novembre 2021

Auteurs / Organismes :

Marie-Aurélie Chanut / Cerema
Clara Lévy / BRGM
Lucas Meignan / Géolithe
Muriel Gasc / Cerema
Emmanuel Trouvé / LISTIC
Abdourrahmane Atto / LISTIC
Nicolas Meger / LISTIC
Hermann Courteille / LISTIC
Guilherme Cunha de Barros-Santos / Master 2 Géoressources Nancy

Cerema brgm GÉOLITHE LISTIC

IREX INDURA C2ROP Chutes de Blocs Risques Rocheux Ouvrages de Protection

Résumé de l'étude :

Le projet RINA proposé dans le cadre de l'AAP 2020 de la Fondation Férec (Intelligence artificielle appliquée aux infrastructures en service) vise à étudier la faisabilité de l'utilisation de l'intelligence artificielle (IA) dans le contexte des risques géologiques. Sur un site test instable, les données d'observation de l'aléa rocheux et les données climatiques (température, précipitation) sont analysées à l'aide des outils d'intelligence artificielle. L'objectif est de proposer une puissante aide à la décision aux exploitants des réseaux routiers et ferroviaires afin de maintenir la qualité de service sur les infrastructures de transport menacées par des aléas rocheux lors d'événements météorologiques intenses.

Trois types de modèles d'apprentissage automatique ont été mis en œuvre (les plus proches voisins, les arbres de décision et les réseaux de neurones) pour construire des modèles prédictifs de chutes de blocs en lien avec les précipitations sur la Route du Littoral (RN1) à la Réunion. La performance des modèles a été évaluée à l'aide de plusieurs métriques : le rappel, la précision et l'exactitude pondérée et a été comparée à celle de la règle experte utilisée de façon opérationnelle par le service des routes. Les modèles d'arbres de décision et les réseaux de neurones fournissent des résultats similaires à ceux de la règle experte avec des compromis différents entre rappel et précision, la règle experte se situant entre les 2.

Plusieurs perspectives au projet RINA ont été dressées. Sur le site d'étude, d'autres développements plus poussés peuvent être envisagés : d'autres modèles d'arbres de décision ou de réseaux neuronaux peuvent être dérivés en fonction de la taille des blocs éboulés car les processus de déclenchement sont différents entre les petits et les gros blocs. La question du transfert de ces modèles d'IA sur d'autres périodes avec moins d'éboulements (comme la période 2008-2020 sur le site de la RN1 à la Réunion) pourrait être conduite pour évaluer la capacité de prédiction des modèles avec beaucoup moins d'événements. La dernière perspective concerne deux sujets qui sont des thèmes de recherche plus larges au sein de la communauté de l'IA : la question de l'explicabilité des modèles d'IA et le transfert des modèles par apprentissage sur d'autres sites. Ces deux sujets sont au centre du projet « C2R-IA » déposé à l'AAPG 2022 de l'ANR au sein de l'axe H.16 : Interfaces : mathématiques, sciences du numérique – sciences du système Terre et de l'environnement.

Les développements réalisés dans le cadre du projet RINA ainsi que les résultats obtenus sont présentés de façon synthétique dans ce rapport. En annexe, figurent deux rapports réalisés dans le cadre du projet RINA : celui d'un stage de Master 2 orienté Risques Rocheux et celui du contrat orienté IA réalisé par un ingénieur de recherche.

Les partenaires du projet RINA souhaitent remercier la Fondation Ferec pour avoir subventionné ce projet et ainsi initié une dynamique forte entre acteurs du risque rocheux et de l'intelligence artificielle. Ils remercient sincèrement le Conseil Régional de la Réunion qui a mis à disposition la base de données de chutes de blocs collectée sur la Route Nationale 1 de la Réunion ainsi que les données pluviométriques. Ce travail a été mené dans le cadre du projet national C2ROP (Chute de blocs, Risque Rocheux et Ouvrages de Protection) et les partenaires de RINA remercient l'IREX et INDURA pour leur soutien ainsi que les maitres d'ouvrages SNCF et CD 73.

SOMMAIRE

1. CONTEXTE ET OBJECTIFS DU PROJET RINA

En montagne, les infrastructures de transport sont exposées à des risques gravitaires (éboulements rocheux, laves torrentielles, avalanches de neige) qui peuvent entraîner leur fermeture ainsi que des dommages significatifs aux biens et aux personnes. Dans un contexte de changement climatique, on constate une recrudescence des événements gravitaires en lien avec la remontée de la limite pluie-neige et la recrudescence d'événements pluvieux violents. Les acteurs du risque rocheux, fédérés au sein du projet national C2ROP, ont réalisé des avancées considérables sur la gestion du risque depuis 2015. Le projet RINA vient enrichir cette dynamique en investiguant l'utilisation de l'IA.

Le projet vise à développer des modèles prédictifs basés sur l'IA pour maintenir la qualité de service sur les infrastructures de transport menacées par des événements rocheux lors d'épisodes météorologiques intenses. Il s'agit de permettre aux gestionnaires d'infrastructures d'anticiper une évolution défavorable de l'aléa afin de mettre en œuvre des dispositifs de mitigation des risques (limitation d'accès, surveillance, mobilisation de kits d'urgence, maintenance prédictive).

Les données météorologiques et d'observation de l'aléa rocheux disponibles sont de plus en plus massives avec les progrès technologiques des moyens de mesure. Les radars météorologiques fournissent des données à haute fréquence temporelle à l'échelle d'un massif. Concernant l'aléa rocheux, des technologies récentes (radars terrestres, scanners laser ou photogrammétrie) fournissent une information spatialisée des mouvements. Dans ce contexte, des défis se posent dans l'intégration des caractéristiques multi-physiques de ces données, qui sont nécessaires pour fournir une compréhension cohérente et une prédiction fiable des phénomènes et de leurs impacts. Les approches standards reposant sur la détermination de modèles multi-échelles et multi-physiques sont assez complexes pour intégrer les couplages thermiques, hydriques et mécaniques et leur capacité est limitée à couvrir des échelles allant du site au bassin de risque. Compte tenu du volume et de la variété des données, des approches issues de l'IA et de la « data science » (en particulier l'apprentissage automatique et sa branche « deep learning ») semblent être un moyen prometteur pour induire des données elles-mêmes des modèles prédictifs du risque.

Les verrous scientifiques de ce projet résident dans le traitement de données massives (multi-sources, fréquences d'acquisition différentes) en géosciences par IA. Par ailleurs, le potentiel réel des méthodes associant l'IA pour des applications pratiques en géosciences n'a pas encore été démontré. Enfin, les chutes de blocs sont des évènements rares ce qui rend l'entrainement des modèles plus difficiles.

Dans le cadre du projet RINA, les résultats attendus concernent une preuve de concept des modèles IA pour la prédiction des chutes de blocs en lien avec les facteurs climatiques. En particulier, trois points sont à établir :

- Définition des données et de leurs traitements pertinents,
- Identification des méthodes d'Intelligence Artificielle pertinentes,
- Pertinence de l'analyse prédictive par IA avec comparaison des résultats et des faits.

Au-delà du démonstrateur, le projet RINA a pour ambition d'établir une base technique et scientifique pour la soumission d'un projet à l'AAPG 2022 de l'ANR, en complémentarité du montage de la suite du PN C2ROP.

2. ORGANISATION ET ACTEURS

Le projet RINA est porté par trois partenaires du domaine du risque rocheux (Cerema, BRGM et Géolithe). Une des ambitions du projet était d'établir un consortium plus large intégrant les compétences du domaine de l'intelligence artificielle pour mettre en place le démonstrateur affiché. Le partenaire IA intégré dans le cadre du projet RINA est le laboratoire LISTIC (Laboratoire d'Informatique, Systèmes, Traitement de l'Information et de la Connaissance de l'université Savoie Mont-Blanc).

Le projet RINA, subventionné par la fondation Ferec, a permis de mobiliser essentiellement des compétences en IA (CDD ingénieur de recherche de 4 mois au LISTIC – Hermann Courteille et accompagnement du LISTIC) et en risque rocheux (stage de master 2 de 6 mois - Guilherme Cunha de Barros-Santos) en plus de l'investissement des partenaires.

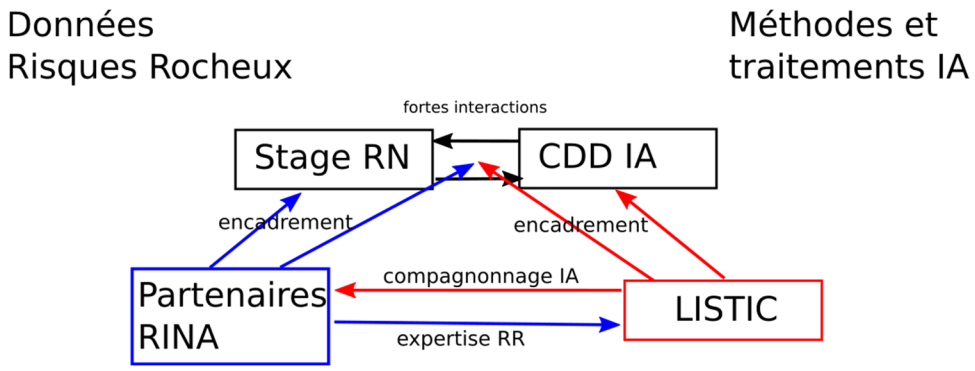

Figure 1 : Organisation et acteurs du projet RINA

3. DEVELOPPEMENTS

3.1 Site d'étude et données utilisées

Le site d'étude considéré dans le cadre du projet RINA est la route nationale 1 sur l'île de la Réunion (Figure 2). La route nationale 1 est situé au Nord de l'ile et relie Saint-Denis à la Possession en 13 km. Sur la majorité du tracé, la route est surplombée par des falaises (Figure 3) d'une hauteur comprise entre 40 m et 200 m.

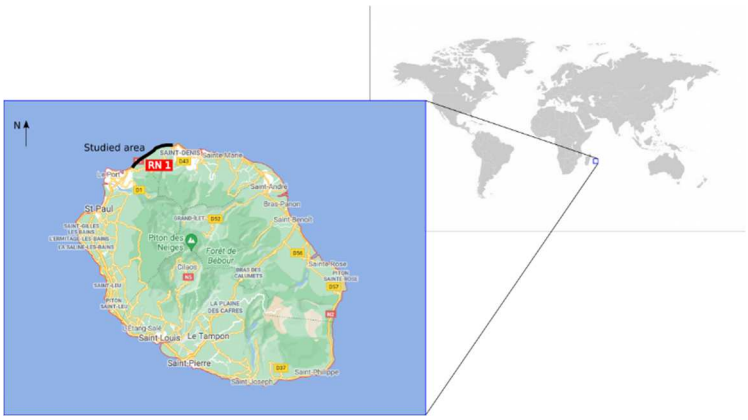

Figure 2 : Localisation de l'ile de la Réunion

Figure 3 : falaise surplombant la RN1

Ce site a été retenu pour deux raisons. D'une part, c'est un site de chutes de blocs particulièrement productif (sur la période étudiée, environ 13% des jours se produisent des chutes de blocs) même si les journées avec évènements sont minoritaires. D'autre part, il a été montré que les précipitations sont le facteur déclenchant prépondérant (Delonca et al, 2014).

Les chutes de blocs (**Erreur ! Source du renvoi introuvable.**) atteignent principalement la route coté montagne. Le service de gestion de la route du Conseil Départemental de la Réunion enregistre la date des chutes de blocs, leur masse (estimée) ainsi que la localisation (PR) le long de la route. Dans le projet, les données utilisées sont donc la base de données constituée par le Conseil Régional de la Réunion sur la période 2000 à 2007 avant la mise en place massive de structures de protection le long de la route. Après cette phase de protection, le nombre de chutes de blocs a été considérablement réduit puisque la production de chutes de blocs a été modifié par les parades actives (empêchement des blocs de se détacher) ou passives (interception lors de leur propagation). Les précipitations quotidiennes sont également des données d'entrée et sont mesurées sur trois pluviomètres installés le long de la route. Sur la Figure 5, on constate que les chutes de blocs ne sont pas réparties uniformément le long de la route. Une analyse spatio-temporelle serait intéressante à mener en prenant en compte la géologie, la présence de protection et la géométrie de propagation des blocs. Mais dans un premier temps, seule une analyse temporelle a été menée.

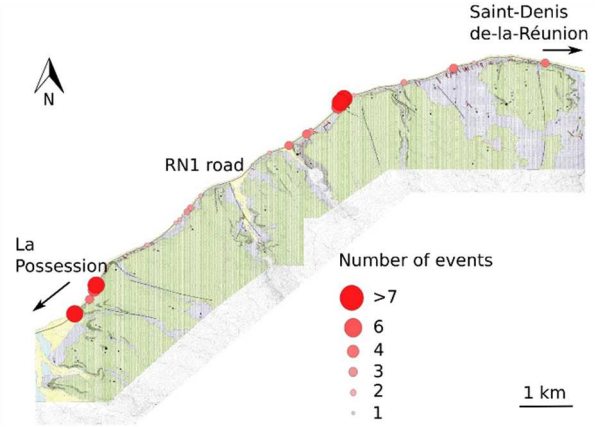

Figure 5 : Distribution des chutes de blocs le long de la RN1

3.2 Analyse descriptive des données

Avant de décrire les modèles IA utilisés, une analyse descriptive des données permet de vérifier et comprendre les données. La corrélation entre précipitations et chutes de blocs est mise en évidence sur la Figure 6. Les nombres d'évènements de chutes de blocs augmentent lorsqu'il pleut plus. La Figure 7 montre la distribution des précipitations les jours avec des chutes de blocs (rouge) et les jours sans chutes de blocs (bleu). On constate que les deux distributions sont très différentes et que la moyenne des précipitations les jours avec chutes de blocs est dix fois supérieure à la valeur moyenne des jours sans chutes de blocs qui est très faible soit 2 mm. On remarque néanmoins que des chutes de blocs se produisent sans précipitations les jours précédents l'évènement. C'est le bruit existant sur le site et ces chutes de blocs ne pourront pas être expliquées par le facteur de forçage étudié dans ce projet à savoir les précipitations.

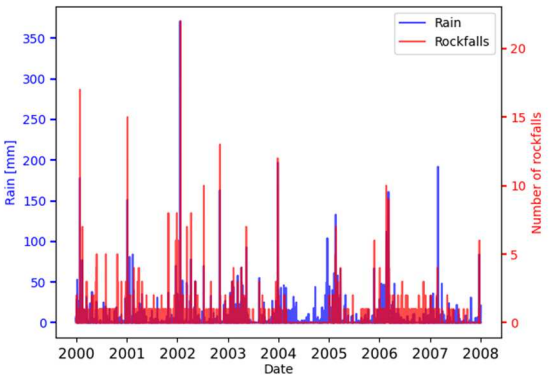

Figure 6 : Précipitations et chutes de blocs quotidiens entre 2000 et 2008

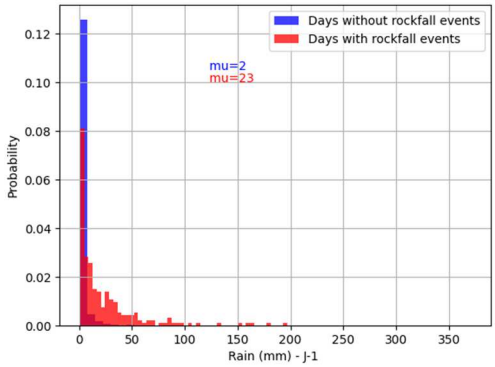

Figure 7 : Distribution des précipitations les jours avec chutes de blocs (rouge) et sans chutes de blocs (bleu)

3.3 Modèles IA et métriques

Dans cette partie, nous présentons les modèles d'intelligence artificielle testés dans le cadre du projet RINA qui visent à construire des modèles prédictifs de chutes de blocs à partir des précipitations des jours précédents.

Il s'agit en fait de répondre à la question : y-aura-t-il une ou plusieurs chutes de blocs le jour J en fonction des précipitations des jours précédents (de J-1 à J-5 ou J-10) ? Les données d'entrée de ces modèles sont donc uniquement les précipitations des jours précédents (on utilise un délai entre 5 et 10 jours). En premier lieu, la sortie est une prédiction booléenne indiquant s'il y a au moins une chute de blocs. Les modèles sont donc des classifieurs binaire.

3.3.1 Modèle de référence

Avant de décrire les modèles IA, le modèle prédictif de référence est la règle de gestion experte mise en place en 2004 (Batista, 2016) et utilisé pendant la période d'étude (2000-2007) par le service de gestion de la route. Cette règle définit une fermeture de 72 heures de la route coté montagne si le cumul des précipitations sur 24 h est supérieur à 30 mm et la fermeture est réduite à 24 heures si ce cumul est compris entre 15 et 30 mm.

3.3.2 Modèles machine learning

Trois types de modèles machine learning ont été utilisés : les modèles des plus proches voisins, des arbres de décision et les réseaux de neurones issus du deep learning.

Pour les modèles type plus proches voisins (Figure 8), la prédiction pour un nouvel individu est basée sur la proximité avec des individus connus. La taille du voisinage est dans ce cas à optimiser.

Pour les modèles type arbre de décision (Figure 9), le classificateur à arbre simple consiste en une série de décisions logiques binaires afin de choisir la catégorie qu'il va prédire. Chaque décision est basée simplement sur l'une des données d'entrée. Des modèles plus sophistiqués avec assemblage (comme la forêt aléatoire) ou le classificateur à gradient basé sur l'histogramme ont été optimisés pour obtenir les meilleurs résultats.

Pour l'apprentissage profond, nous utilisons des réseaux neuronaux (Figure 10). Les réseaux de neurones se composent d'une première couche, avec un neurone par donnée d'entrée, d'une série de couches avec différents neurones et d'une couche finale avec la sortie. Dans notre cas, la sortie est binaire : éboulement ou pas ? Chaque neurone reçoit des valeurs de la couche précédente, les agrège et envoie une valeur à la couche suivante. Il est caractérisé par des poids, un biais et une fonction d'activation. Deux modèles ont été testés : les réseaux de neurones dense (DNN) et les réseaux de neurones convolutifs (CNN).

Avant la calibration de ces modèles, une optimisation des hyperparamètres est effectuée. Les hyperparamètres sont par exemple le nombre de voisins, la profondeur de l'arbre, le nombre de couches cachées ou le nombre de neurones par couche.

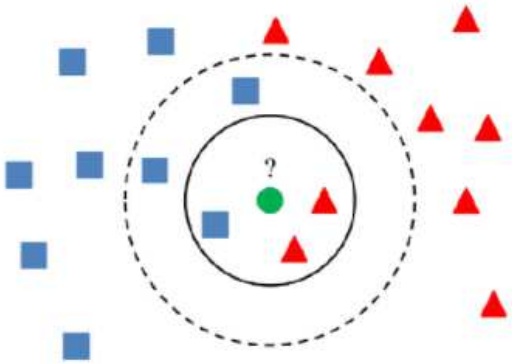

Figure 8: modèle type proches voisins

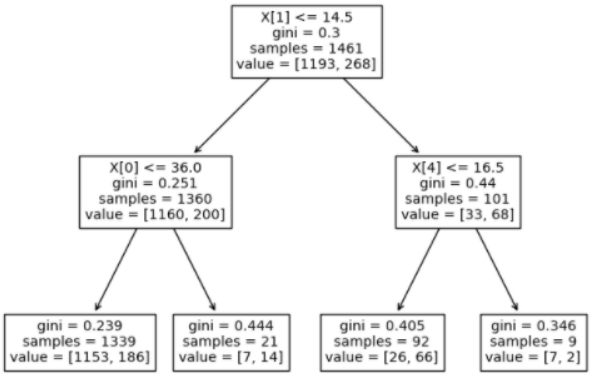

Figure 9 : modèle type arbre de décision

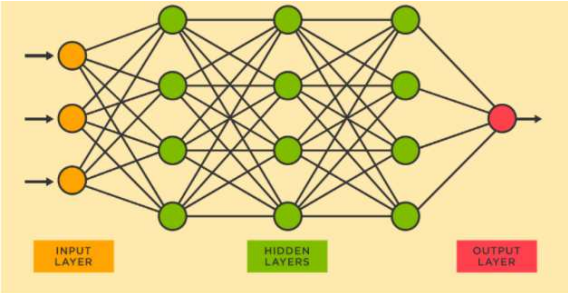

Figure 10 : modèle type réseau de neurones

3.3.3 Métriques d'évaluation des modèles

Les modèles prédictifs sont construits sur un ensemble d'apprentissage (données de 2000 à 2004) et évalués sur les données de test (données de 2005 à 2007), ces deux ensembles arborant un comportement stationnaire dans le temps. Les performances des modèles sont évaluées à l'aide de la matrice de confusion calculée sur les données de test (Tableau 1). La matrice de confusion est un tableau de deux lignes et deux colonnes qui indique le nombre de :

- vrais positifs (VP) : jours avec événements et le modèle prédit des événements,
- vrais négatifs (VN) : jours sans événements et le modèle ne prédit aucun événement,
- faux positifs (FP) : jours sans événements et le modèle prédit des événements,
- faux négatifs (FN) : jours avec événements et le modèle ne prédit aucun événement.

		Réalité	
		Pas de chutes de blocs	Chutes de blocs
Prédic-tions	Pas de chutes de blocs	Vrai négatif (VN)	Faux négatif (FN)
	Chutes de blocs	Faux positif (FP)	Vrai positif (VP)

Tableau 1 : Matrice de confusion

Nous avons utilisé trois métriques, dérivées de la matrice de confusion pour évaluer la performance :

1. **Le rappel** permet de détecter la fraction d'événements de chutes de blocs prédits parmi tous les événements ayant eu lieu. Une augmentation du rappel signifie que nous sommes capables de mieux détecter les événements et donc d'améliorer la sécurité sur la route.

   $$rappel = \frac{VP}{VP+FN}$$

2. **La précision** est la fraction de chutes de blocs réelles parmi tous les événements prédits. Une augmentation de la précision signifie que nous diminuons le nombre de faux positifs et que nous réduisons le nombre de jours de fermeture de la route inutiles puisque sans évènements réels. La conséquence directe d'une augmentation de la précision présente un avantage économique.

   $$précision = \frac{VP}{VP + FP}$$

3. L'exactitude est la fraction de bonnes prédictions parmi l'ensemble ("Accuracy" en anglais). Au vu du déséquilibre des classes dans nos jeux de données, on lui préfère l'exactitude pondérée.

   $$exactitude\ \ pondérée = \frac{1}{2}\left(\frac{VN}{VN+FP} + \frac{VP}{VP+FN}\right)$$

3.4 Résultats

Pour chaque type de modèles, une optimisation des hyper-paramètres a été conduite. Puis les modèles ont été entrainés sur le jeu de données d'entrainement (2000-2004) avant d'être évalués (2005-2007). La capacité de prédiction des différents modèles est maintenant présentée.

3.4.1 Modèle expert

Les performances du modèle de référence (modèle expert) sont d'abord calculées. La règle opérationnelle implémentée sur les données de test, indique que les événements de chutes de blocs se produisent le jour J si les précipitations du jour précédent J-1 sont supérieures à 15 mm ou si les précipitations des jours J-2 ou J-3 sont supérieures à 30 mm

Parmi les 146 jours avec des événements (données de test entre 2005 et 2007), le modèle expert a permis d'en identifier 43 (Tableau 2). L'exactitude pondérée est égale à 0,6, le rappel vaut 0,3 et la précision 0,37 (Tableau 6).

		Réalité	
		Pas de chutes de blocs	Chutes de blocs
Prédictions	Pas de chutes de blocs	870	103
	Chutes de blocs	74	43

Tableau 2 : Matrice de confusion – Modèle expert (référence)

3.4.2 Modèle des plus proches voisins

Pour le modèle des plus proches voisins, le meilleur modèle est obtenu après réglage des hyper paramètres avec 15 voisins. Dans ce cas, 30 chutes de pierres peuvent être identifiées (Tableau 3) avec un taux de prédiction similaire, un rappel inférieur et une précision supérieure à celle du modèle expert (Tableau 6). Comme moins d'éboulements sont prédits avec ce type de modèle, la sécurité des usagers de la route diminue. Mais en même temps, la route est moins fermée quand il n'y a pas de chutes de blocs et les intérêts économiques sont moins contraints.

		Réalité	
		Pas de chutes de blocs	Chutes de blocs
Prédic-tions	Pas de chutes de blocs	917	116
	Chutes de blocs	27	30

Tableau 3 : Matrice de confusion – Modèle des plus proches voisins

3.4.3 Modèle type arbre de décision

Pour les modèles d'arbre de décision, le meilleur modèle est un classificateur d'arbre de décision de type assemblage d'arbres (bagging) défini par une profondeur maximale égale à 3, une valeur minimale d'individus par feuilles égale à 9 et un nombre d'estimateurs égal à 10. Dans ce cas, 50 chutes de blocs sont identifiées (Tableau 4). La valeur de rappel est supérieure à celle calculée avec le modèle de référence (Tableau 6) : plus d'éboulements sont correctement prévus et la sécurité des conducteurs est améliorée. En même temps, la valeur de précision est également supérieure ce qui indique que l'on prédit moins de chutes de blocs quand il n'y en a pas ce qui a pour conséquence de moins fermer la route inutilement.

		Réalité	
		Pas de chutes de blocs	Chutes de blocs
Prédic-tions	Pas de chutes de blocs	863	96
	Chutes de blocs	81	50

Tableau 4 : : Matrice de confusion – Modèle type arbre de décisions

3.4.4 Modèle type réseau de neurones

Pour les réseaux neuronaux, le meilleur modèle est obtenu avec un réseau neuronal dense composé d'une couche cachée de 32 neurones, dont les entrées sont les 10 dernières pluviométries journalières Dans ce cas, 62 éboulements sont prédits (Tableau 5) avec un rappel élevé (Tableau 6). Comme plus d'évènements chutes de blocs sont anticipés par rapport à la règle experte, la sécurité de la route est augmentée. Mais la précision du modèle étant plus faible, la route est fermée plus souvent inutilement.

		Réalité	
		Pas de chutes de blocs	Chutes de blocs
Prédictions	Pas de chutes de blocs	805	80
	Chutes de blocs	133	62

Tableau 5 : Matrice de confusion – Modèle type réseau de neurones

3.4.5 Comparaison des résultats

Le

Tableau *6 : Métriques des modèles plus proches voisins, type arbre de décision, type réseau de neurones et expert sur le jeu test des données*

résume les résultats obtenus pour les trois modèles type machine learning . Nous pouvons constater que des résultats comparables ou légèrement meilleurs ont été obtenus avec les modèles IA par rapport à la règle experte. L'exactitude pondérée est légèrement supérieure pour les modèles de type arbre de décision et réseaux de neurones. Dans ces cas, le rappel est également amélioré et la précision est similaire.

	Plus proches voisins	Arbres de décision	Réseau de neurones	Expert
Exactitude pondérée	0,59	0,63	0.65	0,6
Rappel	0,21	0,34	0,44	0,3
Précision	0,53	0,38	0,32	0,37

Tableau 6 : Métriques des modèles plus proches voisins, type arbre de décision, type réseau de neurones et expert sur le jeu test des données

4. CONCLUSIONS ET PERSPECTIVES

Nous avons vu que les modèles d'arbres de décision ou les réseaux de neurones fournissent des résultats similaires à ceux de la règle experte avec des compromis différents entre rappel

et précision, la règle experte se situant entre les 2. Si nous comparons les jours de fermeture de la route prédits par la règle experte et les modèles d'arbre de décision avec les événements réels (fermeture de la route), nous voyons que tous les événements ne sont pas prédits mais que la prédiction est similaire en terme de fermeture de la route entre la règle experte et les modèles d'arbre de décision (Figure 11). Ces résultats montrent donc la capacité de construire des modèles prédictifs IA avec des performances similaires à celles des modèles experts. Les modèles IA sont développés rapidement et s'avèrent aussi efficaces que les modèles experts. Ces outils d'IA peuvent donc être utilement développés pour d'autres sites avec moins de connaissances expertes.

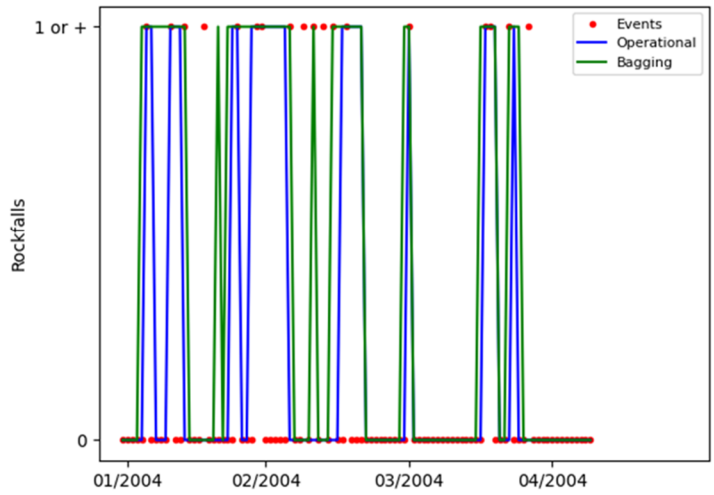

Figure 11 : Comparaison des prédictions des modèles, dans leur ordre chronologique, expert et type arbre de décisions avec les chutes de blocs réelles.

Dans le cadre du projet RINA, outre les développements obtenus sur le site de la RN1 à la Réunion, une méthodologie complète a été mise en place pour traiter les données (événements et données météorologiques) : de la définition des types de modèles IA, à la manière d'optimiser les hyperparamètres et les paramètres des modèles, d'évaluer la performance du modèle et également de traduire les résultats en termes opérationnels. Ce cadre méthodologique peut être réutilisé pour d'autres sites avec des données similaires.

Plusieurs perspectives au projet RINA peuvent être dressées. Sur le site d'étude, d'autres développements plus poussés peuvent être envisagés. En particulier, d'autres modèles d'arbres de décision ou de réseaux neuronaux peuvent être dérivés en fonction de la taille des blocs éboulés car les processus de déclenchement sont différents entre les petits et les gros blocs. Par exemple, le nombre de jours précédents pris en compte peut être augmenté.

La question du transfert de ces modèles d'IA sur d'autres périodes avec moins d'éboulements (comme la période 2008-2020 sur le site de la RN1 à la Réunion) pourrait être conduite pour évaluer la capacité de prédiction des modèles avec beaucoup moins d'événements.

Enfin, deux sujets sont des thèmes de recherche plus larges au sein de la communauté de l'IA : la question de l'explicabilité des modèles d'IA et le transfert des modèles par apprentissage sur d'autres sites. Ces deux sujets sont au centre du projet déposé à l'AAPG 2022 de

l'ANR au sein de l'axe H.16 : Interfaces : mathématiques, sciences du numérique – sciences du système Terre et de l'environnement. Le projet RINA a en effet permis la construction d'un consortium solide. Deux autres partenaires, les laboratoires ISTERRE (Grenoble/Chambéry) et le LIRIS (Lyon) se sont associés à cette dynamique. Indiquer le résumé de la pré-proposition quand il sera stabilisé. Le résumé de la préproposition soumise est indiqué ci-dessous.

« Actuellement, la gestion du risque rocheux est majoritairement abordée par le biais de la construction d'ouvrages de protection, qui représente un coût souvent démesuré par rapport aux ressources financières des communes et opérateurs privés et constitue une solution impossible à développer sur l'ensemble d'un bassin de risque. Une stratégie plus durable et agile de gestion du risque rocheux serait de prendre en compte l'influence des conditions météorologiques sur le niveau d'aléa et d'organiser la mise en œuvre de dispositifs de mitigation du risque en cas d'évolution défavorable (fermeture temporaire d'itinéraires routiers, mise en place de protections temporaires, etc.). Ce type de gestion dynamique du risque étant potentiellement associée à un cout socio-économique élevé, sa mise en place nécessite un processus de décisions motivées.

L'objectif de ce projet est de développer les moyens de passer d'une prise de décision à dire d'expert à une connaissance fine des relations entre l'occurrence d'éboulements et les forçages climatiques, pour établir des modèles de comportement et faire de la prédiction.

Cette problématique est abordée à travers les 3 axes du projet:

-Fiabiliser la détection des éboulements, notamment avec l'utilisation des progrès technologiques pour l'acquisition de données massives et variées et leur exploitation par des méthodes de traitement améliorées, ainsi que par la confrontation des différentes sources de données (fusion de données).

-Développer, à l'aide d'innovations en Intelligence Artificielle (IA), des modèles prédictifs efficaces et avec des résultats interprétables par l'expert et donc aboutissant à leur traduction en potentiels coûts socio-économiques et règles opérationnelles de gestion du risque.

-Adapter les modèles prédictifs à de nouveaux sites à moindre coût en appliquant des méthodes IA innovantes de transfert d'apprentissage, ainsi qu'en testant la pertinence des données issues de dispositifs bas coût pour leur entraînement »

5. RÉFÉRENCES BIBLIOGRAPHIQUES

Seules les références citées dans le rapport sont indiquées ici mais le travail s'est basé sur une bibliographie plus large citée dans les rapports en annexes.

Batista,, Dominique. 2016. Analyse statistique de l'aléa chute de blocs sur la Route du Littoral Île de la Réunion : analyse de l'aléa résiduel après sécurisation et optimisation des règles de basculement. Rapport d' expertise

Delonca, A., Gunzburger, Y., Verdel, T., 2014. Statistical correlation between meteorological and rockfall databases, Natural Hazards and Earth System Sciences, 14,8, 1953--1964, 10.5194/nhess-14-1953-2014

6. ANNEXES

6.1 Rapport de stage orienté Risque rocheux

Prediction of rock falls using artificial intelligence

Author: Guilherme CUNHA DE BARROS SANTOS
Tuteurs: Marie-Aurélie CHANUT, Olivier DECK

Acknowledgments

Firstly I thank my family that made me the person I am today. They encourage me during tough times, helping me overcome the challenges that life had for me, and also rejoicing with my victories. Special thanks to my loving parents that taught me at an early age the importance of studying and seeking knowledge.

I present my gratitude to my teachers that have the arduous and vital job of molding future generations. From all the great teachers that I had through my studies 2 stand out. The first is Elisa Sotelino. She was my materials resistance teacher at PUC-rio and my supervisor during a research project I participated in. She always encouraged me to pursue my dreams and helped guide me through the research. The second teacher that I admire is Olivier DECK who campaigned me through my research about embankment stability.

I am also thankful for the team that guided me through this project. This research is a collaboration of CEREMA, GEOLITHE, BRGM and Université Savoie Mont Blanc. Each participant collaborated with their knowledge and their expertise. Special thanks to Marie-Aurélie who accompanied me during this project with all her capabilities and experience.

Abstract

Rock falls cause numerous accidents around the world. A location where the risk is increased are roads where, due to modifications of topography, there are unstable bedrocks. One method of avoiding accidents is closing the road on days the risk is too high. In order to create the operational code statistical methods are utilized. This study will explore the capacities of artificial intelligence (AI) in order to better understand rock fall events and to compare the capacities of different statistical methods to predict the events.

The first step was selecting which site to study. The RN1 from la Réunion was chosen due to its detailed data set with many events. The data needed to be acquired from different partners. When the data was transferred it could then be formatted and then analysed. It is important to do a simple analysis to have a good understanding of the situation and to spot any early tendencies and possible incorrect data. Subsequently predictive models can be created and finally analysed in order to draw conclusions.

3 different methods were used to create models, nearest neighbors, decision trees and NNs. For each method different models were tuned in order to produce an optimal performance for predictions over the test set. The results were summarized on confusion matrices and the models were compared between each other and with the operational system.

This work permits us to observe that AI is capable of producing models with great performance, but further studies are needed in order to use it operationally.

Summary

1. Introduction

1.1. Rock falls

Rock falls occur when a block of rock detaches from a bedrock and descends by free fall or leaping (Varnes 1978). They are quick and brutal events different from other slope movements. It occurs due to erosion that causes the mass to depart the cliff. Rock fall is a problem when it can reach a target. Rocks that fall in the middle of the forest are not studied, but those which target roads or villages must be studied in order to avoid accidents. This phenomena is frequently studied, it is hard to have accurate predictions due to its infrequent and discontinuous nature (Luckman 2013).

Along the years of studies, geomorphologists separated two classes of rock slides based on their origin (Luckman 2013). The first are primary rockfalls, when the volume detaches from the cliff and falls immediately after. The secondary rockfalls occur when there is a declaration between the detachment and the fall. The block detaches from the bedrock, due to weathering and other processes, but remains stationary until another factor causes it to fall.

Causes of rock falls

Rock blocs fall when the driving force applied on it is greater than the resistance holding it back. It is not always possible to say what the cause is, as it might be a combination of different factors. The final factor is simply a spark that triggered the event that was already on the verge of happening (Varnes 1978). The main factors are briefly outlined in this section.

The removal of support disrupts the equilibrium of forces and may create weaknesses that cause blocks to detach with time. This may happen for various reasons like erosion, previous movements of the slope or human activity. Additional charges will also disrupt the stability of the bedrock. Humans can be responsible for this by constructing structures, leaking water from pipelines, sewers, canals and reservoirs, and other activities. Natural agents that create surcharges include, weight of precipitation, pressure caused by rain on cracks, freeze thaw cycles, clay expanding due to hydration, accumulation of volcanic material and vegetation. Another factor is vibrations of the rockbed. This includes natural tremors like earthquakes and also the ones caused by humans like explosions and drilling. Rock falls are also caused by the low resistance of the cliff. This can be caused by its formation or variations with time. Initial weaknesses include: the geological composition, the resistance of the materials, how are the different materials aligned, how they interact with water, presence of faults and other discontinuities, form of the slope and how steep it is. Changes with time are linked to weather conditions. Rain is one of the main factors of erosion and reacts with many different minerals causing them to erode.

Even though all of those factors are important causes, three of them are usually described as the main triggering factors: rain, freeze-thaw cycles and earthquakes. This is true for primary and secondary rockfalls, but it is conceived that secondary rockfalls are triggered by a wider range of events (Luckman 2013).

Protective elements

Rock falls are destructive events that cause damages and casualties. Diminishing the impact caused by this phenomena can be done by avoiding areas with high risk, by eliminating the hazard and by constructing protective structures (Wyllie 2014).

Blocks fall from bedrocks with steep slopes. Knowing that there is a possibility of masses of rock falling from the bedrock, avoiding building roads and houses beneath it would avoid negative consequences. Paths that are already built can be redirected, protected or temporarily closed when the risk is higher (or a combination of all these solutions).

Detailed analysis of cliffs may present volumes which are unstable and with high probability of collapse. In order to avoid damage it can be purged. This way they can be removed in a controlled manner and will not cause any negative impact. It is important to note that, by removing unstable blocks the bedrock behind it may become unstable.

Protective structures are separated into two groups, active and passive. Active structures will hold blocks on the cliff and prevent them from falling. Passive structures will stop or deviate the rocks that fell from the cliff in order to prevent them from reaching the stakes.

Active protective structures are implemented directly on the bedrock. The most common is the implementation of a steel net that is attached on the bedrock with soil nails [Figure 1.a]. Soil nails can be used without the net in order to hold single blocs.

Passive protections are generally walls or ditches to stop the propagation of the rocks. This wall can be a steel net Figure 1.b, similar to the one used for active protection, but can also be an embankment which could be made of gabions, reinforced soil or a mixture of materials Figure 2.a. A ditch can also be excavated in order to stop the blocs on their path Figure 2.b.

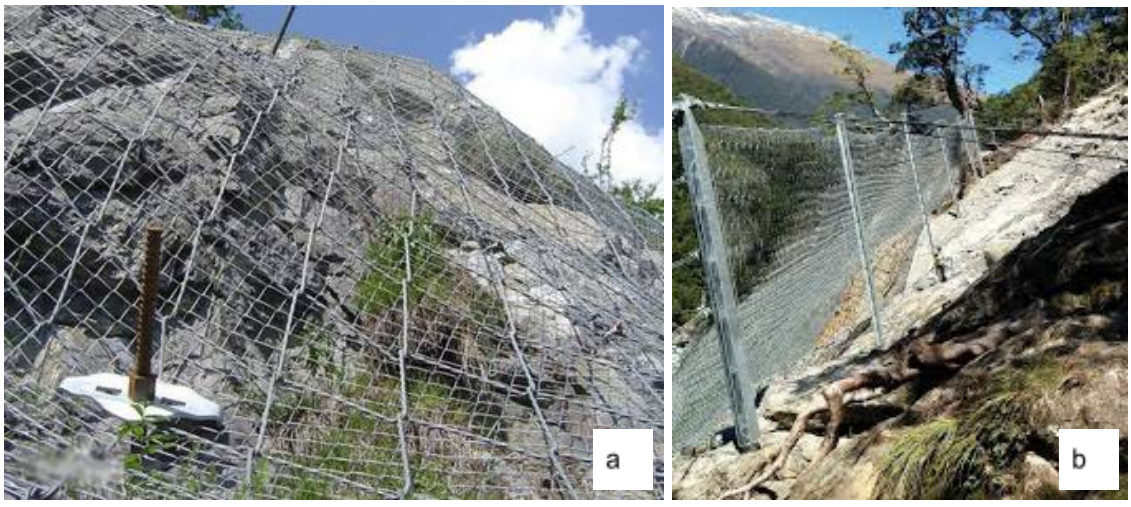

Figure 1 : Steel net protective element (a) fixed on the bedrock by nails and (b) located at the base of the bedrock

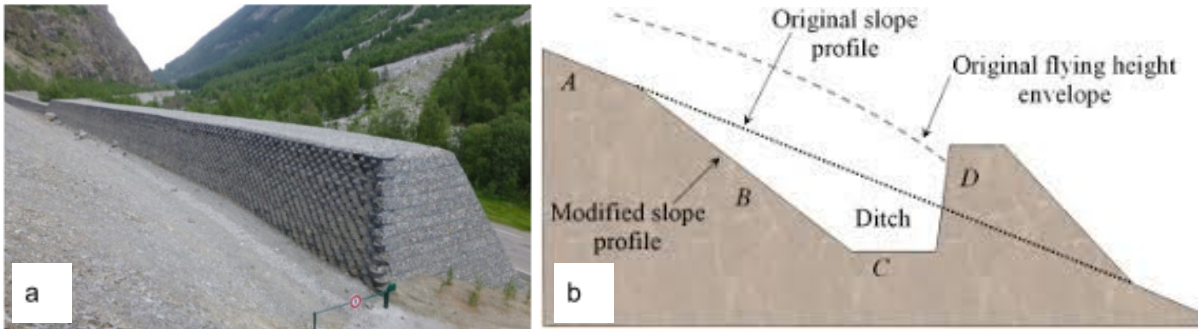

Figure 2: Passive protective structures at the base of a bedrock. (a) Embankment made of gabions (b) ditch caved and embankment raised with material of the ditch.

1.2. Data analysis

Data analysis is observing a dataset and using it to understand the behaviour of a cliff. It helps to find patterns and trends, which may be used to predict future events such as rock falls.

This study will be using Artificial Intelligence (AI) for predictions. AI is a section of computer sciences that attempts to create machines capable of performing tasks that require human intelligence. This term is evolving with time and scientists aim to create more and more complex systems. One of the many challenges is providing machines with the capabilities of having creativity and producing things like screen plays.

One aspect that has been deeply explored is pattern recognition. The capability of looking at an information, receiving new information and fitting this new information inside what is already known. Many different types of models exist in order to do this. Three of those methods, nearest neighbors, decision tree and neural networks (NNs), will be explored in this work.

The branch of machine learning involves pattern recognition, but it differentiates itself by including the concept of learning. It works with the idea of making predictions, comparing their results with the actual values and improving upon them. This is an interactive model. An example of machine learning are NNs and some types of decision trees which will be elaborated on section 3.1.3.

1.3. Aim of the work

As technology advances statistical methods grow in performance, speed and applicability. Classical methods have already been used many times to better understand rock falls and the conditions which trigger it. This study aims to:

1. Better understand the causes of rock falls. By creating predictive models based on past events it is desired to better understand which are the most important factors and at what intensities the blocks will fall.

2. Compare the performance of AI with other methods and its capabilities for predicting rock falls. This includes the precision of predictions, the applicability of the models and the time to produce the models.

This project was a collaboration between different teams with those objectives. The groups involved were: CEREMA, GEOLITHE, BRGM, Université Savoie Mont Blanc. The first 3 groups mentioned had expertise on rock falls while the other was expert in artificial intelligence and data analysis. Hermann Courteille (USMB) [Courteille, 2021] produced the NN models and did a qualitative analysis of the data.

I processed the initial data combining the data sets and eliminating superfluous information. I also analysed the cliff in order to describe it in a way that could be inserted on a data analysis and possibly a predictive model, additionally I created alternative predictive machine learning models, using decision trees and nearest neighbors, to compare with the deep leraning models created by Hermann Courteille.

2. Study case

A choice of the study case needed to be made. A well documented site with years of data, and a good number of events is required. Initially 6 different locations were proposed: RN1 (la Réunion), Val d'Arly (between la Savoie and la Haute-Savoie), la Saulcette (Savoie), cliff from Saint-Eynard (Isère), Mont Faron (Var) and la Rochaille (Alpes de Hautes-Provence). The two first sites collect the data about the events that reached the road while the others monitor the cliff and observe when blocks break off.

The RN1 (Route National 1) from the Réunion was finally chosen because of the numerous well documented rock falls that reach the road and the longest period of observation. It has also been studied previously by CEREMA and Geolithe and experts were able to help with the understanding of the cliff behavior. One of the difficulties of applying traditional methods to rock fall prediction is the rarity of the events. The RN1 is the location with the greatest intensity of rock falls, and this will provide a richer data set.

2.1. La Réunion

La Réunion is a French island on the south east of Africa, as shown on Figure 3. This French department has a population of 850 000 and a surface area of 2 512 km² ("Découvrir La Réunion" 2020). It has a tropical climate with higher temperatures close to the coast and lower temperatures inland as the altitude increases Figure 4. On the coastal area mean temperatures vary from 30° in summer to 24° in winter. Temperatures, in this region, are always superior to freezing temperatures which indicates that freeze-thaw cycles will not be taken into account as a potential triggering factor.

Rain is not equally distributed along the island as displayed by figure 5. The West has much higher volumes of rain. This is due to the volcanoes that block the clouds from continuing to the East. The topography of the island is strongly linked to volcanic activity as it was formed

by two volcanoes: le Piton des Neiges et la Fournaise. This means that the island resembles two cones with the points being the peaks of the volcanoes.

The Piton des Neiges is the oldest and was active 2 million years ago. It created a first lithological structure which is composed of alternating layers of basalt, crystallized magma, and scories, sedimentary rock. Both of those layers have thicknesses below 3 meters. Dykes are also abundantly present on this structure due to volcanic activity. This structure is called the inferieur structure.

The volcano became dormant and a stage of erosion started changing the topography of the island. Sediments also started occupying the bed of the valleys that formed a new geological structure. This is called the intermediary structure.

About 1 million years ago the Fournaise appeared and a new structure, similar to the first, started to appear. The main difference between both volcanoes is that, as many valleys were present, the magma was more concentrated and early layers reached thicknesses 10 fold more important than the original. This corresponds to the superior structure that covers the majority of the island.

With time, a new process of erosion molded the region. As the inferieur structure is older it suffered through more weathering due to the sea and the rain and erosion favored those spaces. All of those steps are illustrated in Figure 6.

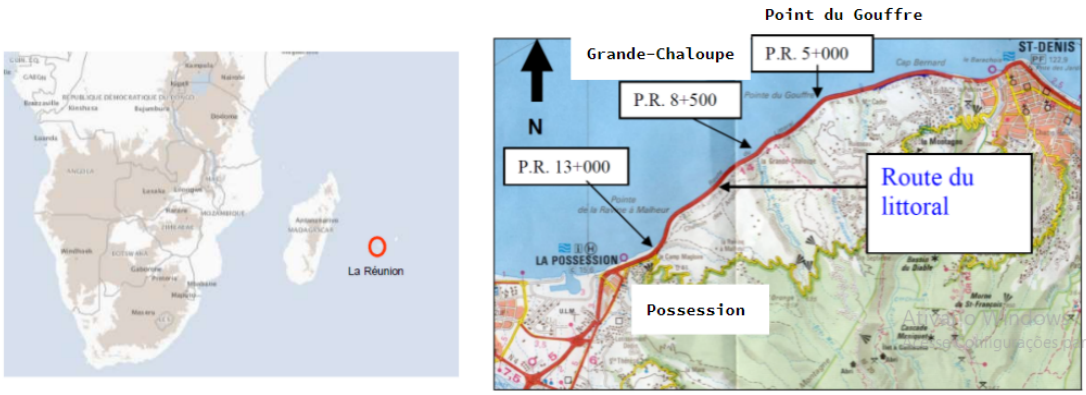

Figure 3: Map showing the road

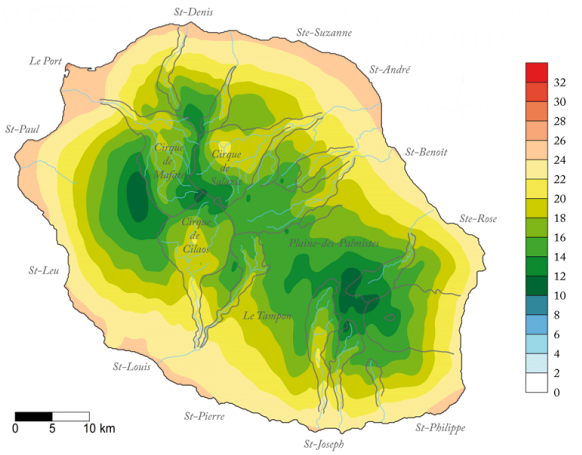

Figure 4: Mean annual temperatures of the island la Réunion

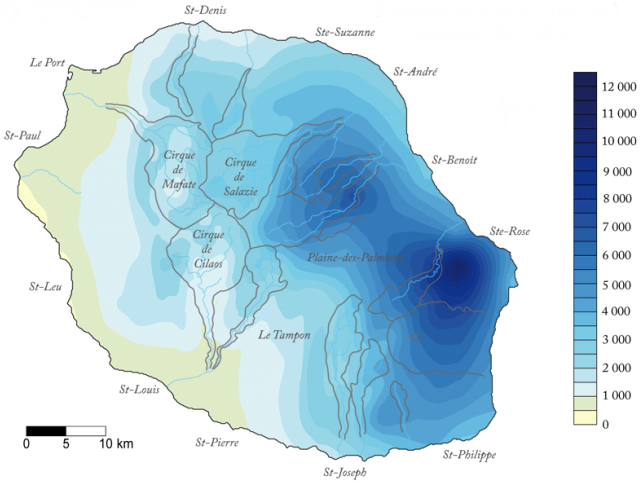

Figure 5: accumulated annual rain of la Réunion.

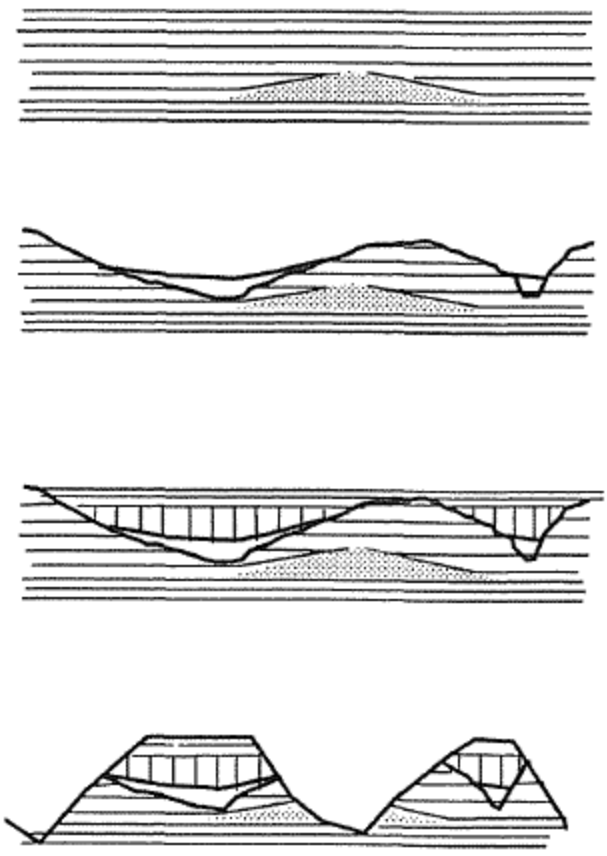

Figure 6: Lithological evolution of la Réunion

2.2. Temporal data

Temporal data include the meteorological information (daily temperature and rain), and also the rock falls and their description (date, location along the road and hitted lane, approximate volume). The data sets contained the rainfall accumulated in millimeters every 24h on 3 different points along the road. Meteorological information was also acquired from stations on the 2 cities which are connected by the road, Le Port and Saint-Denis. Data from those stations are available on the web site https://www.historique-meteo.net/. They provide daily values of the maximum and minimum temperature and the daily rain. The rock falls are listed in a table with all the information regarding each event as described in the previous section. Temporal analysis will consider daily intervals as it is the shortest period of time we have all the information and operationally is what makes more sense as the rules for closing the road were created taking into account daily intervals.

There were a total of 5 different stations measuring values of the daily rain, 3 along the road and 2 on the cities. All 5 values are extremely correlated with a value superior to 0.85 for all cross correlations as expected. Previous studies used the maximum value between the 3 stations on the side of the road. This is done as the maximum value is the one that will probably cause a

rock fall. Only the 3 stations along the road are taken into account as they are the closest to the road.

The temperature was only available from 2009 onwards. The correlation between both stations was 0.97 for the max and min temperature so the values of Saint-Denis were used.

Rock falls were on a table with a list of events with the description of each event. For the majority of the models the objective was to predict if there was an event or not, so a table of days was formed with the data and a 1 was indicated for days with a rockfall event or a 0 if there wasn't.

3. Preprocessing, data description and previous work

The analysis will treat the problem temporally, but a spatial understanding could be useful for a better understanding of the behaviour of the cliff. The cliff did not change significantly during the studied period. The only changes that will be considered were the structures built in 2008 to protect the road as they had a significant influence on the probability of blocks hitting the road. This leads us to study the periods before and after the construction as 2 different data sets.

The data can be studied in 3 different formats: temporally, spatially or both. Rock falls occur on a certain time and a certain location and statistical models can help to predict both of them. This study will only study the rock falls in function of the time. Being able to predict the date of the event permits its temporary closure in days the risk is higher in order to avoid accidents. A model that combines spatial and temporal elements would not be ideal as the data set contains few events and adding dimensions would increase this problem.

Even though the predictive models will be used for the predictive models a spatial study was made to better understand the cliff. Different descriptive values were calculated to try to understand which characteristics of this bedrock increased the risk of blocs detaching and reaching the road.

The time set for each entry is on the order of the day. This permits an easy comparison with the operational code as the closure of the road is dictated by the behavior of the rain accumulated in a period of 24h. Grouping the data by month would facilitate finding tendencies but would make it harder to apply the model for the operation of the road.

In this study, we used existing data (1- 4) and had to calculate other (5 -7) :

1. Information about the rock falls that occurred between 1999 and 2018. For each event the table gave the date and time of the event, the point it fell along the road (PR), the lanes affected, the mass that fell and the consequence of the event.
2. Values of the daily rain from 2000-2020 collected on three different stations (Figure 3)
3. Location of the protective elements along the road before and after the constructions of 2008.
4. Geological characteristics of the bedrock along the road

5. Standardized area of a profile of the cliff every 10 m
6. Standardized Concavity Index (SCI) of a profile of the cliff every 10 m
7. Topographic Wetness Index (TWI) of the top of the cliff every 10 m

3.1. Previous works

Many works have studied the temporal occurrence of rock falls using statistical methods [Damato 2015, Rosa 2012, Westoby 2020] and some of those studied specifically the occurrence of rock falls at the RN1 (Réunion Island) as it is a site with many well documented incidents. Two studies of rock falls at the RN1 are highlighted hereafter: an operational study and a statistical study.

3.1.1. Operational rules for road RN1 maintenance

The most relevant studies of rock falls temporal occurrences at RN1 were made in order to create rules for closing the road in order to avoid accidents [Batista, 2016]. The rules are updated with time as better performing rules are created and as conditions change, notably the installation of protective structures that help diminish the number of blocks that hit the road.

Each version of the code is composed of 2 rules. The first indicates a maximum amount of rain in the former 24 h and the second indicates the maximum amount of daily rain in the former 48/72 h. Two rules will be highlighted in this work, the R7 and the Rps4b.

The R7 was developed through a study in 2003 and started being applied in 2004. It was based on data acquired between 1998 and 2002 and was developed after analysing a study made by CEREMA. The R7 dictates that the road will be closed if:

- The daily rain in the former 24h is superior to 15 mm
- The daily rain in the former 72h is superior to 30 mm

This rule was able to reduce by 30% the duration when the road was closed and maintained the same intensity of 0.151 blocks falling per day on the period that the road is open leading to the road being closed during 11% of the time.

In 2009 a new suggestion was presented. As the protective structures were built, and dimensioned to reduce the number of blocks that hit the road by almost 80%, less restrictive rules could be used in order to have longer periods of allowed traffic on the road whilst maintaining a similar level of risk for drivers. The Rps4b dictates the road will be closed if:

- The daily rain in the last 24h is superior to 30 mm
- The daily rain in the last 48h is superior to 50 mm

This permits the road to be closed during only 5,8% of the year, about half of the time given by R7. Between 2010 and 2015, the intensity of rockfalls per day while the route was open was inferior to 0,015 being less than 10 times the historically accepted risk. This work provides a basis of what can be done on this case study with two expert rules.

3.1.2. Statistical Study of rock falls on the RN1

Many different factors are known to cause rock falls. The two main ones are the pressure caused by water infiltrating and freeze-thaw cycles. A study made by Delonca [Delonca, 2014] analysed the correlation between the different meteorological factors and rock falls trying to determine which factors are linked to rock falls and which one is the most important.

The general difficulty of studying rock falls is the scarcity of data, making it difficult to find important correlations between the event and the meteorological condition. A new method is proposed in order to find correlation when events are rare. The proposed methodology weights the number of rockfalls by the probability of occurrence of the studied triggering factor and this is done by grouping the days based on the descriptive feature, calculating the probability of having a day with an event inside each grouping. The final step is a linear regression analysis which is performed to compare the magnitude of the triggering factor and the probability of occurrence for each value. The grouping is arbitrary but two rules should be followed, the number of days in each grouping should be superior to 5 and each grouping should contain at least one event.

This analysis was made for the rain, the accumulated rain, the maximum temperature, the minimum temperature and the difference between both. For each factor studied it would also be compared with the value of the previous days as the triggering factor may take some time to detach the block. It is important to note that the correlation between all of those factors and the presence of an event was calculated, but only the correlation between the rain and the events was seen as sufficient. Figure 7 shows the values of the cross-correlation coefficient between the daily number of rockfalls and the daily rainfall at RN1 (La Réunion) for several delays and how the most significant coefficient is obtained for the rain from the previous day.

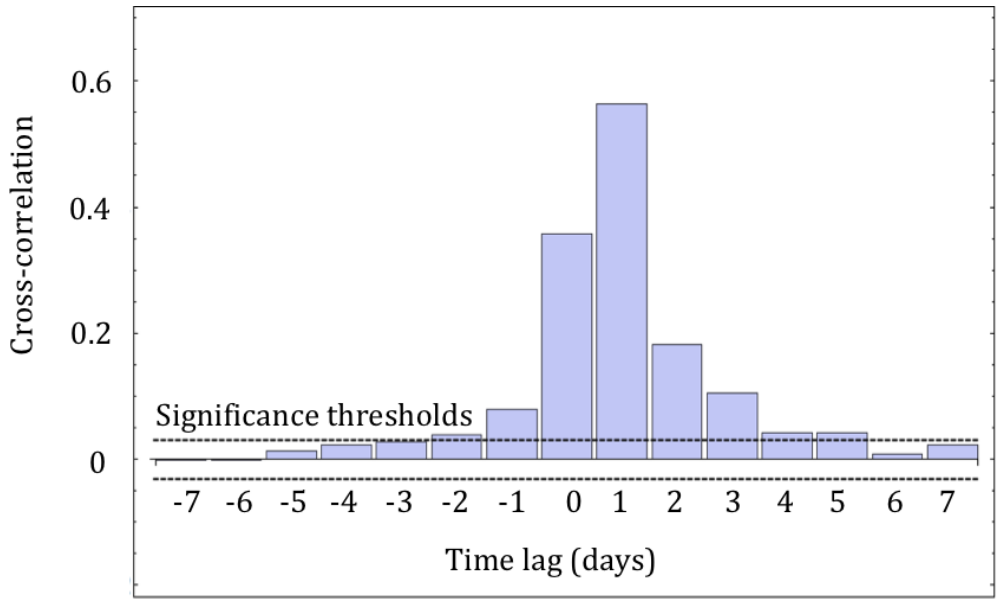

Figure 7: Cross-correlation coefficient between the daily number of rockfalls and the daily rainfall at RN1 (La Réunion) for several delays. The significance threshold, equal to 0.031, is represented by the dashed lines. Source: Delonca et al. (2014)

The proposed model also found the correlation with the rain and, as the classical method, indicated that the previous day was the most directly related to rock falls, but with the proposed method other correlations appear. The table ? shows the maximum correlation for each feature and it is interesting to see how the temperature, specially the minimum temperature, is related with the rock falls almost as much as the daily precipitation. In most cases this could be due to freeze thaw cycles, but it is not likely as temperatures in La Réunion do not descend below zero. An explanation could be the correlation between the temperature and rain as both are seasonal events.

	Réunion
Daily precipitation (P)	For $D_1 - R^2 = 0.70$ p value $= 10^{-9}$ Correlation coefficient $= 0.12$
Cumulative daily precipitation (P_c)	For $D_1 - R^2 = 0.74$ p value $= 10^{-13}$ Correlation coefficient $= 0.10$
Daily minimum temperature (T_{min})	For $D_1 - R^2 = 0.69$ p value $= 10^{-6}$ Correlation coefficient $= 0.5$
Daily maximum temperature (T_{max})	For $D_1 - R^2 = 0.60$ p value $= 10^{-5}$ Correlation coefficient $= 0.8$
Daily temperature amplitude (T_{amp})	No correlation

Table 1 Correlations between the chosen meteorological factors and the daily number of rockfalls; results obtained with the proposed method on the real databases. Only the maximum correlations are presented here.

Source: Delonca et al. (2014)

3.2. First analysis of rockfall events

Before creating models it is important to understand the behaviour of the cliff based on the data. Basic statistics and visual representations can help with understanding the relying dynamics of events and limitations of results.

Observing the times series of daily number of events(Figure 8) the difference before and after 2008 is easy to see.

It is also possible to observe a gap in 2004 as part of the data seems to be missing for this year. There are no remarks of any important changes during this year that would cause such a

change in the behaviour of the cliff and previous works used data from this year with no special treatment. It is believed the data received is incomplete so the period with no data will not be considered. Thesecond part after 2007 also sees a significant reduction in the number of events. This can be associated with the beginning of construction of protective structures and this change will also be considered.

Besides, the number of events also follows a seasonal nature. The values peak between December and January, the same moment as the rain peaks as observed in Figure 8.

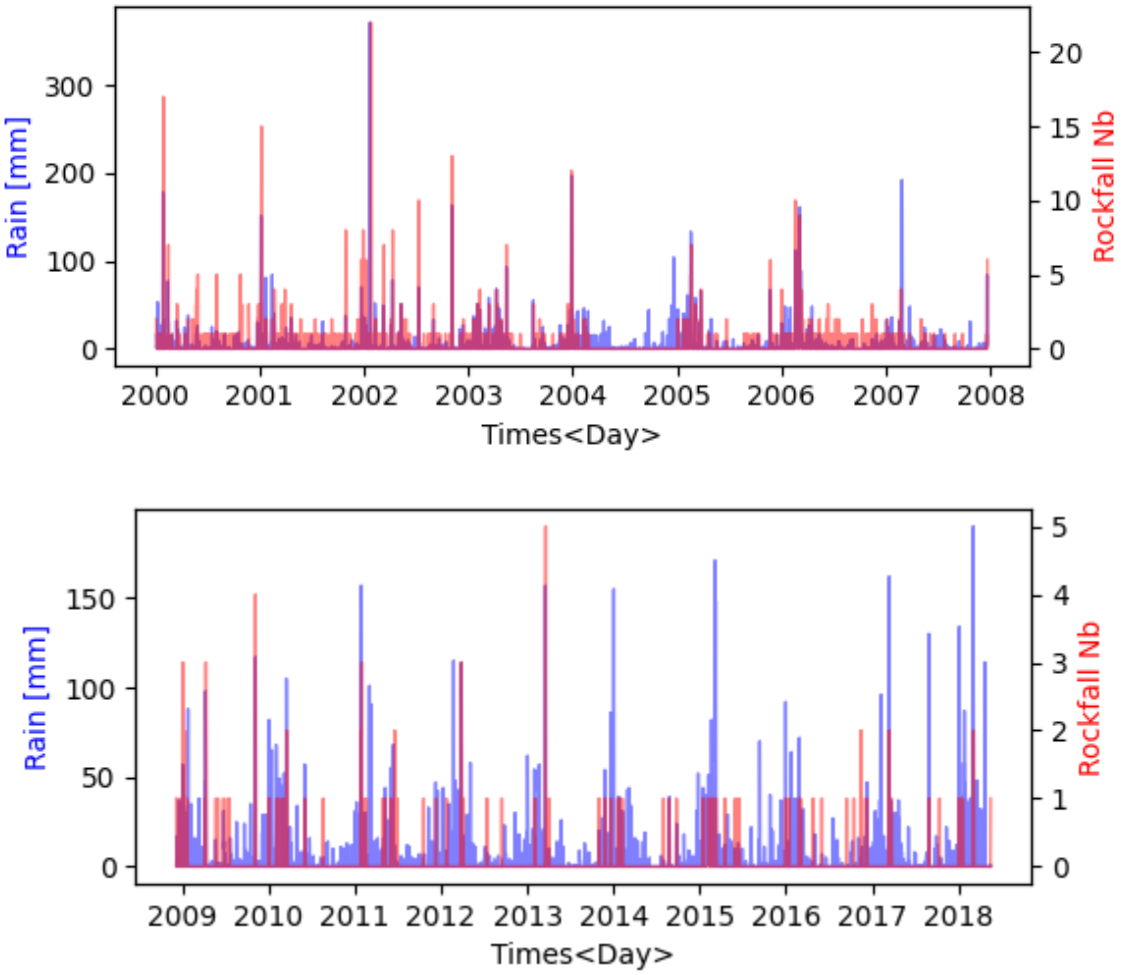

Figure 8: Evolution of the amount of rain and the number of rockfall in fonction of time

A representation of the number of events along the road before (FIG à ajouter??) and after the construction of protective structures allows the visualisation of the difference of the amount of events, but also their spatial distribution. Observing only the events that occurred before 2008, some zones are noticed with fewer events. The most noticeable is the region before pr 3+500. This can simply be explained by the fact that this region already had protected elements installed (Figure 9).

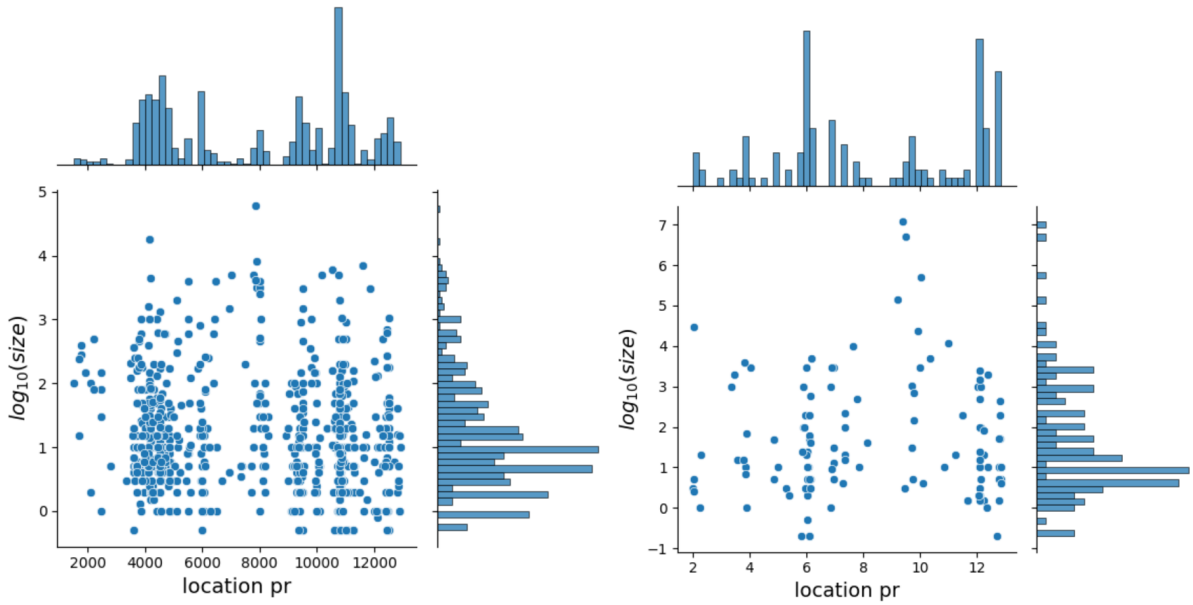

Figure 9: mass of those rock falls in fonction of the location and histogram representing their frequencies.

3.3. PCA analysis of rockfall events

Présenter données sur lesquelles la PCA est faite? Pourquoi (P0, P1, P2, P3, P4) ? Pourquoi réduire la dimension dans ce cas précis?

Principal Component Analysis is a dimension reduction method that creates new features based on the existing ones. It creates the same amount of features as previously existed and by combining the different features is capable of creating new ones that are more adapted to explaining the data in fewer dimensions. The objective is being able to plot and create models on lower dimensions.

With the new variables it is important to observe how much of the variance each feature is capable of explaining and consequently how effective the reduction of dimensions is. For the data before and after the implementation of protective structures the figure 10 (a) and (b) was created based on the rain of the previous 5 days (P0, P1, P2, P3, P4). Both bar graphs are extremely similar, with all values being superior to 10% indicating that ideally all features should be considered. The first analysis that should be made is with the two first components. Together they explain 60% of the variance.

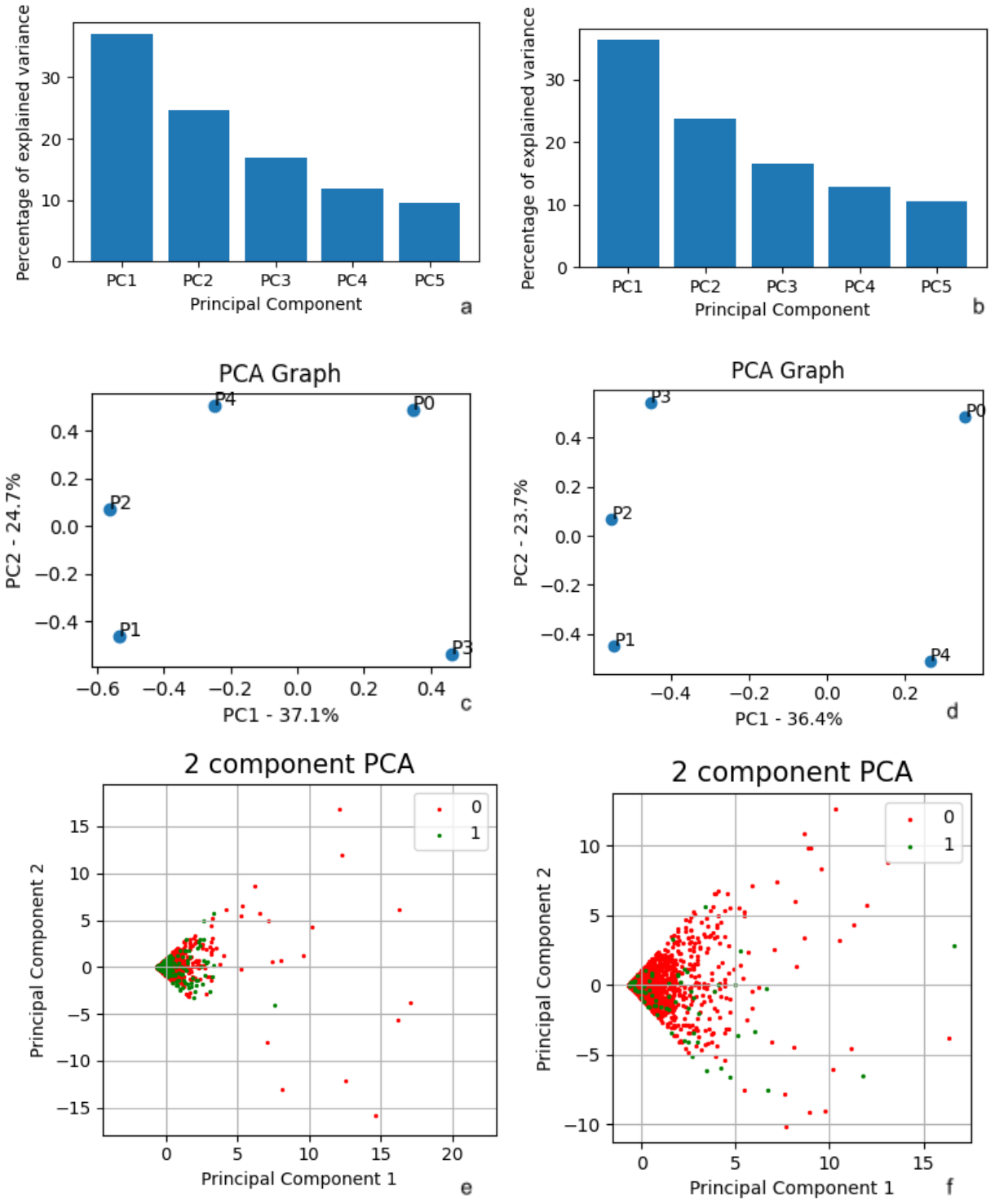

Figure 10: PCA images, percentage of explained variance by different variables for (a) the data before constructions and (b) data after construction of protective elements. Representation of importance of different variables for the principal components 1 and 2 for (c) the data before constructions and (d) data after construction of protective elements. Distribution of the data points along the principal components 1 and 2 for (e) the data before constructions and (f) data after construction of protective elements.

The PCA Graph represents how much of each variable is explained by each component. Figure 10 (c) and (d) shows how PC1 and PC2 are formed. We observe that all 5 features seem spread on the 4 corners. No two features are clustered together.

Plotting the data on the two principal components with the events being colored it is possible to look for clustering of events and try to identify any clustering. Figure 10 (e) and (f) shows a concentration of points close to the origin. This makes sense as the majority of the original data is also clustered as the majority of the days there is no rain. The main objective of this exercise was to try to identify a visual separation between the days with rock falls and the others. Neither of the graphs provides a clear separation that could be exploited.

3.4. Spatial data

Having the topography, the geology and the information about the protective elements installed along the cliff values were assigned, along the road, to represent the geomorphological characteristics of the cliff. This spatial study aims to improve the understanding of what characteristics of the cliff are linked to the blocks reaching the road.

The protective elements were simply described as present or not present for each point of the road. Various protective structures are used for different parts of the road and they can also have different dimensions depending on the demand, but those were not considered. The final separation before and after the constructions are shown in the figure 11.

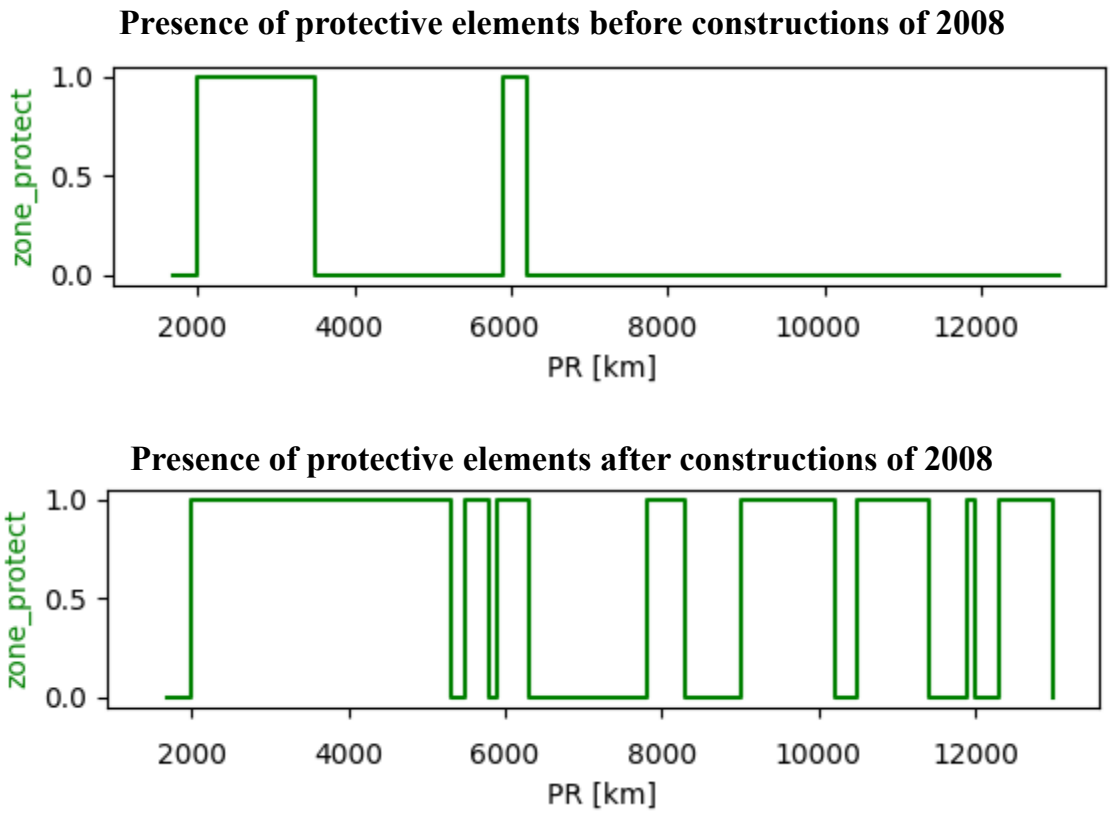

Figure 11: protection zone (0 no protection/ 1 with protection)

Taking into account the geology was done by defining norms to separate the cliff into different zones. The criteria was chosen in order to find zones which have similar geological structures. The cliff is composed of three different geological structures: inferieur, intermediary and superieur, as described in section 2.1. The inferieur and superieur structures are present throughout the cliff. The intermediary is more ponctuel. The chosen criteria is as follows:

- Zone 0 : predominance of the inferieur structure
- Zone 1 : predominance of the superieur structure
- Zone 2 : presence of the intermediary structure

The inferior and superior structures are similar, but have different behaviors, as the inferieur is more compact, leading to different resistances. The présence of the intermediary structure leads to irregularities that could cause rock falls. The separation was made by analysing the different images taken of the cliff. The final separation is found in Figure 12.

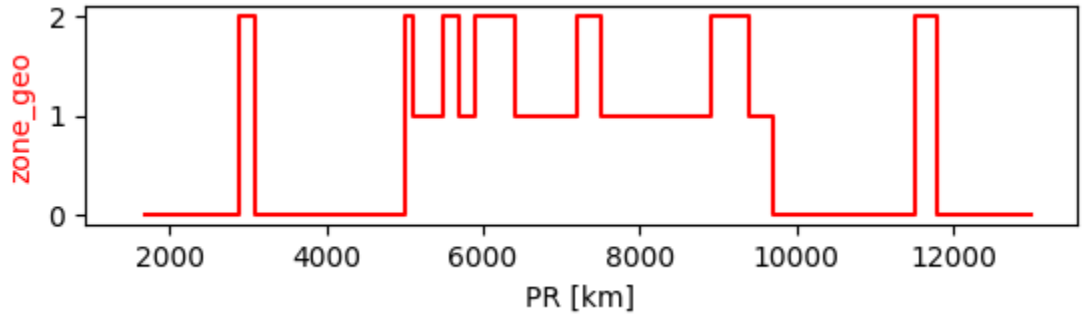

Figure 12: Geological zoning

Two topographic indexes, the standard area and the Standard Concavity Index (SCI), are calculated from profiles of the cliff. The standard area is a value that normalises the vertical and horizontal dimensions, as shown in Figure 13, and calculates the area under the cliff []. Smaller values tend to indicate a steeper cliff while larger values tend to indicate a smoother cliff.

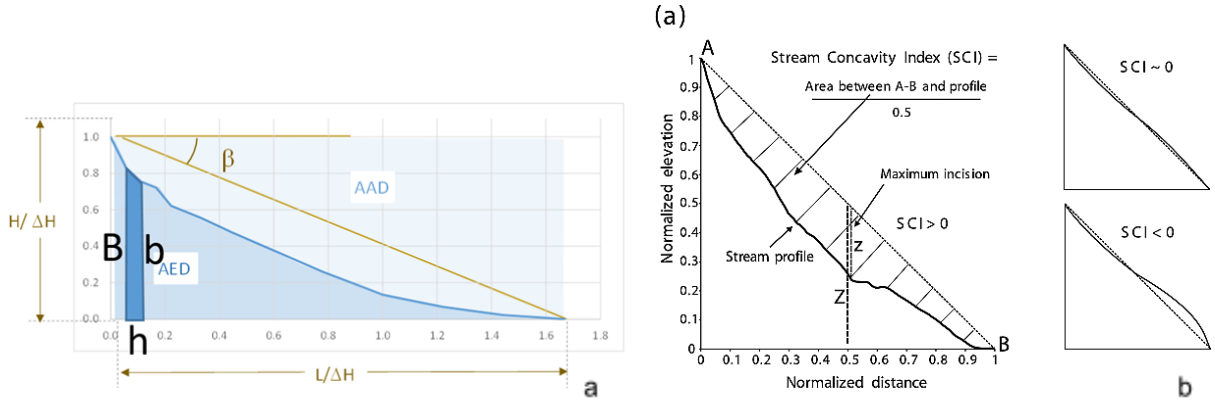

Figure 13: normalisation of dimensions for the calculation of the (a) standard area index and (b) SCI

The SCI is similar to the standard area index as both involve normalising the dimensions and calculating an area, but in this case the normalisation is different and the area calculated is between the cliff and the straight diagonal line that joins both extremities of the cliff []. This is illustrated in Figure 14. Comparing both formulas it is observed that profiles with a small standardized area will have a large SCI value and vice versa.

Profiles perpendicular to the road were extracted from the DTM map every 10 m along the road. A total of 1122 profiles were extracted and analysed. This was done by transferring the DTM map to Cloud Compare and extracting sections of the cliff over the trace of the road seen in Figure14.

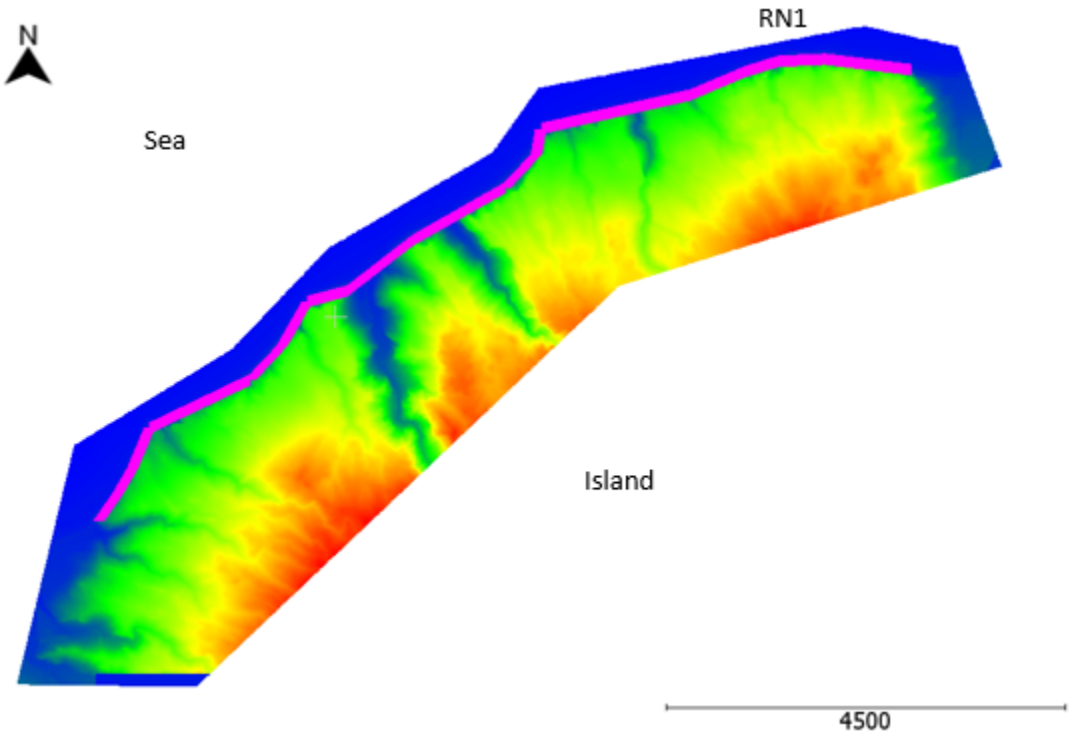

Figure 14: RN1 trace used for the extraction of the topographical profiles

In order to calculate both indexes it is necessary to estimate the highest point the block may depart from the cliff and arrive at the road. This was done with a specific developed code. The program looked for the extremity of the road closest to the cliff as the arrival point of the blocks and for the point from which blocks fell. The second one is complex as different people have different interpretations as one segment can have more than one cliff. The last point with a strong slope, but without any large segment in front of it with a weak slope, would be considered the departure point. Some examples of the profiles are shown in Figure 15.

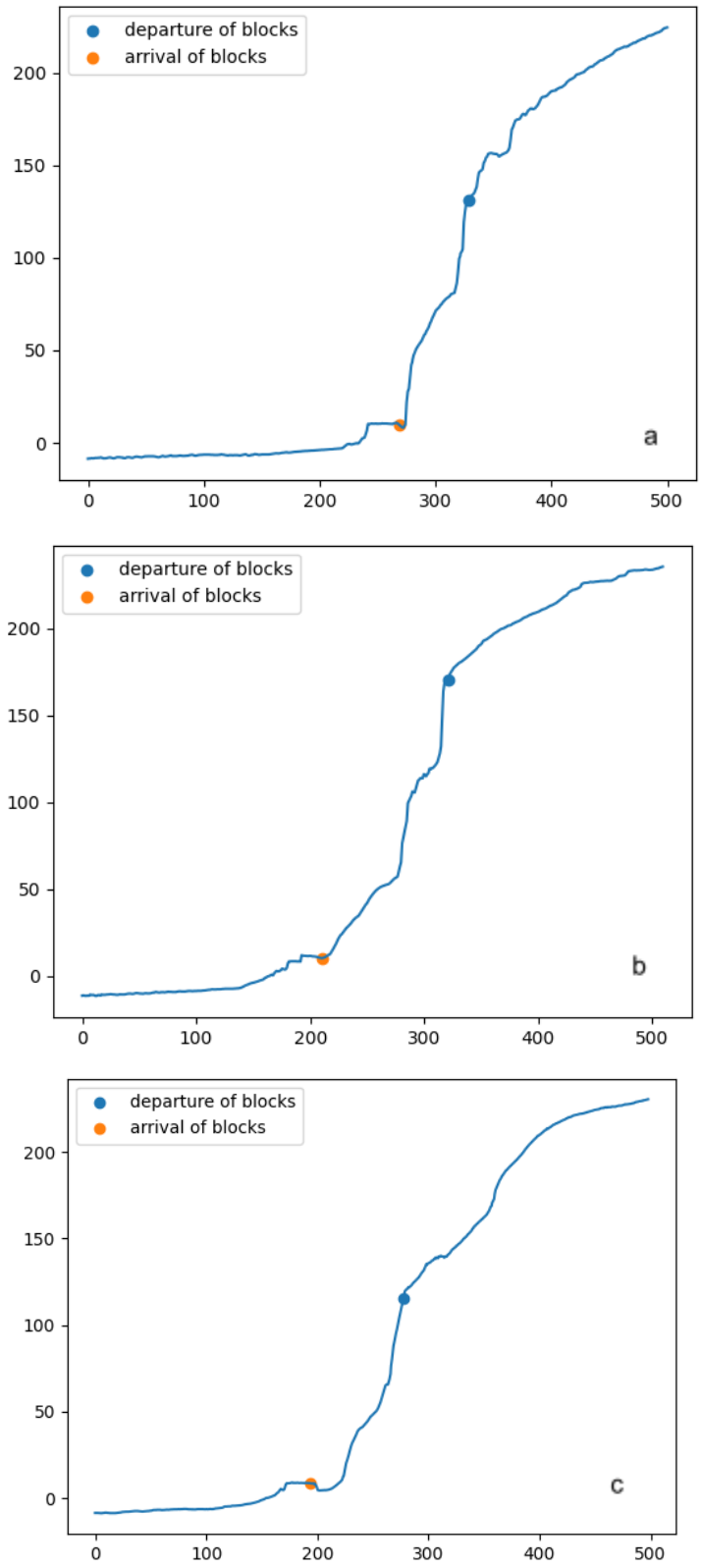

Figure15: 3 examples of profiles along the road with the estimated departure point and arrival point of the detached blocks

The code calculates the slope for each point and finds the arrival point by searching for the last point with a weak slope, with a magnitude below 0,1, and also with a vertical value between 8 and 12,5. There are 2 criterias for the departure points. The first is that it has to be steep, with a gradient stronger than 2, and also it cannot have a long distance with a mild slope, gradient below 0,5.

The coordinates for the departure points were exported to QGIS where they were combined with the contour lines in order to better examine the ensemble of points found by the program. The result is shown in Figure 16. The points which are in the middle of the sea are when the program was not capable of finding a departure zone. This phenomenon is present on the exit of both rivers. Generally the program gave consistent results that are seen by the continuity of points representing the departure points. Two zones were selected to show the discontinuities that are present. On both zones we observe the presence of many cliffs over one another and all of them are possible sources of rock falls.

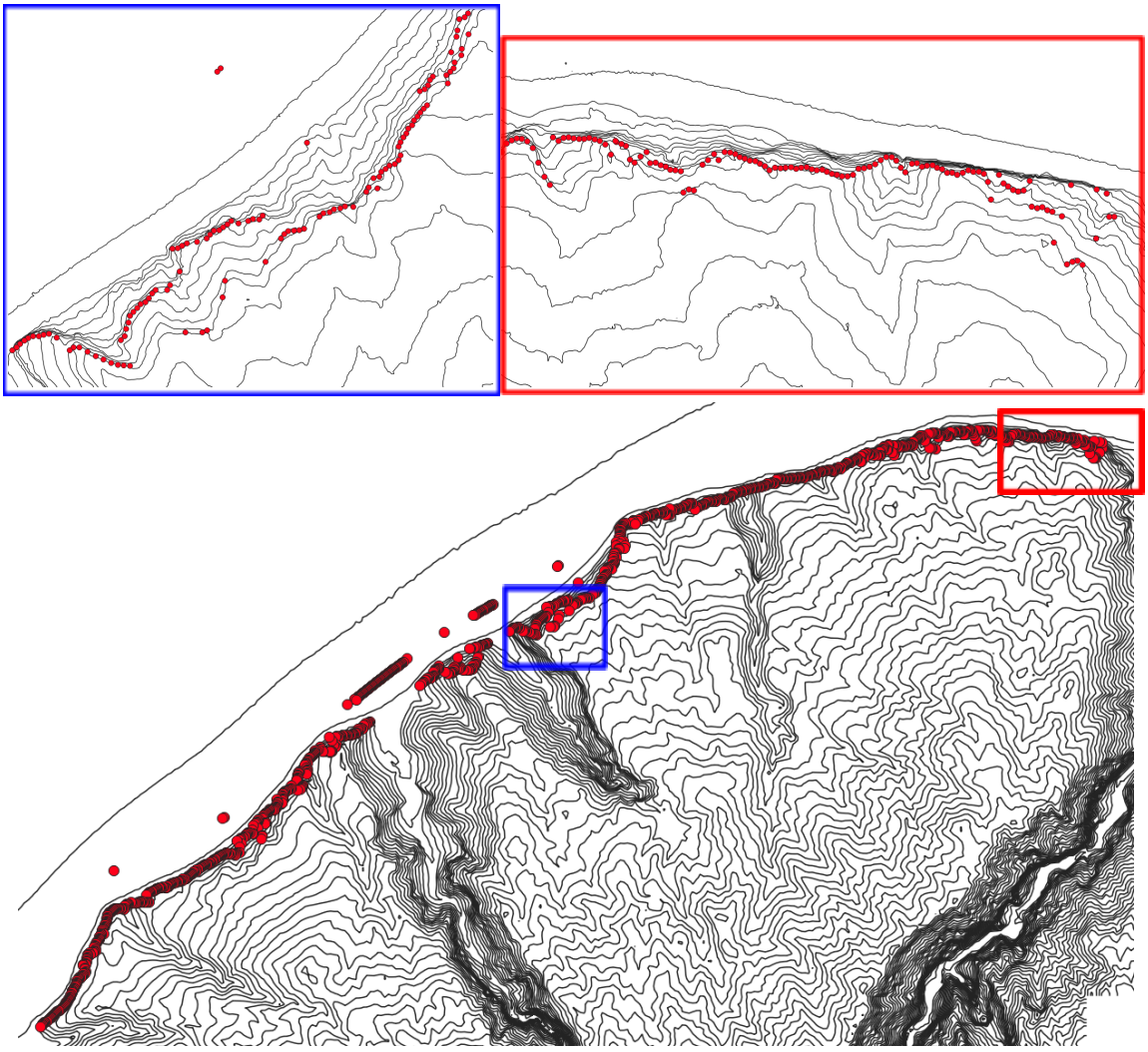

Figure 16 : combination of the departure points found by the program and the contour lines. Two zooms to demonstrate more problematic regions, just before the first river and on the beginning of the road

Based on the calculated points a manual line was traced in order to have more spatially continuous points and therefore, more continuous values should be expected for the indices calculated for the topographic and hydrological description. The final points used still had some discontinuities, but less of them and, moreover, there aren't outliers as seen in the image above (Figure 16). With those points, the equations shown on Figure 13 are applied and Figure 17 can be produced. As noted previously, both values seem mirrored, higher values of Standardized area leads to smaller values of SCI.

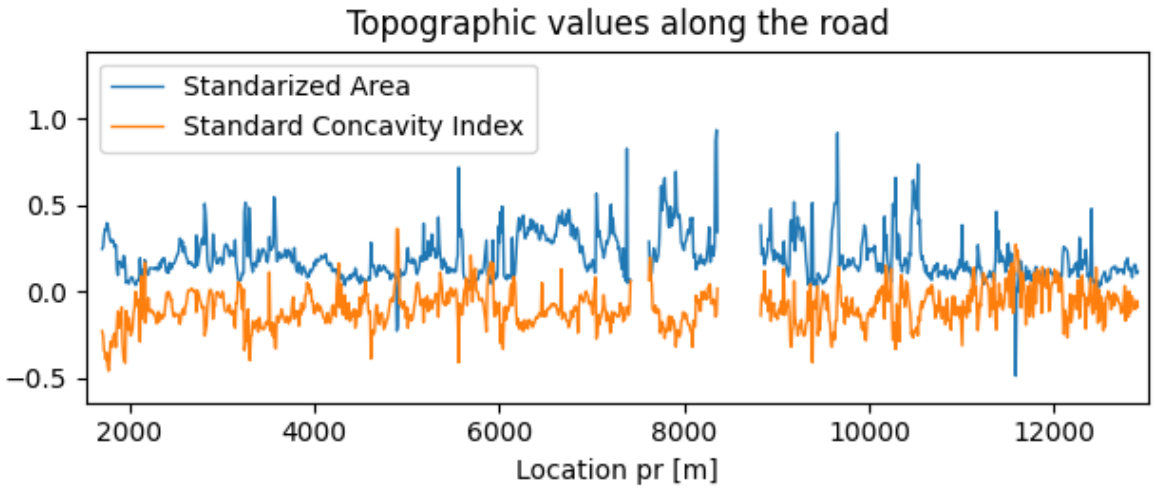

Figure 17 : Variation of the topographic indices along the road.

The fifth and last value calculated to describe the cliff was the Topographic Wetness Index (TWI). It is calculated with the topographic map and indicates the concentration of water of a point according to the direction of runoff. Q-GIS was used to calculate the value for every 10 m square inside the watershed []. In order to have one value for the different points it was decided to use the value for the point from which blocs depart as generally cumulated waters behind the cliff infiltrates and causes pressures that cause blocs to detach. The final result can be seen in Figure 18.

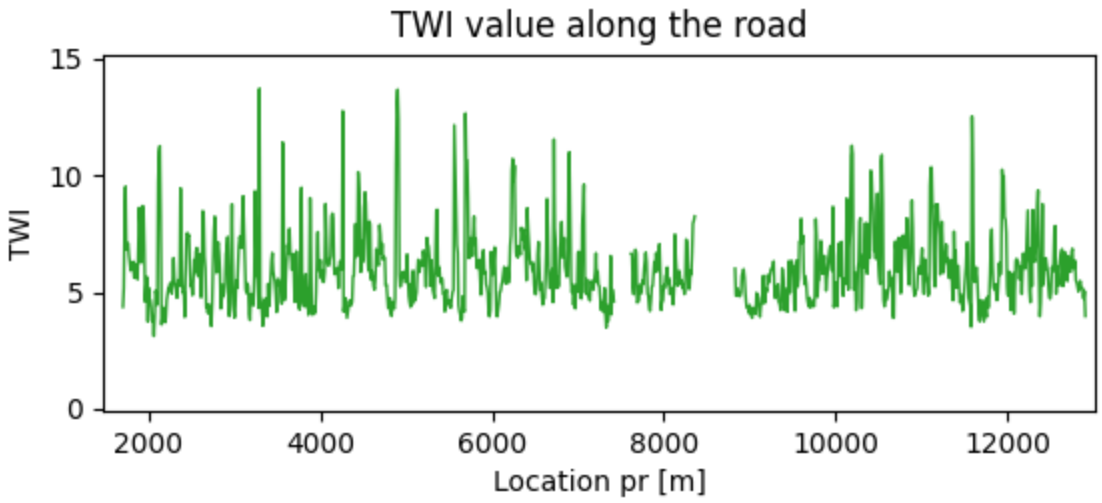

Figure 18:Variation of the TWI along the road.

This spatial study aimed to better understand the characteristics of the cliff which were related to the rock falls. Different values were plotted in function of the pr. The clearest correlation is that in zones with protection there are less rock falls. Observing Figure 9 and 11 it is possible to observe the difference of the amount of rock falls before and after the addition of protective elements along the road. Analysing the data before the constructions of 2008 the data already showed that the initial section (1+700 - 3+500) the number of events was visibly lower.

The lithological structures do not seem to be strongly related with the distribution of events, but a deeper study could show some correlation. On The other hand, there are some correlations between the topography and the number of events. For values of the standardized area (Figure 17) closer to 0 (example 4+000 - 6+000) more rock falls tend to fall (Figure9). The region with a higher standardized area close to 0,5 (example 6+200 - 7+500) has fewer rock falls.

This spatial analysis shows how the number of rock falls are related to a number of different characteristics of the bedrock. A spatial analysis could be done using those features as the input in order to predict the location a rock will fall.

4. Methodology used for prediction models

AI was used to create models that will predict the probability of rockfall that reach road. if rock falls will reach the road based on the data. There are various ways to create prediction models and various criterias to measure their capacities. The objective is, given a certain amount of information, what will happen in another case. All the data could be given to the program as an input, but then it would only be possible to evaluate the model based on the information it used to produce the model. One solution is separating the data into two parts, training data and

test data. The first is used to train the model based and the second to test the capabilities of the model. In this section, different types of models and the criteria will be explained.

4.1. Model

Artificial intelligence contains a multitude of different types of models. Some are more complex than others, but they are all adapted to different types of problems. Different types of models will be created and compared between each other and the rules created to control the flow of the road.

This problem could be solved as a regression, predicting the mass that fell each day, or as a classification problem, solving simply for the presence of an event or even bining the number of events. The simplest is creating classification models as it is closer to what is done operationally and also the difficulty of not having any event is even more drastic.

Three types of classification models were chosen: nearest neighbors, decision tree and Neural Network. For each one, various tunings are needed in order to optimise the model. Python's library scikit learn and tensorflow were used to generate all predictive models.

4.1.1. Nearest Neighbors

This is the simplest of the methods but it is also quite effective. The idea is that when a new point of data is added, the model will analyse the data already available and observe a specified number of points which are closer to it. This can be visualised in a 2D space as in Figure 19 but can be translated to higher planes having each dimension a feature. In order to do so it is important to scale all the variables.

There are many variations of this method. The simplest one is predicting the point chosen to be the most current between its neighbors. For example, observing the figure with 3 neighbors (plain circle) it would predict the missing point is red as there are 2 red points and one blue. If we changed and considered 5 neighbors (dotted circle) the point would be blue.

Another possibility considered in this work is weighting inversely to their distance. This means that closer points will have a larger weight. This means that, for 5 neighbors the point could be predicted as red as they are closer. This is an additional level of complexity that can help to improve the performance of the model.

The third variation adopted was weighting the events. As it may be desirable to predict an event when there isn’t in order not to miss some events and as nothing happening is much more frequent than having an event it could be desirable to give a greater weight to one class rather than another.

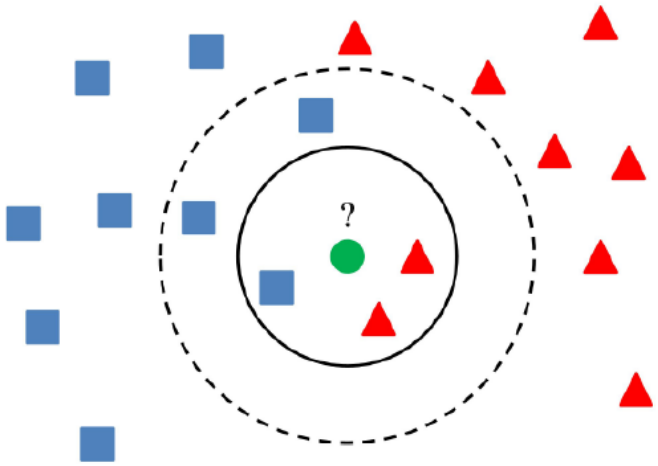

Figure 19: representation of data set to illustrate nearest neighbors model

The parameter that was tuned for each one of the different types of models is the number of neighbours that would be taken into consideration and weighting strategies.

4.1.2. Decision Tree

This model consists in a series of binary logical decisions in order to choose which category it will predict. Each decision is based simply on one of the features. One example is, if the rain of the previous day was superior to 30mm. This rule can lead to other branches or to the final decision. Four different types of decision tree models were explored.

The first and simplest is a simple **decision tree** classifier. During the training stage the model will create rules that maximize performance. Adjusting the value of the hyperparameters can help to avoid the model overfitting, fitting too perfectly to the train set and not only finding general tendencies, or underfitting, not being able to find the general tendencies of the model. Those parameters are the maximum depth of the trees, the minimum number of samples in each leaf and the weights of each class.

The maximum depth of the model stops it from growing excessively and overfitting. The minimum number of samples of a leaf stops branches extending with few samples once more avoiding the model overfitting. The class weights is a form of giving more importance to detecting events. As for the third variation of the nearest neighbors it is interesting to give more importance to predicting the maximum number of events than predicting an event that did not happen. If both classes had the same weight and a leaf had 21 days with no event and 10 with events it would predict there would not be an event. If the weight of the class with rock falls was 3 times larger it would predict that there would be an event.

The other **decision tree** classifiers, **bagging**, **random forest** and **histogram-based gradient boosting,** are produced by combining different trees. The idea is to create a more flexible model.

Bagging classifier starts by separating the training data into smaller sets randomly created. It is important to note that the same event can be present in different sets. This final group of sets is called a bootstrapped dataset. Decision trees are formed for each one of the subsets. The model is then a combination of those trees. The prediction will be the mode of the predictions.

The third method is called **random forest** and is similar to bagging but with one variation. During the process of creating the decision trees, instead of creating a rule based on one of the total number of variables, two variables will be selected randomly and the decision will be made between those variables. This process is done for each branch.

As for the simple decision tree classifier the maximum depth and minimum number of samples of a leaf will be optimized but for those methods the number of trees/ estimators will also be tuned. It is important to note that overfitting of the estimators is not a problem as the combination of them annulates those variations.

The last method is **histogram-based gradient boosting**. This method creates decision trees sequentially boosting the importance of points depending on previous error. The training process starts with the creation of a tree with one leaf, comparable to a dummy classifier, a model that will always predict the most common class. The residual is calculated for each element and a new tree will be created with the objective of predicting the residuals. The new predictions are now a sum of the dummy classifier with the predicted residuals multiplied by a learning rate. This process repeats many times in order to improve the quality of the result. As this method improves from previous mistakes this is considered machine learning.

In order to optimize gradient boosting the same parameters will be tuned as for bagging and random forest, but the impact is different. Gradient boosting generally thrives with underfitting decision trees with each one correcting the previous ones.

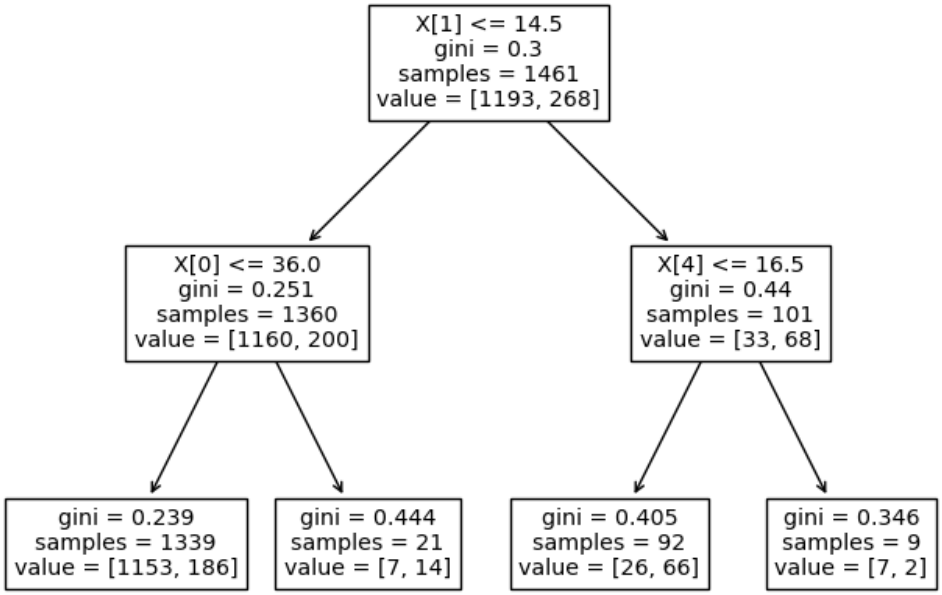

Figure 20 : Representation of simple decision tree

4.1.3. Neural Network (NN)

Receiving its name after the human brain this method mimics its logic having a series of neurons that activate each other. The simplest model is created by a first layer, with one neuron per feature, a series of layers with various neurons and a final layer with the output that gives the confidence of the model for each classification (Figure 21). Each neuron receives a value from the previous layer of neurons, transforms the value and sends this value to the next layer. The value each neuron from a layer receives is a linear combination of a number of neurons from the previous layers (Figure 22). For each neuron there will be one weight for each neuron on the previous layer that is connected to this neuron and a bias. Each neuron is not necessarily linked to neurons from the previous layer. This linear combination is the input value that will than be inserted on an activation function that will transform it into a value between 0 and 1. This value will then be passed on to the next layer until the last layer which will provide the probability the value is a certain class.

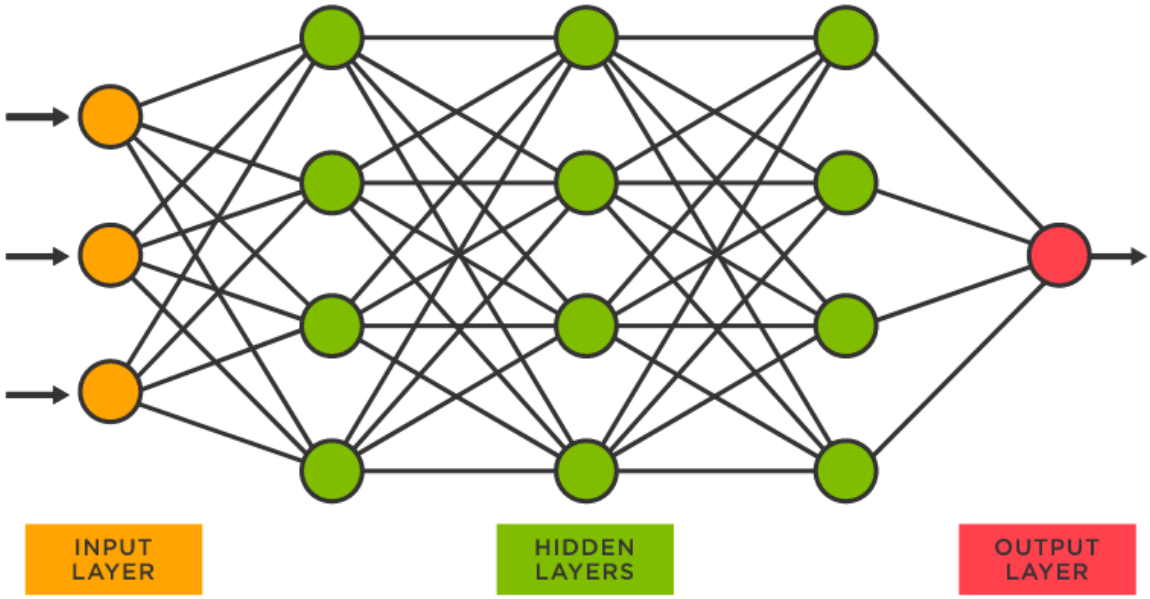

Figure 21: example of a Neural Network structure

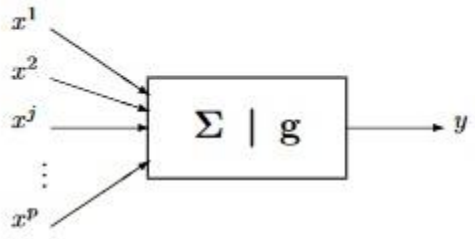

Figure 22: representation of a neuron (x_i the input, y output, Sigma for the sum of the inputs, and g is the activation function)

We say that a neural network learns from data because of its operating steps.

First, all trainable parameters of the model, like weight of connexion and biases are randomly initialised.

After, the model improves itself to minimize a loss function that measures the error between predictions and true labels. This is achieved by iterating over each sample of the train dataset:

- A prediction is made by the actual model configuration, then the loss is computed and its gradient with respect to all trainable parameters.
- Model configuration (all weights and biases) is upgraded, in the direction of gradient with a scaling factor called the learning rate.

These two points, forming a learning step, are repeated over the entire dataset and are called a training epoch.

After one epoch, model capabilities are evaluated on the validation dataset. Depending on the results, some hyper-parameters could be adjusted like the learning rate.

To learn correctly, we repeat many epochs to reach a good predictive model

There are many parameters that can be optimized for each problem like the structures for the intermediate layers, changing the activation function, the error function and also the learning rate.

This part of the work has been carried on by LISTIC laboratory. They decided to use two types of Neural Networks s:

- Dense Neural Network (DNN). are characterized by having all the neurons from a certain layer connected to all neurons from the previous and following layer. Figure 21is an example of a DNN model with all neurons connected to all neurons from neighbor layers. The hidden layer was formed by one or two dense layers using a logistic regression function to activate the neurons in order to observe non linear behaviour of the data.
- Convolutional neural networks (CNN). They are famously used to identify patterns on images. Convolution can be used to identify edges, curves and even more complex patterns. The convolution acts like a filter that enhances certain patterns. On images, the convolution kernels, called filters, are matrix while vectors on 1D times series, as for our dataset.

4.2. Performance metrics

A common way to observe the performance of a classification model is the confusion matrix. It compares the prediction and the actual values as shown by Figure 23. In order to compare different models, some values, representing the performance of the matrix, are calculated. Many different indicators may be used. The ones chosen were: accuracy, balanced accuracy, precision, recall, and F_β.

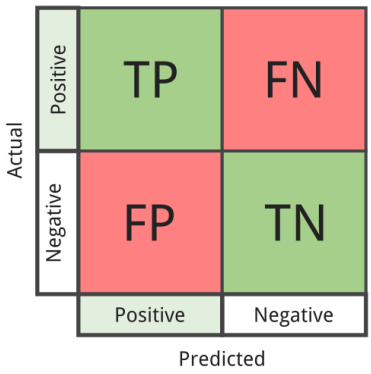

Figure 23: example of a confusion matrix with 2 classes: positive and negative

source: https://towardsdatascience.com/visual-guide-to-the-confusion-matrix-bb63730c8eba

The most classic value that measures the performance of a classification is the accuracy (eq?). It calculates the percentage of results that have been correctly classified. The main disadvantage of this method is when the classes are not equally present. For example, if 80% of the data were on class A a dummy classifier would always predict A and the accuracy of this model is 80%. This example shows the importance of comparing different indices to better understand the performance of a model.

An alternative is the balanced accuracy (eq?(Pb reference). It is not as intuitive as the accuracy, but the most important is that as the model better identifies the classes the value grows closer to one, and a dummy classifier will always score 0.5.

Two important values are the precision (eq?) and the recall (eq?). The precision is the probability that the event occurred when the model predicts it will happen. The Recall is the probability that the model predicted an event given it occurred. On operational terms for road management, augmenting the recall augments safety, as more rock falls will be predicted, and enhancing the precision will be economically important, as it will reduce the days the road is closed without any occurence of rock falls. It is important to have those concepts in mind as enhancing one will generally diminish the other.

$$Precision = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}$$

$$Balanced\ Accuracy = \left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right)/2 = (\text{recall} + \ ?)/2$$

An interesting indice to represent the performance of a classification model is the F_β (eq?). It is based on the precision and recall explained earlier and the analyst can tune the value

of β giving more or less importance to the precision or to the recall. The greater the β the greater the importance of the recall.

$$F_{\beta} = (1 + \beta^2) \cdot \frac{Precision \cdot Recall}{\beta^2 \cdot Precision + Recall}$$

5. Results of models on RN1 data

This section will present the performance of the different models. Firstly the performance of the operational code, following the results of the different models produced by artificial intelligence described in the previous section 4.1.

Training and validation general settings

As the period before the construction of protective elements (2000-2007) had more events it will be studied more in depth. It was decided that the period between the 01/01/2000 and the 25/02/2004 would be used to train the model while the data between 02/01/2005 and 27/12/2007 would be used to test the model. This permits us to avoid the year of 2004, that is distinct from the others, maintain temporal continuity of training and test data and also similar to reality as the operational code, that will be compared to our results, was also based on the events previous to 2004.

The predicted value, the output, was whether there would or would not be a rock reaching the road (0 when no rock fell, 1 if one or more rocks fell). One DNN model experimented with an additional class (0 representing no rock falls, 1 representing 1 rock fall, 2 representing more than 1 rock fall).

The input data for those predictions was the daily rain of the day (P0) and of the four previous days (Pi for i between 1 and 4, where "i" is the number of days before the studied date).

5.1. Operation code

The operational code takes into account the rain of previous days and uses it to define if the risk of blocks reaching the road is too high. The code studied is the R7 that dictates the road will be closed when:

- The daily rain in the last 24 h is superior to 15 mm
- The daily rain in the last 72 h is superior to 30 mm

This means that, in a certain instant, if the rain in the last 24h was superior to 15 mm the road will be closed, and if in the last 72h there was a period of 24h when the rain was superior to 30 mm the road will also be closed. As we are working with daily data a small adaptation must be made. It will be considered the model predicts an event if:

- The daily rain in the previous day is superior to 15 mm
- The daily rain in one of the previous 3 days is superior to 30 mm

This can be simplified as follows: P1≥15 ou P2≥30 ou P3≥30. With this model we can observe, for each day, the rain of the 3 previous days and know if the model would predict that there are or are not rocks falling on the road. A confusion matrix is built for the test set as shown by Figure 24. Out of the 1085 days of the test data set, the model predicted that 104 (in table) days would have an event and about ⅓ of those predictions were correct.

		Actual	
		0	1
Predicted	0	872	104
	1	71	38

Figure 24: Confusion matrix for the operational model for the test data set.

accuracy	0.84
balanced accuracy	0.60
precision	0.35
recall	0.27
F1	0.30

Table 2: Indices calculated based on the confusion matrix to quantify performance of the operational model.

For this confusion matrix all the indices described in the previous section were calculated and are presented on table 2. We can observe that the accuracy is quite high, which is expected as the data is not balanced. During 87% of the days there are no rocks falling along the road. This means that even a dummy classifier could have a better score than the operational one. On the other hand, we know that it has its value as it is capable of correctly predicting 38 events.

5.2. Nearest Neighbors

The first and simplest model created is nearest neighbors. This method predicts if there will be rocks falling along the road based on similar days in the past. In order to produce the best model the number of neighbors, that will be taken into account in order to predict new data points, will be optimized. This is done by varying this number and observing the result of the

model for the training and test sets. The indice considered to measure the performance of the model is the balanced accuracy. The number of neighbors is varied between 1 and 50 and the result is presented by Figure 25.

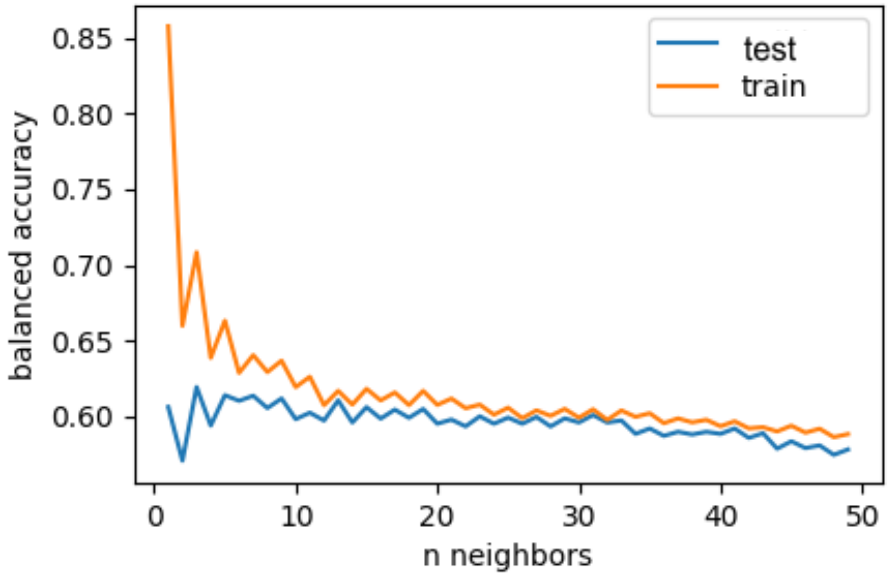

Figure 25: Balanced accuracy of the models for training and testing data and its variation with the number of neighbors.

With one neighbour we have the maximum performance of the training data but the difference between both data sets is significant indicating that the model is overfitting. As the number of neighbors decreases the performance of the training data also decreases, becoming closer to the one of the test data. The balanced accuracy for the test data set also decreases with a larger number of neighbors, but much slower maintaining it's value approximately 0.6. The value of 15 is adopted as the model does not overfit. Comparing figure 24 and figure 26 we observe that the Operational code is capable of correctly predicting 8 additional rock falls, but at the cost of incorrectly predicting 77 additional events.(indiquer les conséquences pour la gestion de la route)

By comparing the indices calculated for this model and the operational model it is noticed that the accuracy has increased and is even slightly superior to the dummy classifier, which has an accuracy of 0.866. The precision also increased being superior to 0.5, meaning the majority of the times the model predicts a rock will fall it actually happened. A higher precision is important as less false predictions will happen leading to less days the road would be closed without any risk to drivers. On the other hand, the accuracy and the recall have fallen. This means that the operational system is safer as it predicts more events than this first model.

		Actual	
		0	1
Predicted	0	917	116
	1	27	30

Figure 26: Confusion matrix representing the result of predictions based on the nearest neighbors method for the test set.

accuracy	0.87	0.84
balanced accuracy	0.59	0.60
precision	0.53	0.35
recall	0.21	0.27
F1	0.30	0.30

Table 3: Indices calculated based on the confusion matrix to quantify performance of the nearest neighbors model and the operation model.

5.3. Decision trees

For this model, more than one parameter are tuned in order to optimise the results. A heat map was plotted to observe how the maximum depth and the minimum samples per leaf would affect the results. The $F_{\beta=1}$ score and the balanced accuracy are calculated for each combination and a heat map is created (Figure 27). This image permits the choice of the combination of parameters that optimises the result. It is important to note that both indices are only calculated in respect to the test set (a grid search is made) and it must be later verified if overfitting does occur.

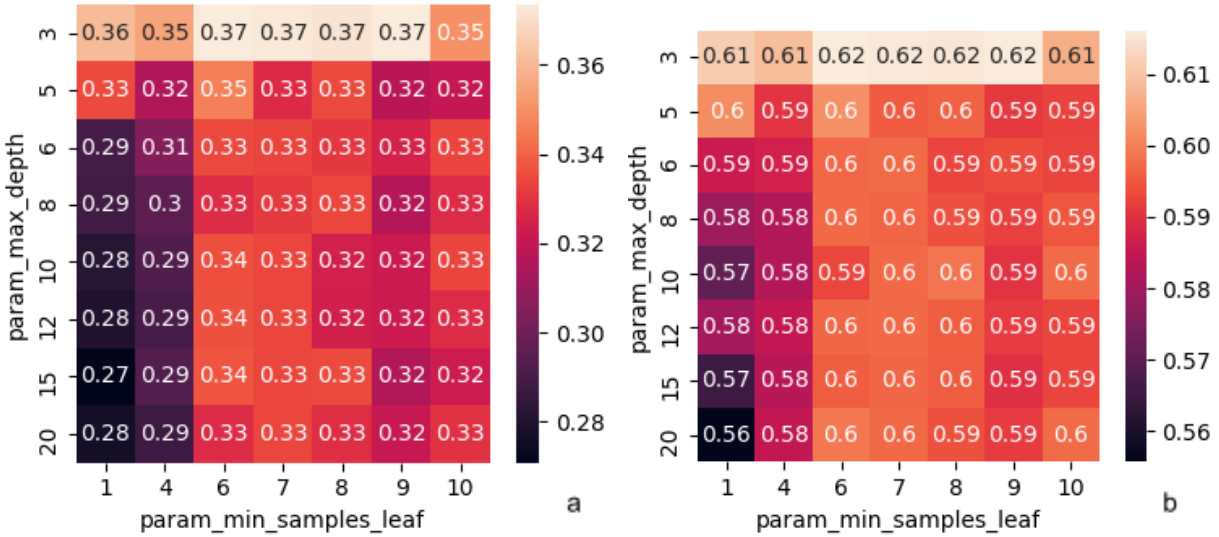

Figure 27: heat mapping of the performance of decision tree model as the hyperparameters varie for models with classes with same weights. Performance of model measured by (a) $F_{\beta=1}$ (b) balanced accuracy

Observing both heat maps (Figure 27) it is visible that the $F_{\beta=1}$ and the balanced accuracy are related. This is logical as both indices should increase as the model improves and is capable of correctly predicting more rock falls. The model improves as the maximum depth decreases. This indicates the trees are overfitting and searching to fit the model too closely. This is especially true when the min sample leaf is smaller facilitating the overfitting. The optimum model would be a small tree with a maximum depth of 3 and a minimum number of samples per leaf of 8. This model was created and the confusion matrix is presented in figure 28.

		Actual	
		0	1
Predicted	0	909	114
	1	35	32

Figure 28: Confusion matrix representing the result of predictions based on the decision tree method for the test set

	Decision tree model		
	Train	Test	Operational
accuracy	0.85	0.86	0.84
balanced accuracy	0.63	0.59	0.60
precision	0.73	0.48	0.35
recall	0.28	0.22	0.27
F1	0.40	0.30	0.30

Table 4: Indices calculated based on the confusion matrix to quantify performance of the decision tree model, for the test and train set, and the operation model.

The results for the train set are superior to the one for the test set. This is normal as the model performs better on the data it was trained on. Both models have similar accuracies and balanced accuracies, but the training data is much more precise than the test set leading to a superior $F_{\beta=1}$ score. This happens even though a small depth was adopted. Possibly an even smaller depth could be adopted or even a larger number of samples per leaf, but this result is considered satisfactory.

Similar to the nearest neighbors this model has a smaller balanced accuracy as it predicts less rock falls than the operational model, but it also has a higher precision, meaning that when it predicts that there will be a rock fall there is a higher chance it will be true. In order to increase the safety on the road the importance of predicting rock falls should be increased. This means that the model will favor predicting rock falls even if it will have a negative impact on the precision leading to more days the road is closed having a negative economic impact and becoming an inconvenience to drivers. So, it is proposed to weight the predictions and it will be adopted that the weight of predicting an event be 3 times greater than the ladder. The heat maps are re-calculated with the adjusted weights.

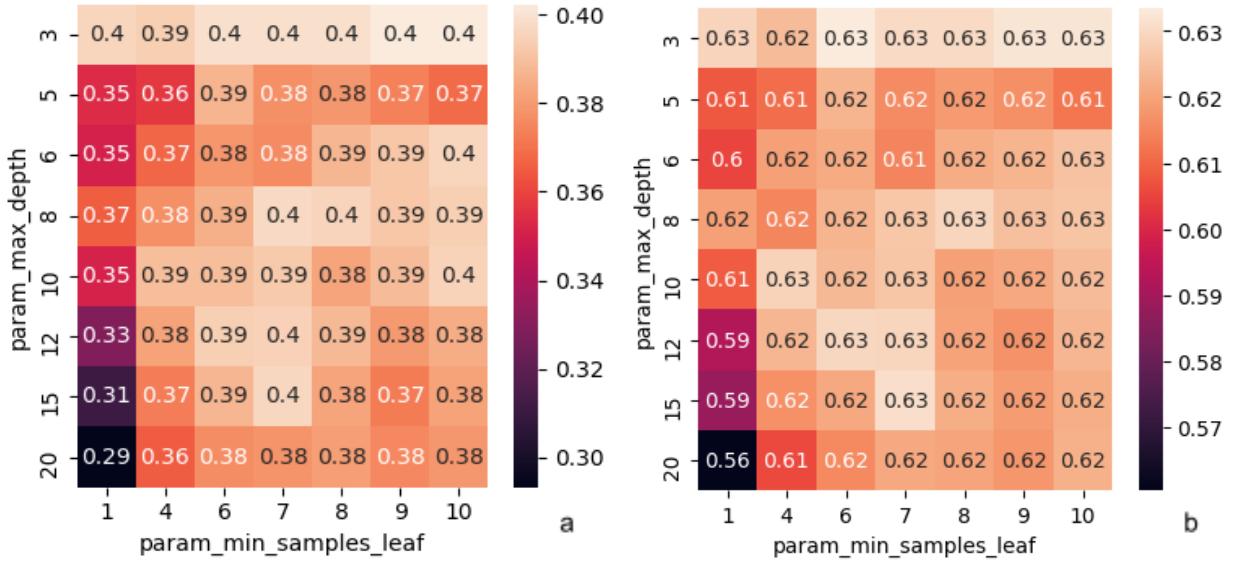

Figure 29: heat mapping of the performance of decision tree model as the hyperparameters varie for models with classes with different weights. Performance of model measured by (a) $F_{\beta=1}$ (b) balanced accuracy

Similar to the heat maps on Figure 27, Figure 29 shows a great performance for trees with minimum depth and high amount of samples per leaf. On the other hand, there seems to be an optimum model for a tree with a depth of 12 and min_samples_leaf equal to 7. Either option, max_depth = 3 with min_samples_leaf = 9 or max_depth = 12 with min_samples_leaf = 7, could be the best model. Both were calculated and the second one overfitted considerably. The balanced accuracy for the training set was 0.74 and the $F_{\beta=1}$ was equal to 0.56 while the test set was considerably worse (balanced accuracy = 0.63 and $F_{\beta=1}$ = 0.4). The model with lower depth had a similar performance and the results for the training and testing sets were much closer.

		Actual	
		0	1
Predicted	0	867	99
	1	77	47

Figure 30: Confusion matrix representing the result of predictions based on the decision tree classifier method for the train set.

	Decision tree classifier	Operational
accuracy	0.84	0.84
balanced accuracy	0.62	0.60
precision	0.38	0.35
recall	0.32	0.27
F1	0.35	0.30

Table 5: Indices calculated based on the confusion matrix to quantify performance of the decision tree classifier, for the test and train set, and the operation model.

This is the first model which is capable of correctly predicting more rock falls than the operational model with 18 additional correct predictions. Those predictions come with an additional 6 days when the model incorrectly predicted that there would be an event. This means 6 additional days the road could be operational with no accidents. Even though this model is safer for drivers the precision is greater than the operational. For future models this balance of weights will be adopted. Additionally the tuning will be done solely with the balanced accuracy as it is strongly correlated with the $F_{\beta=1}$.

Bagging Decision tree Classifier

As described in the previous section, this method is an optimization of decision tree approach by using a combination of trees in order to predict if a rock will detach on a certain date. For this model both hyper-parameters tuned for the decision tree will be optimized once again and additionally the number of estimators will also be tuned. In order to visualise the interaction between the three variables 3 distinct heat maps will be represented. This enables the choice of the hyper-parameters that will be maintained.

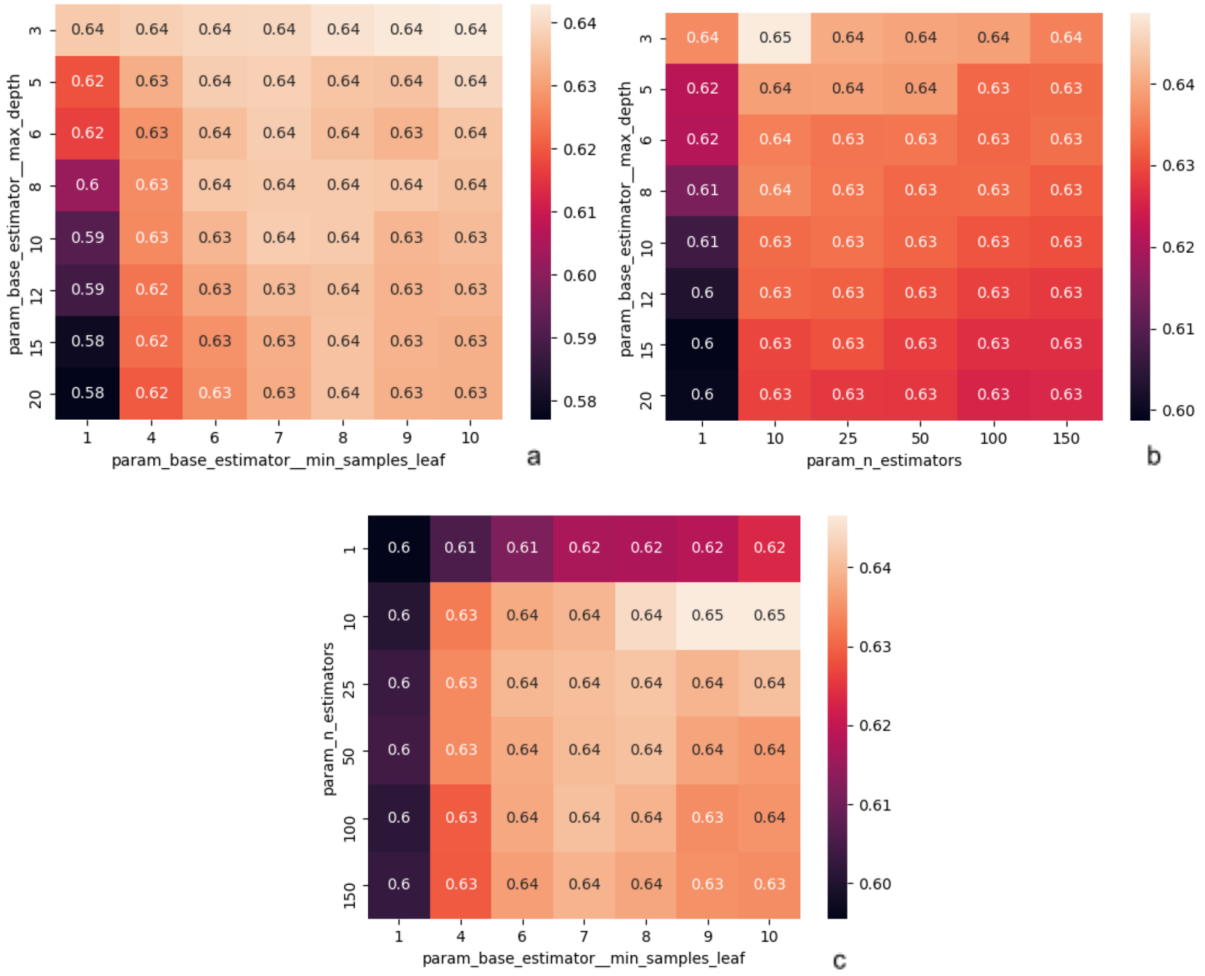

Figure 31: heat mapping of the balanced accuracy of bagging tree classifier as the hyper-parameters vary. The combination for each figure are: (a) minimum number of samples per leaf and maximum depth, (b) number of estimators and maximum depth, (c) minimum number of samples per leaf and number of estimators

Some general tendencies is that when the min samples leaf and the number of estimators are equal to 1 the models perform worse. It is important to note that when only one estimator is used the model is a simple decision tree as the one on the last section. Lowering the max depth tends to increase the performance of the models. It was decided to use: max_depth=3, min_samples_leaf=9, n_estimators=10.

		Actual	
		0	1
Predicted	0	863	96
	1	81	50

Figure 32: Confusion matrix representing the result of predictions based on the Bagging tree classifier method for the train set.

	Bagging tree classifier	Operational
accuracy	0.84	0.84
balanced accuracy	0.63	0.60
precision	0.38	0.35
recall	0.34	0.27
F1	0.34	0.30

Table 6: Indices calculated based on the confusion matrix to quantify performance of the bagging tree classifier, for the test and train set, and the operation model.

The model produced by the bagging method has superior results for all the indices calculated. It is capable of identifying 3 additional rock falls while increasing the precision of predictions. On the other hand, even though it increases the precision, as it predicts more rock falls it also increases the days the road is closed and an argument could be made that the operational model gives enough safety to drivers while maximizing profit.

5.4. Neural Networks

Dense Neural Networks

Various models were created by Courteille [Courteille, 2021] to examine the capabilities of NN to predict the rock falls based on the rain. The same training and testing set were used, but the input data included not only the rain of the previous 5 days but instead the rain of the previous 10 days. A DNN was created with one hidden dense layer with relu activation and a final dense layer with softmax activation. Weighing samples by inverse of their occuring frequencies <- unbalanced dataset was also done. The resulting confusion matrix is represented by the Figure 33

DNN		actual	
		0	1
predicted	0	780	82
	1	158	60

Figure 33: confusion matrix representing result of DNN model with the rain of the previous 10 days for the test data

	DNN	Operational
accuracy	0.78	0.84
balanced accuracy	0.63	0.60
precision	0.28	0.35
recall	0.42	0.27
F1	0.33	0.30

Table 7: Indices calculated based on the confusion matrix to quantify performance of the DNN and the operation model for the test set.

This model produced by the DNN is safer than the operational model with a higher recall. It is capable of identifying almost half of all the events. But, this comes with a lower precision. The model is capable of predicting more rock falls making it a safer alternative, but it will also predict more frequently an event will occur while no rock falls. This model would lead the road to be closed 20.2% of the time.

With the objective of better understanding the data a third class was added. Class 0: no event, class 1: 1 rock fall, class 2: more than one event. It is expected that when we have more than one rock fall the same day it will be related to more important rains. In order to more easily compare to the other models and also to have a more operational result classes 1 and 2 are combined and a final confusion matrix (Figure 34) is produced.

Comparing the overall result of the model using 3 classes and 2 classes we observe they are similar. With 3 classes the model was slightly safer predicting 2 additional rock falls, but also decreased the precision closing the road 38 additional days. The confusion matrix with the 3 classes conforms to the hypothesis that days with more than one instance will be easier to predict. Only 14 of the 43 days (33%) with more than one event are classified as a day with no event while two thirds of the days with only one event were classified as days no rocks would fall.

In order to improve predictions a model was constructed including not only the rain as a variable but also the number of blocks that reached the road on the previous days. When a block

falls it destabilizes the bedrock. The distribution of stresses is altered and this can cause other blocs to fall. This additional information did not improve the performance of the model.

DNN 3-classes		actual		
		0	1	2
predicted	0	737	66	14
	1	164	29	14
	2	32	4	15

Figure 34 : confusion matrix for DNN model using 3 classes for the test data

DNN 3-classes		actual	
		0	1
predicted	0	737	80
	1	196	62

Figure 35 : confusion matrix for DNN model using 3 classes adapted to 2 classes

	DNN 3-class	DNN 2-class	Operational
accuracy	0.74	0.78	0.84
balanced accuracy	0.61	0.63	0.60
precision	0.24	0.28	0.35
recall	0.44	0.42	0.27
F1	0.31	0.33	0.30

Table 8: indices for the DNN 3-classes model adapted to 2-classes and the Operational model

Convolution Neural Network

This method is more adapted to this data set compared to the other neural networks as it was capable of detecting at least as many rock falls as the others while incorrectly predicting less events. Comparing it to the operational model it is safer with a higher recall and a similar, yet lower precision. This model is safer than the operational code, but it will dictate the road stays closed for a longer period of time.

CNN		actual	
		0	1
predicted	0	805	80
	1	133	62

Figure 36 : confusion matrix for DNN model using 3 classes adapted to 2 classes

	CNN	Operational
accuracy	0.80	0.84
balanced accuracy	0.65	0.60
precision	0.32	0.35
recall	0.44	0.27
F1	0.37	0.30

Table 9: indices for the CNN model and the Operational model

5.5. Other models

As described in the methodology Random forest classifier and Histogram gradient boosting classifier were also calculated. The same processo of optimization and analysis of the confusion matrix were done. Those results are present on the annex of this document.

The random forest model had a similar result to the bagging classifier with a slightly worse result for the test set, but superior for the training set indicating overfitting. The parameters for the generation of this model were:

5.6. Discussion

The tested models return varied performances [Table 10]. Generally there is a trade off of some characteristics in order to improve in others. The general tendency is that with stronger rains there is a greater chance that an event will happen. The different models find this general rule. Some will be safer for drivers meaning a smaller amount of rain will trigger the model. This is reflected by a high recall. An example are the NN models which tend to favor security. On the other hand they also have a lower precision meaning the road will be closed more frequently.

The most economically interesting models are those with the highest precision for which the models will predict a rock will fall with a greater accuracy avoiding closures on days an event is not present. An example of this are the nearest neighbors model and the decision tree model. Both of those have the highest precision but the smallest recall.

The NN models were tuned to give great importance to the safety (not tuned, but data-driven with unbalanced dataset) while the nearest neighbors and simple decision tree gave

equal importance to predicting and not predicting, which led to models that were not as safe as the operational, meaning the risk of driving the road would decrease if this model was used. Two models (decision tree 1:3 and bagging) were created with an importance 3 times greater to correctly predicting rock falls rather than correctly predicting no rock falls and those seem to be a satisfactory trade off. Those models have similar yet slightly higher indices. Even though those models do have a higher precision than the operational they do close the road a greater amount of times. It could be argued that the risk of the operational model is already sufficiently low and that as it closes the road less than the other models it would be preferable to the other models.

The 7 models created are different and are superior to the operational on some aspects, but all of them have similar accuracy and balanced accuracy. This indicates there is a part of the data that can not be explained by the rain by itself. Other variables could be included and could help to explain a part of those rock falls or they could simply be random and no other explicative variable could improve the performance.

Models	Accuracy	balanced accuracy	precision	recall	F1
Operational	0.84	0.60	0.35	0.27	0.30
Nearest neighbors	0.87	0.59	0.53	0.21	0.30
decision tree	0.86	0.59	0.48	0.22	0.30
decision tree 1:3	0.84	0.62	0.38	0.32	0.35
bagging	0.84	0.63	0.38	0.34	0.36
DNN	0.78	0.63	0.28	0.42	0.33
DNN 3-classes	0.74	0.61	0.24	0.44	0.31
CNN	0.80	0.65	0.32	0.44	0.37

Table 10: Combination of result of different models

When considering the application of an AI model to the operation of the road and comparing how it would affect drivers it is important to consider the frequency of the closures and their durations. Will the closures be punctual or will they be continuous? As rock falls are correlated with the rain which is a seasonal event the events will also be seasonal and be concentrated on certain moments of the year. This means there will be some continuity as is the case for the operational system. Figure 37 illustrates this continuity during the first months of 2004 when there is the greatest concentration of events.

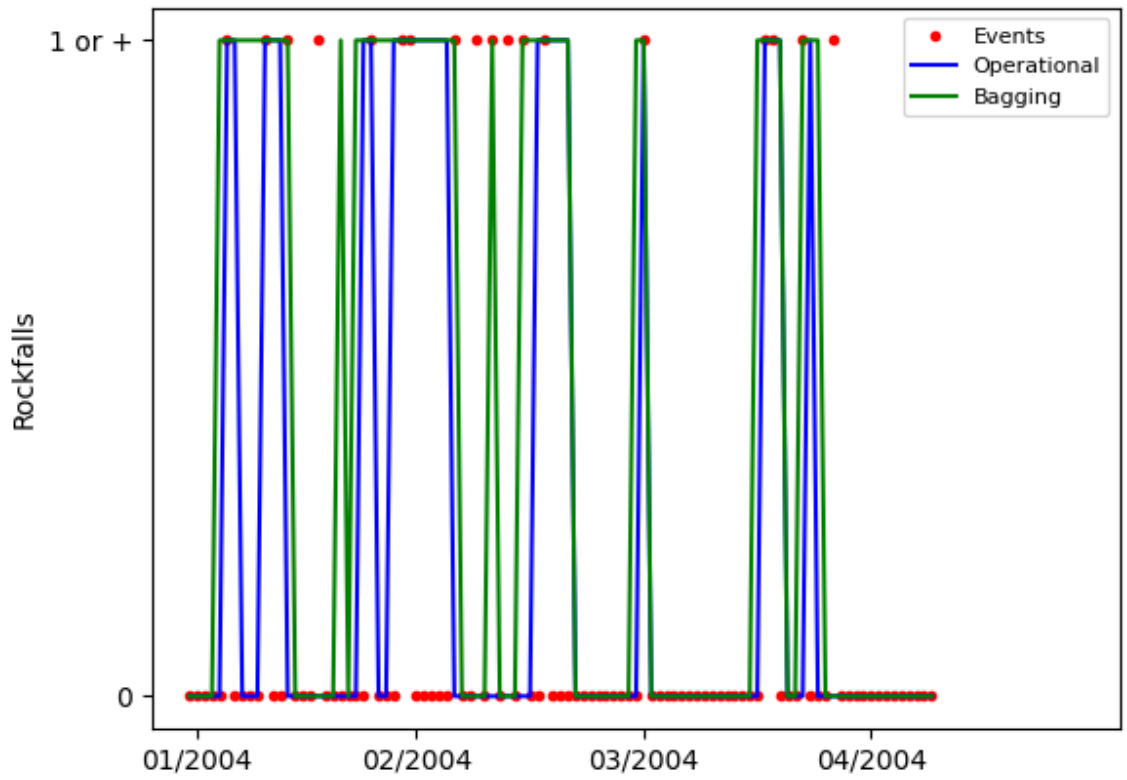

Figure 37: Comparison of predictions with time of the operational model and the model created by bagging with weight 5 times greater to predicting an event during the first part of the year 2004.

Conclusion

This study consisted in studying a cliff and the rock falls that occurred on the road parallel to it in previous years.The first step was selecting which site to study. The RN1 from la Réunion was chosen due to its detailed data set with many events. The data needed to be acquired from different partners. When the data was transferred it could then be formatted and then analysed. It is important to do a simple analysis to have a good understanding of the situation and to spot any early tendencies and possible incorrect data. Subsequently predictive models can be created and finally analysed in order to draw conclusions.

3 different methods were used to create models, nearest neighbors, decision trees and NNs. For each one models were tuned in order to produce an optimal performance for predictions over the test data set. The results were summarized on confusion matrices and the models were compared between each other and with the operational system.

This study has two goals. The first is to better understand the causes of rock falls. By creating predictive models based on past events it is possible to understand which are the most important factors and at what intensities the blocks will fall. The models created by this study indicate that the rain is capable of explaining a part of the events but not all of them. There is a

percentage of them that are not linked to the rain. The highest percentage of events that were correctly predicted was 44% by the NN models. As developed in the introduction, a combination of factors cause the rocks to detach and reach the road and this study focuses on one, the rain.

The second objective is to compare the performance of AI with other methods and its capabilities for predicting rock falls. This includes the precision of predictions, the applicability of the models and the time to produce the models. 3 different types of predictive models were compared to the operational one, and the performances (Table 10) were similar. Some models were safer and some were less restrictive, but with the same input we can affirm that both provide similar performances.

It is also important to consider the applicability of the models. The operational model that was adopted after 2004 was simple, making it easy to apply. The other models are more complex. It is harder for an operator to manually predict if the road will be closed or not, and even harder for the driver to do the same. The applicability of the model would also be harder and a greater inconvenience for drivers if the road was constantly opening and closing. As discussed in section 5.6. the openings and closures of the road with the different models are similar.

One of the greatest improvements of the AI over the operational model is the time needed to produce a model. Both methods need an initial arduous task of preparing the data, but after it is concluded the process of creating an effective model is reasonably simple and quick to a data scientist. The time to adapt the model, created by artificial intelligence, to be operational is yet to be analysed.

This study presented positive results that indicate that AI could be used in operational scenarios in order to optimize road control. The models could produce results at least as performant as the one that was adopted. It is quicker to create the AI model saving time. Even though the results are promising there are still limitations to this work and areas that could be expanded.

As the different hyper-parameters were tuned for the different models, the goal was to maximize the balanced accuracy of the test set. This means they were tuned specially to the studied test set. In order to avoid this, a validation set is used. The model is trained using the train set, the hyperparameters are tuned to the test set and a validation set is used to verify the performance of the model to an unseen data set.

This study focused mainly on the period between 2000 and 2007. Studying the period after 2008 would give us an alternative data set to test the performance of the different methods. As it has fewer events it would help analysing the capability of the model for data sets with a greater imbalance of classes.

As explained in section 5.6., the rain is not capable of explaining the totality of the events. Studies including other possible explicative variables could help explain another part of the rock falls. During the discussions the applicability of the AI model is taken into account. It wouldn’t be ideal to apply the model directly. A future study would be needed to modify the model in a way it becomes easily applied to control the road circulation.

In order to better understand the limitations and capacities of AI to predict rock falls it is important to study different sites. This variety will permit us to verify what was done on this work making sure it is applicable to different contexts. Moreover, the RN1 was a great case for a first study due to its simplicity not having the freeze thaw cycles and earthquakes, but creating models where those features are also taken into consideration could be more adapted to AI.

The predictive models always aimed to predict the day a rock reached the road while another important information is the position it fell. Initial analysis of the spatial distribution was made, but no models were created incorporating the spatial aspect of the cliff. A space time model could be operational by incorporating partial closures of the road. Having sections of the road completely open helps the driver have more comfort and reduces the time of travel.

References

n.d. observatoire regional des risques majeures. Accessed 09 15, 2021. http://observatoire-regional-risques-paca.fr/article/eboulements-chutes-pierres-blocs.

BATISTA, Dominique. 2005. “Etude statistique de l'aléa chute des pièrres sur la RN1 à la Réunion.” (05).

Batista,, Dominique. 2016. “Analyse statistique de l'aléa chute de blocs sur la Route du Littoral Île de la Réunion.” (08).

“Découvrir La Réunion.” 2020. Habiter à la Réunion. https://habiter-la-reunion.re/decouvrir-ile-de-la-reunion/.

DELONCA, Adeline. 2014. “Les incertitudes lors de l'évaluation de l'aléa de départ des éboulements rocheaux.” (12).

HUMBERT, M., R. PASQUET, and L. STIELTJES. 1981. *Les risques géologiques dans les cirques de salazie et de cilaos*. N.p.: BRGM.

Introducing Uncertainty in Risk Calculation along Roads Using a Simple Stochastic Approach. 1998. N.p.: BRGM.

Jaboyedoff, Michel, Tiggi Choanji, and Marc-Henri Derron. 2021. “Introducing Uncertainty in Risk Calculation along Roads Using a Simple Stochastic Approach.” (03).

Luckman, B. H. 2013. *7.17 Processes, Transport, Deposition, and Landforms: Rockfall*. Vol. 7. N.p.: Academic Press. https://doi.org/10.1016/B978-0-12-374739-6.00162-7.

Varnes, David J. 1978. “Slope Movement Types and Processes.” In *LANDSLIDES: ANALYSIS AND CONTROL*.

https://web.archive.org/web/20190224023805/http://pdfs.semanticscholar.org/5add/f9aca1f44c5c643180c48ae6015e859c8c9b.pdf.

Wyllie, Duncan C. 2014. *ROCK FALL ENGINEERING: DEVELOPMENT AND CALIBRATION OF AN IMPROVED MODEL FOR ANALYSIS OF ROCK FALL HAZARDS ON HIGHWAYS AND RAILWAYS*. https://open.library.ubc.ca/media/stream/pdf/24/1.0167542/1.

Attachments

Additional result of models : Random Forest

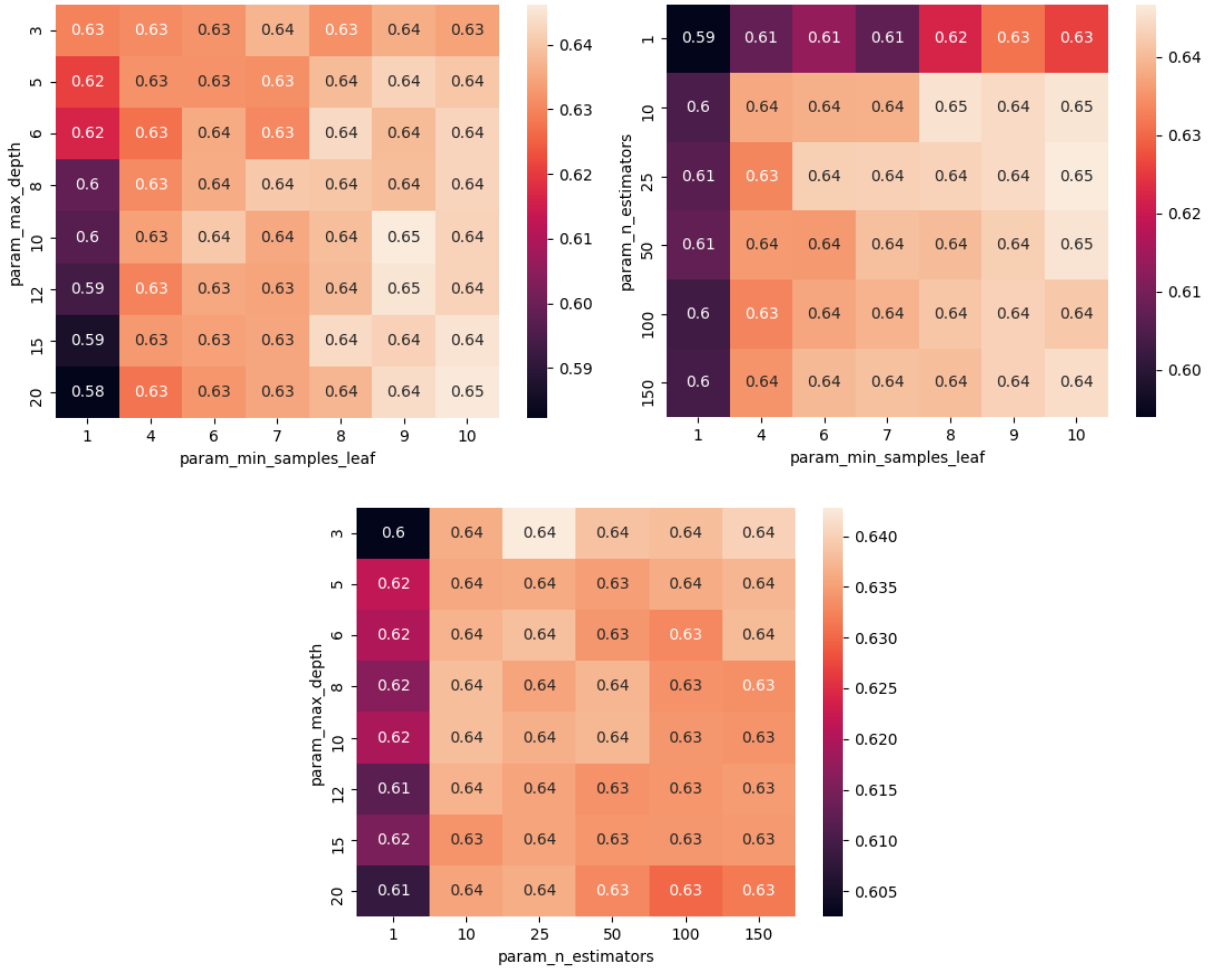

Figure 1 : heat map representing the balanced accuracy for different combinations of hyper-parameters for a Random forest model with a weight 3 times greater for predicting there will be an event

Values adopted :

n_estimators=100
max_depth=10
min_samples_leaf=9

Random Forest		actual	
		0	1
predicted	0	849	94
	1	95	52

Figure 2 : confusion matrix for Random forest model adopting a weight 3 times greater for predicting there will be an event

	Random Forest	Operational
accuracy	0.80	0.84
balanced accuracy	0,62	0.60
precision	0,35	0.35
recall	0,34	0.27
F1	0,35	0.30

Table 1: indices for the Random forest model and the Operational model

Additional result of models : Histogram based gradient boosting

Simple model made with no optimisation. The default value, defined by sklearn, was used for most of the hyper-parameters with the exception of the maximum depth that was fixed at 3.

HistGradBoosting		actual	
		0	1
predicted	0	901	109
	1	43	37

Figure 1 : confusion matrix for histogram based gradient boosting model

	HistGradBoosting	Operational
accuracy	0.82	0.84
balanced accuracy	0,60	0.60
precision	0,46	0.35
recall	0,25	0.27
F1	0,31	0.30

Table 1: indices for the HistGradBoosting model and the Operational model

6.2 Rapport du contrat orienté IA

Rockfall analysis and predictions using machine learning on RN1 road (Réunion)

Octobre 2021

Technical Report for RINA project

Author: Hermann Courteille

Collaborator: Abdourrahmane Atto, Emmanuel Trouvé, Nicolas Méger (LISTIC)

Contents

0.1 Introduction

In this report, we study rockfall characteristics in Réunion RN1 road and aim at building a predictive model, which could replace existing, empirical, operational rules. Understanding more precisely dynamics and dependencies of the process will allow us to design relevant data-driven model.

This work have been done by LISTIC as part of RINA project, asked by the other partner(CEREMA, BRGM, Géolithe), which are more in the applied fields of rockfall. Besides, a complementary work on other machine Learning models (decision trees, nearest neighbour) on same data has been achieved by CEREMA trough the internship of Guilherme Cunha de Barros Santos. LISTIC follow up this second work and supervise it on the machine learning part.

In Chapter 1, we present available data, their format, first analysis with previous work and descriptive statistics about on the dynamic of rockfall event.
In Chapter 2, we precise different mathematical manner to represent rockfall events. Some relevant papers , linked to them, are presented in Annexe B.
Chapter 3 summarize numerical experiments with different models (Neural Networks, General linear Model...), comparing best models with expert rules used in operational center. In conclusion, we provide some preliminary results on the capability of machine learning to predict rockfall and some perspectives of improvement.

Chapter 1

Réunion Rockfall data analysis

1.1 Data characteristics

All available data are recorded in several tables that could be used in a relational database, see Table 1.1. As it is simpler for single user and access, we just use pandas library to operate on these files ".csv". We list here available information in spatial and/or temporal domain, on RN1 road. Whole dataset is physically **split in 2 periods**, according to the **major protection works in 2008**.

1. **events.csv**(*Temporal* and *Spatial*) contains each rockfall with its date, its position along RN1 road called PR for "Point de Repère" (in km) and its size (weight in kg, this variable is not complete) . This table count 902 events from January 2000 to May 2018,

2. **rain.csv**(*Temporal*) contains daily quantity of rain in mm from January 2000 to December 2019, more precisely, it is the maximum value between the 3 stations along the RN1 road(high correlation across this 3 geographical points)

3. **zone_protect.csv**(*Spatial*) which associate at each slice of the road, $[pr_{start}; pr_{end}[$ a binary value : 1 if there is a protection over the road and 0 otherwise.

4. **zone_geo.csv**(*Spatial*) which associate at each slice of the road $[pr'_{start}; pr'_{end}[$ the type of geological layer

   - Zone 0 : lower layer predominance
   - Zone 1 : upper layer predominance
   - Zone 2 : intermediate layer present

5. **standarized_area.csv**(*Spatial*) of the cliff profile every 10 m

6. **SCI.csv**(*Spatial*): a Standardized Concavity Index of the cliff profile every 10 m

7. **TWI.csv**(*Spatial*): a Topographic Wetness Index of the cliff top every 10 m

date	pr	size
2008-12-06	9.45	3
2008-12-21	8.15	40
2008-12-29	12.8	5
2009-01-03	12.25	1.5
2009-01-03	12.35	1
...	...	...

(a) events.csv

date	rain
...	...
2008-12-19	2
2008-12-20	28
2008-12-21	8
2008-12-22	0
...	...

(b) rain.csv

zone	pr_{start}	pr_{end}
0	1.7	2.9
2	2.9	3.1
0	3.1	5
...	...	...

(c) zone_geo.csv

zone	pr_{start}	pr_{end}
0	1.7	2
1	2	5.3
0	5.3	5.5
...	...	...

(d) zone_protect.csv

Table 1.1: Database related to Réunion

Remarks:

- Two or more rockfalls can occur the same day but at different position PR.
- Protection and geological zone can be see as categorical function define for each PR of the road, i.e. $f(pr)$=type of zone.
- Indexes standarized_area, SCI and TWI.csv are computed every 10 m from the cliff profile using a DEM model. Time, location and size are own characteristics of rockfall hazard. Rain, geological and protections, are external but can induce rockfall process.

1.2 Previous studies on rockfall and rainfall

In the paper [3], they look at **rockfall hazard** which is the probability that a rockfall of a given volume occurs in given area within a specified time interval. This definition is mathematically reflected by point process definition (see next chapter). They examine correlation between meteorological factors and rockfall on 3 databases including RN1 in Réunion like us (Burgundy and Auvergne too). Their database is composed of 949 rockfalls events between 1998 and 2009. They highlights presence of correlation between rockfalls and rainfall and between rockfalls (and minimum temperatures for other sites). More precisely correlation is significant from 0 to 5 days.

In the paper [2], they study rockfall from the Mont Saint-Eynard (Grenoble) thanks to laser scan and photos at different time frequencies. Their database count 1068 rockfall events acquired from photos and a laser scan. Two external factors are well studied : rainfall and freeze-thaw cycle. As negative temperature does not occur in Réunion, we just mention here results about rainfall.

Adding to rainfall times series, they used **rainfall episode**. A rainfall episode begins when it rains back after a given delay without rain. This time delay is set in order that 2 rainfalls episodes are independent, i.e. the effect of first one stop before the second one begins. Such delay on the

Mont Saint-Eynard was set to 24 hours.
They show that rainfall frequency during rainfall episodes is 3 times higher than without meteorological event. Thus, it is multiply by factor of 7 in the first 25 h and decreases to 1 after 50h. "Considering the mean intensity since the beginning of the episode, the influence factor amounts to 27 if the intensity is higher than 5 $mm.h^{-1}$. Considering the rainfall amount, it amounts to 7.5 if the rainfall amount is between 30 and 40 mm".
Therefore, **rainfall episode** belong to factors which can well explain a part of rockfall hazard and should be take into account in future predictive model.

1.3 Comparison before/after major protection works in 2008

During this period, 139 rockfalls events occurs. If we aggregate them in space, we get only 104 times where 1 or more rockfall occurs. As we can see on Figure 1.6, daily number of rockfall span from 0 to 5. Sample auto-correlation and correlogram show a slight annual pattern in number of rockfall.
If we look at times between events, mean time between rockfall is about 25 days and we can see in Figure 1.7, that inter-times distribution is badly fitted by exponential law. So point process of rockfall times is clearly not an homogeneous Poisson Process.

Rainfall (Figure 1.6)

Daily rainfall quantity span from 0 to 175 mm with. Times between two consecutive rainfall is around 3 days in average. Sample auto-correlation and correlogram confirm the annual seasonality that we could already see in time series. A second significant peak in correlogram is present for 2 cycle/year ie an half-year periodicity.

Correlation between rockfall and rainfall (Figures ***)

If we superpose daily or monthly time series of rockfall and rainfall, we can observe temporal similarities already observed in [3].To quantify such dependencies, we plot as well cross-correlation (see chapter 2 for definition).
Daily rockfall times series is correlated to rainfall time series at short-term. More precisely, maximum of correlation is for 1 day lag and it decrease till 5 days where it becomes not significantly.

Location, size, protection and geological zones (Figure ***)

Location of events are beyond cliff (between PR 1.7 and PR 13). They are unevenly distributed : mainly around PR 6 and PR 12 .
Size of events span from 0.2 to 12 000 000 kg. Majority of events have size lower than 100 000 kg (see join distribution location/size). Only 4 events are bigger and all are located between PR 9.2 and PR 10.05.

We can see on right figure that lower layer predominance is mostly present before PR 5 and after 10.

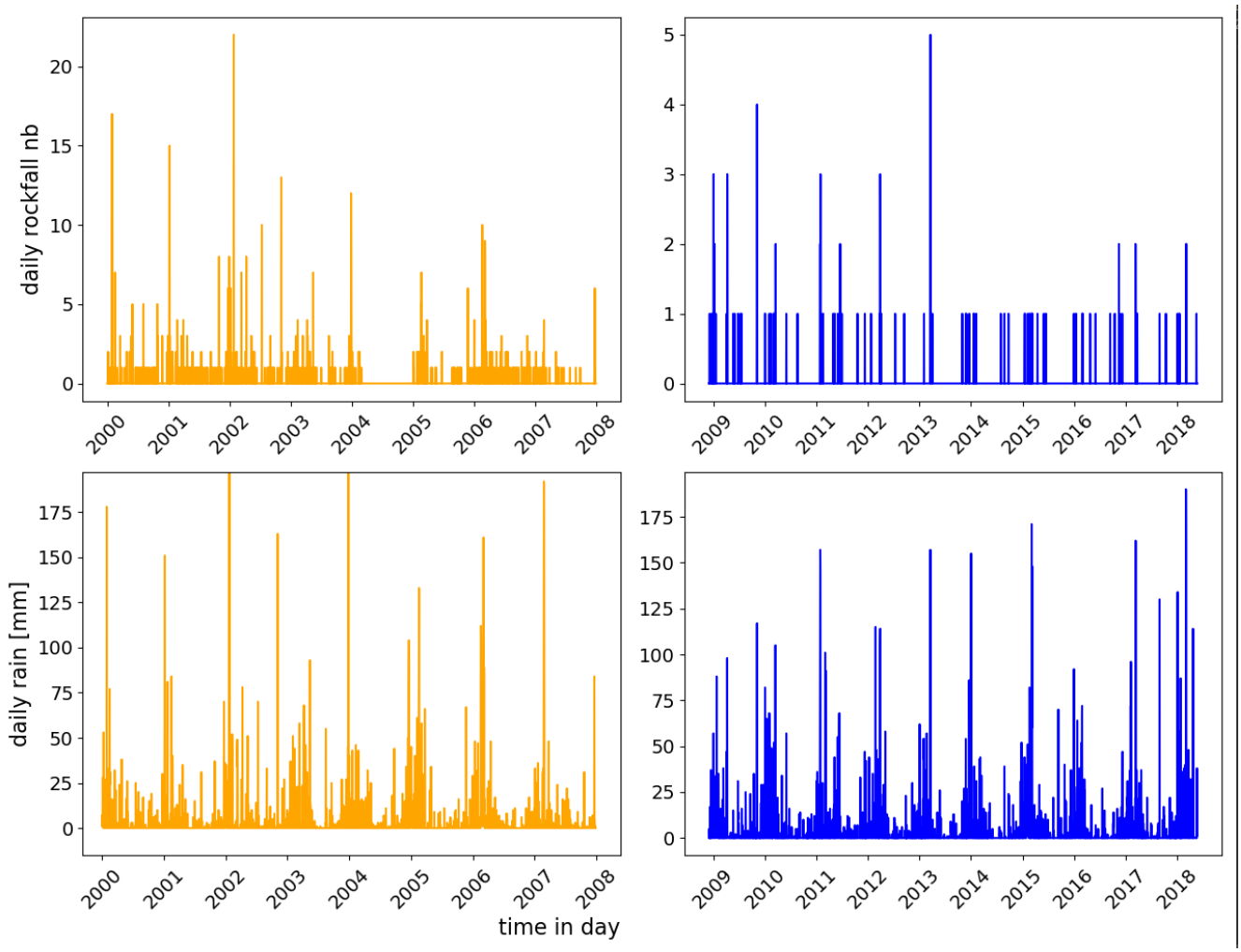

Figure 1.1: Daily time series for the number of rockfall (top) and for quantity of rainfall (bottom) before (left) and after (right) major protection works

1.4 Conclusion

First analysis on RN1 data show:

- a clear change-point in rockfall hazard after major works in 2008. We will focus on data before 2008 because there are much more,

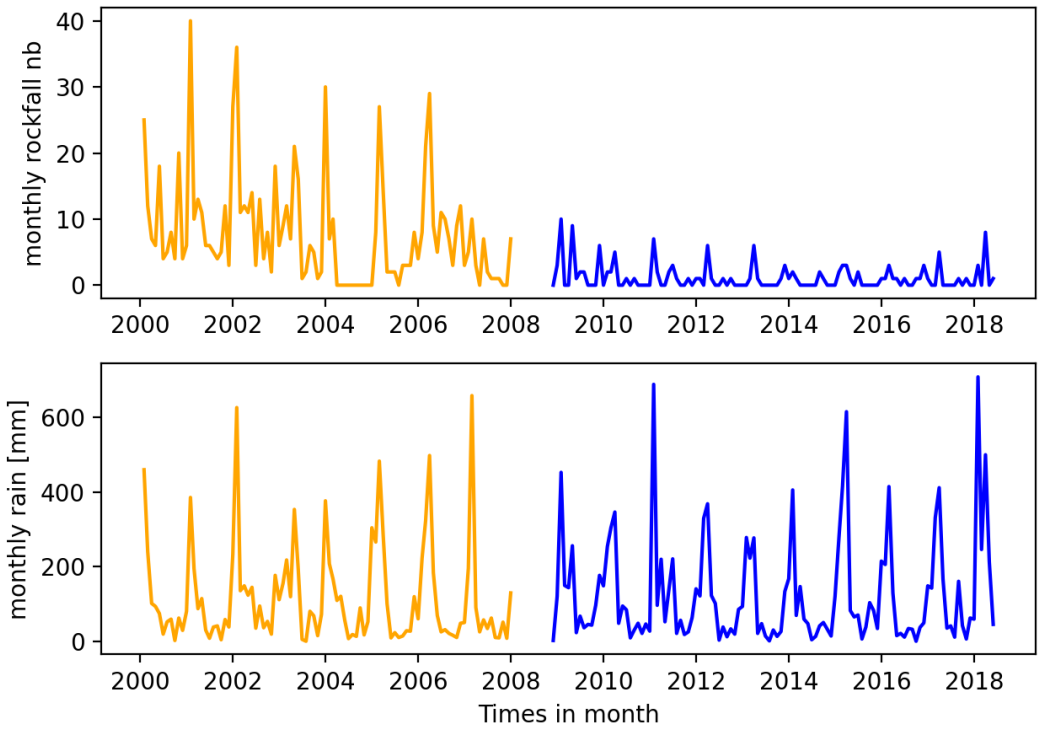

Figure 1.2: Monthly time series for the number of rockfall (top) and for quantity of rainfall (bottom) before (left) and after (right) major protection works

- point process of rockfall times is not an homogeneous Poisson Process, as time between event doesn't follow exponential law,
- **strong seasonality** in rainfall times series and quite strong in rockfall one,
- significantly **correlation between rockfall and rainfall** from day-1 to round day-5

Some perspectives for future works could be:

- **rainfall episode** in[2] belong to factors which can explain a part of rockfall hazard and should be take into account in future predictive model,
- in [2], consider other variable dimension like : intensity of rain in $mm.h^{-1}$, rockfall frequency in h^{-1},
- consider **threshold of event size** ($> 0.1m^3$) because dependencies and dynamics for small or big rockfall seem to be not the same.

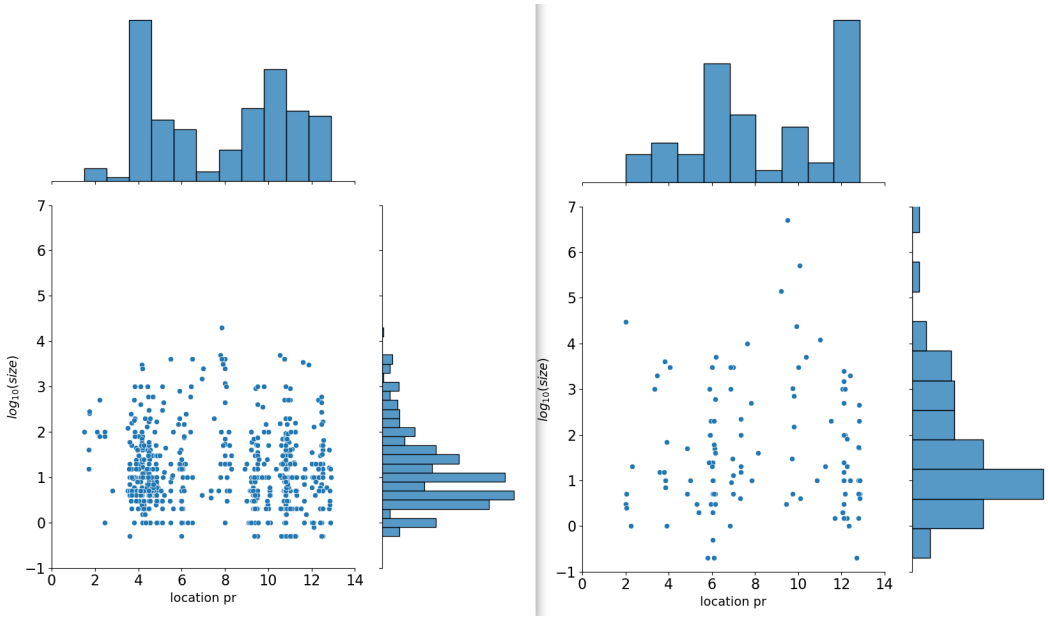

Figure 1.3: Monthly time series for the number of rockfall (top) and for quantity of rainfall (bottom) before (left) and after (right) major protection works

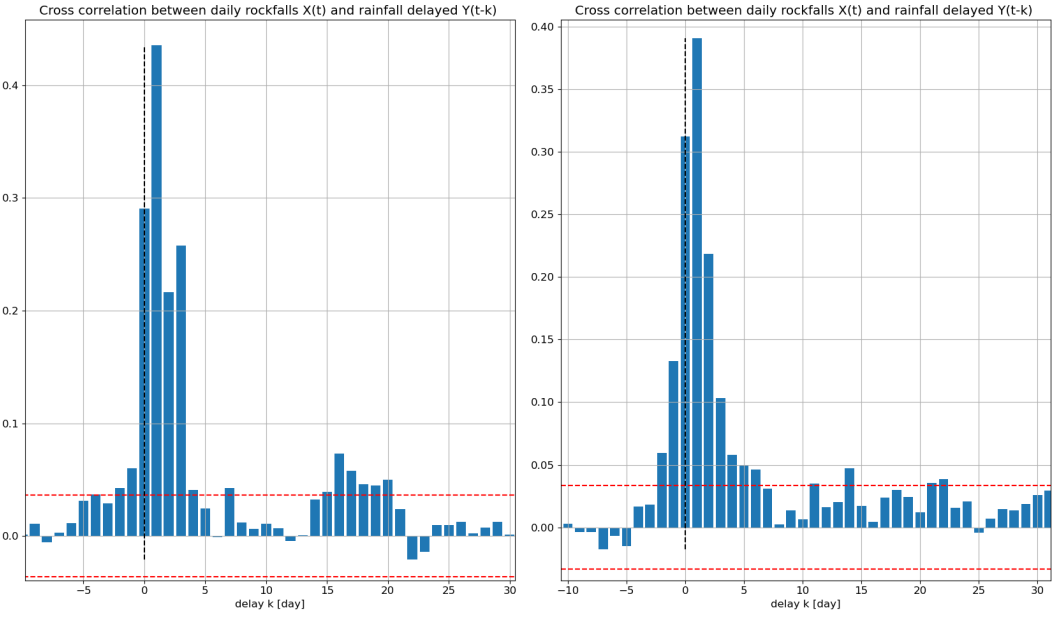

Figure 1.4: Monthly time series for the number of rockfall (top) and for quantity of rainfall (bottom) before (left) and after (right) major protection works

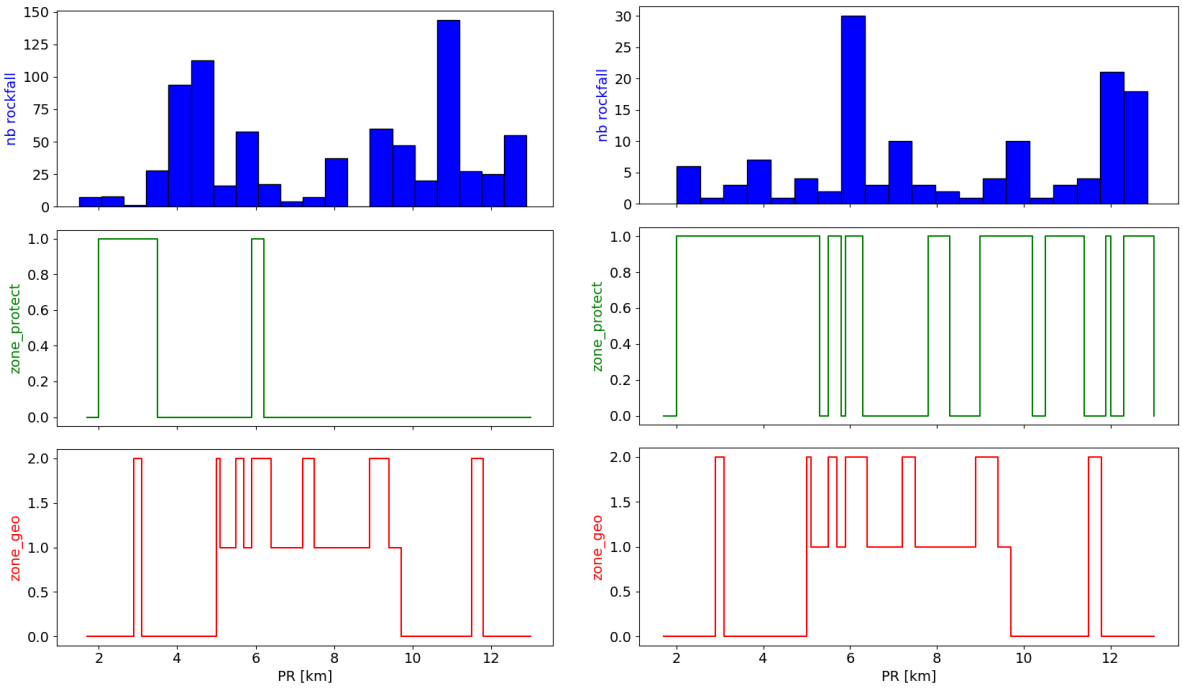

Figure 1.5: Monthly time series for the number of rockfall (top) and for quantity of rainfall (bottom) before (left) and after (right) major protection works

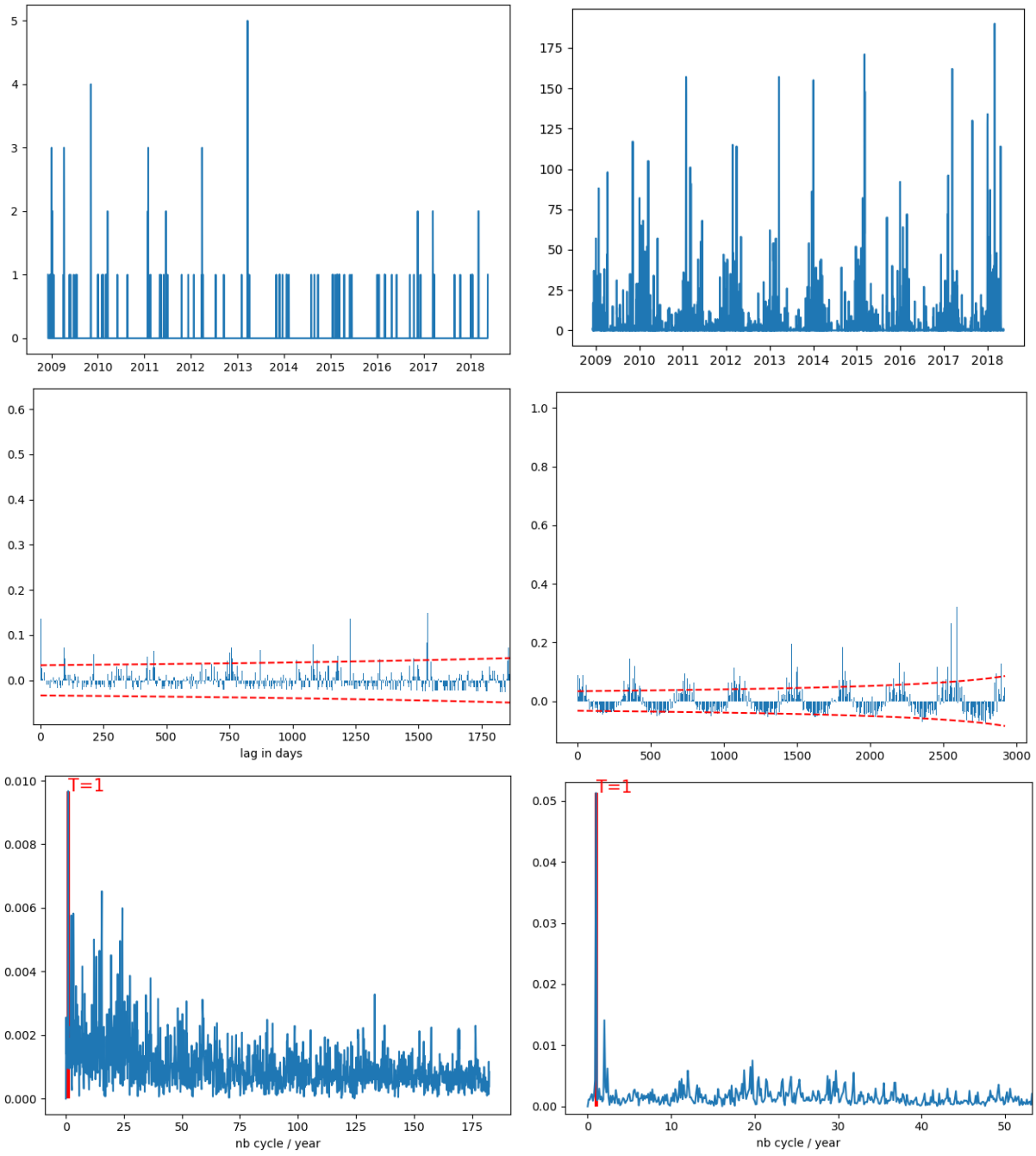

Figure 1.6: At top, Daily time series (on left) for the number of rockfall and for quantity of rainfall (on right). In the middle, sample auto-correlation with significance limits. At bottom, correlogram , spectral counter-part of auto-correlation

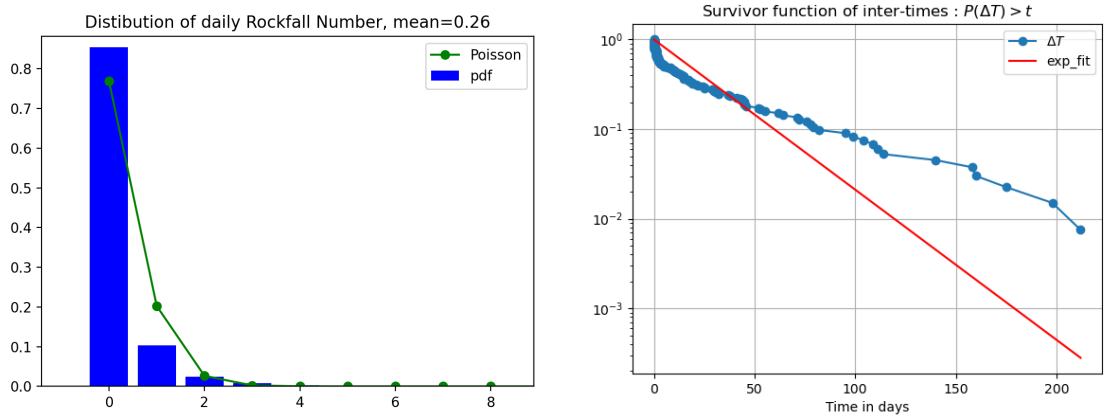

Figure 1.7: Distribution of inter-times: a) density function on left, b) inverse-cumulative density function on right. All with an exponential fit with mean $\simeq 25$

Chapter 2

Toward Formalisation

We can study rockfall phenomenon at **3 different level**:

1. **at temporal level**, where events are spatially aggregated. Available variable are daily times series of rain and of the number of rockfall.

2. **at spatial level**, where events are temporally aggregated. Available variable are the number of rockfall, by binned location, protection zone and geological zone.

3. **at spatio-temporal level**, with all information. Available variable are daily times series of rain, but also size of event, its location PR. This latter give extra information like geological zone and protection zone. As resolution is better, the size of dataset should be much more than here (less than 1000 rockfall),

Remark. *At temporal level, when we aggregate spatially event. We could ask if there is a sense for other variables to sum size, to average location ?*

Different point of view : Mostly depending of sampling time of events, we can consider different mathematical objects to model for this phenomenon if the amount of data allow it:

1. When events occurs at irregular times, we speak about **point process** (with only time event, or with location and exogenous variable and size as markers). For these process, many recent developments have been done with neural network . **Counting process** are naturally associated with point process and they permit to view point process at a regular sampling.

2. When events occurs at regular times, **time series** are the more natural way to see data, like popular ARMA models with Auto-Regressive and Moving Average part. These process are linear but we could add non-linearity like in Non-linear Auto-Regressive process (NAR) and exogenous variable (NARX process).

Here, we present 2 types of machine learning model, which have been tested in this work:

- **Regression models** where we only focus on the dependencies between rockfall and rainfall. Those models don't take into account time but we can introduce lagged time variable as exogenous one. If target values are binary, then we call it logistic regression.
- **Neural Networks** with plenty of interconnected neurons and specifical functions. Due to sparsity of dataset, we investigate only Supervised Learning. In that case, networks ability to map inputs to target is learnt from labelled data.

2.1 Regular/Irregular sampling of events

2.1.1 Irregular sampling: Point process

Sometimes, a physical phenomenon is just seen at irregular points (for d-dimensional space) or times, and can be modelled by a point process. In temporal dimension, a sample from point process is a set of increasing times $(T_i)_{i=1,\dots,N}$, each $T_i \in \mathbb{R}^+$. This sample can be encode in a **counting process** $N(.)$:

$$\begin{array}{rccl} N: & \mathbb{R}^+ & \longrightarrow & \mathbb{N} \\ & t & \longmapsto & N(t) = \sum_i \mathbb{1}_{T_i<t} \end{array}$$

$N(t)$ represent the number of events occurring between time interval $[0, t[$. So, $N_{[a,b[} = N(b) - N(a)$ is number of points in $[a, b[$.

Let $\lambda(.)$ a integrable function defined on $\mathbb{R}^+$, named **intensity**. Mean of the random number of events can be computed from this intensity: by

$$\mathbb{E}[N(t)] = \Lambda(t) = \int_0^t \lambda(s)ds$$

Definition 1. *Point process is call **Inhomogeneous Poisson Process** with rate $\lambda(t)$ if :*

1. *$N(0) = 0$ and $N(t)$ has independent increments, i.e $N_{[a,b[}$ is independent from $N_{[c,d[}$ as soon as $[a, b[$ and $[c, d[$ are disjoint intervals.*
2. *$\mathbb{P}[N(t + h) - N(t)] = \lambda(t)h + o(h)$. Events occurs at time t with a the rate $\lambda(t)$. Moreover it should be is a simple process (no more than one point at each time, almost surely)*

Compared to general case, Poisson Process has independent increments! Such process are called "Poisson" because they have the following property.

Property. *The number of events in any interval $[t, t + h]$ is a Poisson random variable $\mathcal{P}(\mu)$ with mean parameter μ depending on the location t and the width h of the interval:*

$$\mu = \mathbb{E}[N(t + h) - N(t)] = \int_t^{t+h} \lambda(s)ds$$

is the expected number of events occurring between times t and $t+h$

Example. *The most basic example of a temporal point process is a* ***homogeneous Poisson process****, which assumes that the events are independent of each other and events occurs at a constant rate i.e $\lambda(t) = \lambda$. In that case, times between events $\Delta T_i = T_i - T_{i-1}$ follow an exponential law.*

Example. *Another popular example is the* ***Hawkes process****, which is a simple model of a self-exciting point process. The intensity of the Hawkes process is given by $\lambda(t|H_t) = \mu + \sum_{t_i<t} g(t-ti)$, where $g(.)$ is a kernel function ($g(s) = 0$ if $s < 0$) that represents the triggering effect from the past event*

Application to rockfall:
As inter-event times distribution doesn't follow an exponential law (previous chapter), Rockfall events is not a homogeneous Poisson Process.

As previous basic statistics seem to show, rockfall event is at temporal level a point process with an inhomogeneous seasonal intensity rate $\lambda(t)$ (probability that a rockfall occur in time interval $[t; t+dt[$). This intensity is correlated with rainfall point process which have a strong annual seasonality.

We could model a point process with spatio-temporal intensity function $\lambda(t, x)$ where variation among x could be explained by protection, geological zone... As general for these point process, we can define $N(t, x)$ is the number of event between time interval $[0, t[$ between location $[0, x[$. $N(t, x)$ can be expressed from an 2D-intensity $\lambda(t, x)$ which can be seen as instant rate of rockfall at time t and at location x.

2.1.2 Regular sampling: Times Series

In this framework, the quantity of interest $X_t = (X_i)_{i\in\mathbb{Z}}$ is evenly sampled each Δt, generally in time domain, so that i-th value is $X_i = X_t[i\Delta t]$.

Definition 2. *X_t is said* ***second-order stationary*** *if :*

- $\mathbb{E}[X_i]$ *is constant over time and named mean of process $\mu = \mathbb{E}[X_i]$*
- $\mathbb{E}[X_i X_{i+h}]$ *doesn't depend of considered time i but only of time delay $h \in \mathbb{Z}$. We call auto-covariance:*

$$\gamma(h) = \mathbb{E}[(X_i - \mu)(X_{i+h} - \mu)]$$

Remark.
- *Usually we prefer its normalised version named auto-correlation $\rho(h) = \frac{\gamma(h)}{\gamma(0)}$. $\gamma(h)$ and $\rho(h)$ are symmetric i.e $\gamma(-h) = \gamma(h)$ for $h \in \mathbb{Z}$.*
- *Its spectral domain counterpart is the Power Spectral Density $\Gamma(\nu)$ through Fourier Transform .*

- *For our application, times series which values X_i lie in $\mathbb{N}$, they are generally named count times series.*

If we have 2 times series X_t and Y_t, like rainfall an rockfall daily times series for us, on can measure (linear) dependencies between them through:

Definition 3. *The cross-correlation between two times series X_t and Y_t, for a given time lag $h \in \mathbb{N}$ is :*

$$C_h(X_t, Y_t) = \frac{\sum(X_t - \overline{X_t})(Y_{t-h} - \overline{Y_t})}{\sigma_X \sigma_Y}$$

where $\sigma_X = \sqrt{\sum(X_t - \overline{X_t})^2}$ is the standard deviation of variable X

2.2 Regression models

These models regress a target variable Y against explicative variables $\mathbf{X} = [X_1, X_2, ..., X_p]$ without taking explicitly time in consideration (like a succession of samples without order). General Linear Model can be formulated as :

$$\mathbb{E}[Y|\mathbf{X}] = g(\mathbf{X}\beta)$$

where β are coefficients vector, $g(.)$ is the link function adapted to the probability law of Y. For instance, if $Y \in \{0, 1, 2...\}$ has Poisson distribution, g is logit ; if $Y \in \{0, 1\}$ is a Bernouilli variable, g is...

Example. *In next chapter, we first try regression with previous rain and number of rockfall as regressors. If NB_t stands for the number of rockfall and R_t for rainfall quantity at day t (with lag time or cumulative specified before), then regression vector is $\mathbf{X} = [R_t, R_{t-1}, ..., R_{t-p}, NB_{t-1}, ...NB_{t-k}]$. Number of lags p and k should be determined by different statistical test.*

2.3 Neural Networks (NNs)

From a statistical point of view, NNs are just non-linear function estimated $f_\theta(.)$ from data, that attempt to map inputs to target $\hat{Y} = f_{\hat{\theta}}(X)$. Contrary to classical estimators, the number parameters $card(\theta)$ is often around million. They can be tuned thanks to their structure and effective parallel computing.

2.3.1 Learning process

We say that a neural network learns from data because of its operating steps. First, all trainable parameters θ of the model, like weights of connection and biases are randomly initialised.

After, the model improves itself to minimize a loss function $L(Y, \hat{Y}) = L_\theta$ that measures the error between predictions and true labels. This is achieved by iterating over each sample of the

train dataset. Starting from model configuration θ_n, we can learn from one sample of labeled data (X, Y) better weights θ_{n+1}:

- **a Forward Pass**: given the input X and the actual model configuration θ_n, a prediction $\hat{Y} = f_{\theta_n}(X)$ is compute,
- **a Backward Pass**: given the label Y, the loss L_θ and its gradient ∇L_θ with respect to all trainable parameters θ are computed. Model configuration θ_n is upgraded, in the opposite direction of gradient to lower loss function (therefore errors between prediction and labels) :

$$\theta_{n+1} = \theta_n - lr \nabla L_\theta$$

where lr, the learning rate is a scaling factor, tuned during learning.

These two points, forming a learning step, are repeated over the entire dataset and are called a training epoch.

After one epoch, model capabilities are evaluated on the validation dataset. Depending on the results, some hyper-parameters could be adjusted like the learning rate.
To learn correctly, we repeat many epochs to reach a good predictive model. There are many parameters that can be optimized for each problem like the structures for the intermediate layers, changing the activation function, the error function and also the learning rate.

2.3.2 Forward Propagation: from input to predictions

To better understand their structure and how they works, we will illustrate it through a simple Dense Neural Network (DNN). Besides, this example will be tested on RN1 Réunion data in next chapter.

Definition 4. *DNNs are composed by different layers of many neurons, all the neurons from a layer being connected to all neurons from the previous and following layer.*

Example. *Let's formalise a simple classification DNN with one hidden layer with M neurons and one last layer with K neurons the number of classes (see example on Figure 2.1 and scheme of one neuron).*

For inputs $\mathbf{X} = (x_i) \in \mathbb{R}^N$, the j-th neuron h_j from dense hidden layer integrate in a linear way **all** neurons (x_i) from previous layer:

$$h_j = \sum_{i=1}^{N} w_{i,j} x_i + b_i$$

where:

- $w_{i,j} \in \mathbb{R}$ is the **connection weight** between this hidden neurons and the i-th neuron from previous layer (input),
- $b_j \in \mathbb{R}$ is the **bias**,
- classically h_j is followed by an **relu activation** to to model non linearity: relu(x)=max(0,x).

In a matrix way, hidden layer $H = (H_j) \in \mathbb{R}^M$ is given by:

$$H = W_H X + B_H$$

where $W_H \in \mathbb{R}^{N \times M}$ and $B_H \in \mathbb{R}^M$ are respectively connection weights and biases.

If $W_{last} \in \mathbb{R}^{M \times K}$ and $B_{last} \in \mathbb{R}^K$ denote weights and biases of last dense layer, output vector is $W_{last} \cdot relu(H) + B_{last}$ (with relu point-wise extended). Classically, this last layer is followed by a $softmax(.)$ activation define from $\mathbb{R}^K$ to simplex $[0,1]^K$ by:

$$softmax(c_i) = \frac{exp(c_i)}{\sum_{i=1}^{K} exp(c_i)}$$

that transform output vector to probability vector $\hat{Y}$ on the K classes ($\forall\ 1 \leq i \leq K,\ \hat{Y}_i \in [0,1]$ and $\sum_{i=1}^{K} \hat{Y}_i = 1$),

This give the **Forward Propagation** from input $\mathbf{X}$ to $\hat{Y}$:

$$\hat{Y} = softmax(W_{last} \cdot relu(WX + B) + B_{last})$$

The predicted class is finaly the maximum argument of this output vector $\text{argmax}_{1 \leq i < K}(Y_i)$.

In that case, trainable parameters are $\theta = [W_H,\ B_H,\ W_{last},\ B_{last}]$, and the mapping between inputs and label is the parametric non linear function $\hat{Y} = f_\theta(X)$

Example. *As example, if model compute* $\hat{Y} = [0.2,\ 0.1,\ 0.7]$ *then predicted class is the last class "3".*

2.3.3 Backward Propagation: from loss function to weights update

Given a labelled data sample (X, Y), to modify appropriately model weights θ, one should measure errors between prediction $\hat{Y}$ and true label Y, through a loss function $L(Y, \hat{Y})$. As $\hat{Y} = f_\theta(X)$, for a given input X and Y is known, loss became a function of weights θ.

As we mention before, an simple gradient descent (like in optimizer SGD) $\theta_{n+1} = \theta_n - lr \nabla L_\theta$ permit to decrease iteratively loss function.

For classification task in our tested networks , we used the classical Cross-Entropy for the loss function.

$$L(Y, \hat{Y}) = L_\theta = -\sum_{i=0}^{K} Y_i \log \hat{Y}_i = -Y^T \cdot \log(\hat{Y})$$

where Y is the true label one-hot encoded and $\hat{Y}$ is the model prediction as above.

Example. *To follow previous example, if models predicts class "3" with* $\hat{Y} = [0.2,\ 0.1,\ 0.7]$ *and GT class is "3" then* $Y = [0, 0, 1]$ *and then* $L = -1 \times \log(0.7)$

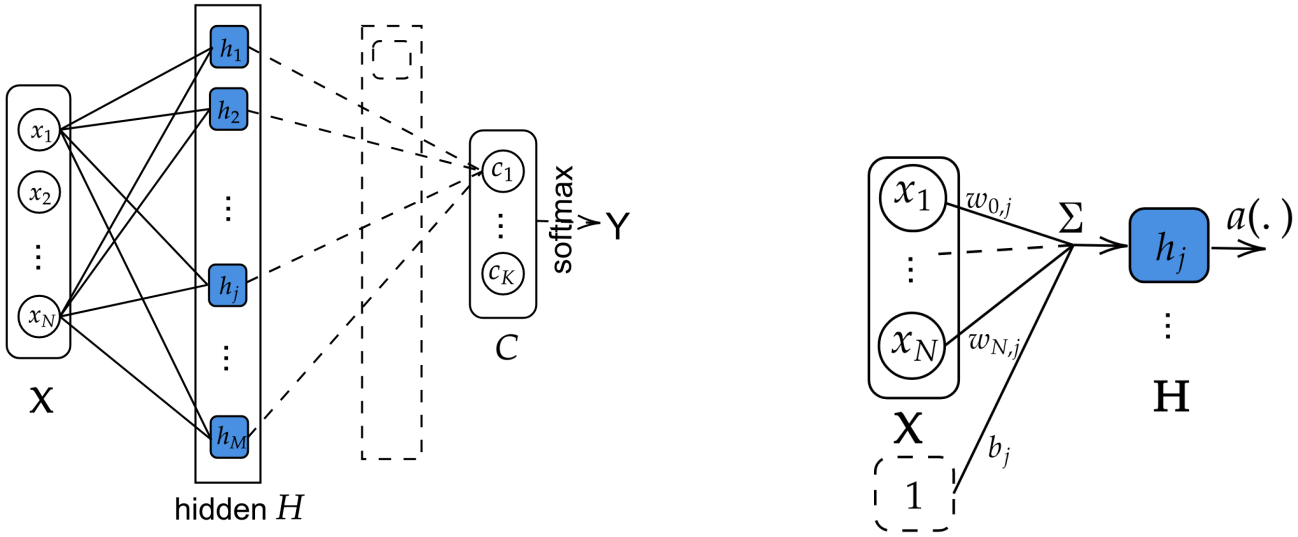

Figure 2.1: On left, scheme of DNNs used with one (or more) hidden layers. On right, focus on one neuron h_j in hidden layer.

2.4 Conclusion

Here, we introduce some theoretical point around point process and time series that can be interesting for our application. Some models was presented, in particular we precise how we train a neural network from labelled data (supervised learning).

Rockfall event can be naturally modeled by point process in temporal/spatial domain. As RN1 data are sparse, we aggregate spatially data and look at temporal dynamics (previous chapter): a point process with an inhomogeneous seasonal intensity highly correlated with rainfall point process. Therefore, quantity of interest would be event time (T_i) with some features (sum of size ?...). Such point processes can be modeled by neural networks, see Annex B for a recent bibliography.

In next chapter, we focus on simple models which target is the daily number of rockfall NB_t with sometimes transformation on targets.

Chapter 3

Experiments

Here, we present different models tested on Réunion Data. As we have sparse data, we just investigate models at temporal level. We first present operational rules used during period 2000-2020. Those expert rules applied on data spanning over the same period, give us a baseline for comparing 3 types of models with different variations:

- Neural Networks : Dense ones (DNNs) or Convolutional ones (CNNs)
- Regression models : General Linear Model (GLM)

In all cases, targets are the daily number of rockfall NB_t and inputs are previous rain quantities R_t (and sometimes previous number of rockfalls) .

3.1 Experiments settings

3.1.1 Train/validation settings

As events are more frequent before protection works, we focus on the first part of data from 2000 to 2007 (Table 3.1).
Split Train/Val: For this period, rockfall data are absent during year 2005. So we split data into train/val set around this year (See Table 3.1 for precision). Such split is permit because rockfall event are quite stationary in time.

3.1.2 Grouping label for classification

As data are sparse, it would be very hard to predict exactly the number of rockfall at day t given the history. In first attempt, we group labels (which span from 0 to K_{max} over the whole history) for classification task:

- 2 classes with '0' no event, '1' one or more event: a simple binary model that math exactly expert rules

	period	Nb events	Nb sample	0	1	2
train	2000-2004	281	1512	81%	13%	5%
val	2005-2007	142	1046	87 %	9%	4%
X	2008-2018	134	3462	97 %	2%	1%

Table 3.1: 3-class Distribution of labels in 2000-2007 dataset in train/val set and in 2008-2018 dataset

- 3 classes with '0' no event, '1' one event, '2' two events: a bit more accurate representation of data

Therefore, targets Y_t for classification networks belong to $\{0, 1\}$ or $\{0, 1, 2\}$

More groups could be considered with more available data. On other hand, we could also consider a quantitative output with regression task (labels are bounded integer and could be normalise by K_{max} to give a probability)

3.1.3 Metrics for classification model

First, let recall classical metrics for classification:

- **confusion matrix** (see example in Table 3.2) to have all information on well , missed classifications.
- **accuracy** which measure the proportion of correct classification among all, $acc = \frac{TP+TN}{TP+TN+FN+FP}$,
- **precision** of each class which measure the proportion of well classified samples among all predicted in this class, $precision = \frac{TP}{TP+FP}$. Poor class precision means that there are many over-detection (FP for binary) in this class,
- **recall** of each class which measure the proportion of well classified samples among all real labelled in this class. In binary classification, recall usually indicate recall of posive class $recall = \frac{TP}{TP+FN}$. Poor class recall means that there are miss-detection (FN for binary) in this class.

When classes are unbalanced, we prefer other metrics. In our binary case, negative samples (0 rockfall) is much more frequent in our dataset.

- **balanced accuracy** is the mean of positive and negative recall, $ba = \left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right)/2$,
- F_β **score** , $F_\beta = (1+\beta^2)\frac{precision \times recall}{\beta^2(precision+recall)}$. When $\beta = 1$, F_1 which weights precision and recall equally, is the variant most often used when learning from imbalanced data. For $\beta > 1$,F_β score put less more weight on recall and more weight on precision if $0 < \beta < 1$

		Predicted 0	1	2
GT	0	300	50	0
	1	5	33	10
	2	0	2	12

		Predicted 0	1	Recall
GT	0	TN	FP	$\frac{TN}{TN+FP}$
	1	FN	TP	$\frac{\mathbf{TP}}{\mathbf{TP+FN}}$
	Precision	$\frac{TN}{TN+FN}$	$\frac{\mathbf{TP}}{\mathbf{TP+FP}}$	

Table 3.2: Confusion matrix for classification: an example for 3 classes on left, the general form for 2 classes (Positive/Negatives and True/False) on right. For 2 class version, row/column normalized versions of True Positives statistic, called Recall or Precision (of positive class)

3.2 Tested Models

3.2.1 Reference: Operating rules after year 2000

Some practical rules, based on rainfall statistics, have been set for RN1 Road. They are based on different threshold for rainfall quantities.

Before major works, we can consider rules for road tipping ($Y_t = 1$) and road opened ($Y_t = 0$) on day t, defined in 2004 by, based on rain quantities R_t :

- if $15 \leq R_t < 30$, then a road tipping for the next day: $Y_{t+1} = 1$
- if $R_t \geq 30$, then road tipping for 3 days: $Y_{t+1} = 1$, $Y_{t+2} = 1$ and $Y_{t+3} = 1$

After major works in 2008, we used other rules defined by :

- if $40 \leq R_t < 50$, then a road tipping for the next day: $Y_{t+1} = 1$
- if $R_t \geq 50$, then road tipping for 2 days: $Y_{t+1} = 1$ and $Y_{t+2} = 1$

	Open	Close	*Recall*
Nothing	872	104	
Rockfall	71	38	**0.35**
Precision		**0.27**	

- Accuracy: 0.84
- Precision: 0.27
- Recall: 0.35
- Balanced accuracy: $0.62 = (0.89 + 0.35)/2$

Table 3.3: Operating Rules defined in 2004 applied on validation set (1085 days from 2005 to 2008)

3.2.2 Dense Neural Networks (DNNs)

First, we test simple DNN with one hidden layer. Targets is the daily number of rockfall NB_t at day t binned in 2 or 3 classes, inputs are previous rain R_{t-1},.. and previous rockfall number NB_{t-1} ...
As class targets NB_t, even grouped in 2 or 3 classes are very unbalanced (see Table 3.1), we try weighting loss by frequency of each class. All tested configuration settings for DNNs are listed below, as complete enumeration of configuration settings is numerous, we test them by subset of parameters.

Different configuration settings for learning NN classifier

DATA PRE-PROCESSING
- *inputs*: rain R_{t-1},..., R_{t-k} with $k \in \{5, 10\}$ or with previous rockfall NB_{t-1},..., NB_{t-k}
- *rescale* in [True, False]: rescale input data X by $\frac{X-mean(X)}{std(X)}$ to center data around 0 where relu activation are more efficient
- *grouping labels* in $K = 2$ or 3 classes

LEARNING HYPER-PARAMETER
- *hidden*: nb of neurons in hidden layer [32; 64]
- *weighting_class* in [True, False] : weighting sample in loss function by their frequency in training set
- *batchsize* in [32, 64, 128] : number of samples fed together in model
- *dropout* in [0, 0.2, 0.4] : dropout in the hidden layer to avoid over-fitting
- *learning_rate* in [0.001, 0.0005, 0.0001] :
- *nb epochs* for training

1st DNN Series: 3 class
FIXED SETTINGS
input: rain with 5 days lag
hidden: 64
epochs: 60

TESTED SETTINGS
batchsize: 32, 64, 128
lr: 0.0001 , 0.0005, 0.001
weighting: True, False
dropout: 0.0, 0.2, 0.4
rescale: True, False

2d DNN Series: 3 class
FIXED SETTINGS
input: rain
epochs: 60
lr: 0.001

TESTED SETTINGS
nb-lags:5, 10, 15
hidden: 32 or 64
weighting: True, False
dropout: 0.0, 0.2
rescale: True, False

3d DNN Series: 2 class	TESTED SETTINGS
FIXED SETTINGS	*input*: rain, rain and nb rockfall
learning rate: 0.001	*nb-lags*:5, 10, 15
batchsize: 64	*hidden*: 32 or 64
epochs: 60	*weighting*: True, False
	dropout: 0.0, 0.2
	rescale: True, False

3-class DNN classifier

You can see in Annex A all experiments results:

- first series (108 trainings) with 3 classes and 5 days rain for inputs, show globally poor results but slightly better with learning rate=0.001 and batchsize=64, and when dropout is less than 0.4.

- second series (46 trainings) start with previous settings (+learning rate=0.001 and batch-size=64), and test for lag of inputs. Globally, we still have poor result. When sample classes are not weighted , we get degenerate confusion matrix, class 1 or 2 is not predict. When we weight classes, class performance are much balanced with 73% accuracy :
  - precision by class [0.90, 0.14, 0.29]
  - recall by class [0.79, 0.29, 0.30] and confusion matrix:

		Predicted		
		0	1	2
	0	737	164	32
GT	1	66	29	4
	2	14	14	15

2-class DNN classifier

To improve performance, we train a third series (with 80 trainings) of binary classifier with different inputs and other hyper-parameter. Results are quite better than with 3 classes, especially when rescale=True, weighting=True, inputs = (rain,nb), lag =(10,10), lr=0.001, batchsize=64.

Two experiments show better results (see Annex A for logs), one with hidden=64, dropout=0 and the other with hidden=32, dropout=0.2. For last one, we get 72% of accuracy with 0.24 for precision, 0.51 for recall and confusion matrix:

		Predicted	
		0	1
	0	710	228
GT	1	69	73

Despite, these networks give many over-detection (no true rockfall predicted as one) which can be seen through the pour precision.

3.2.3 Convolutional Neural Networks (CNNs)

We trained also CNNs. They are famously used to identify patterns on images. Convolution can be used to identify edges, curves and even more complex patterns. The convolution acts like a filter that enhances certain patterns. On images, the convolution kernels, called filters, are matrix while vectors on 1D times series, as for our dataset.

For rockfall events, they could permit to identify temporal pattern in previous rain or rockfall daily series which could lead to rockfall. Contrary to DNNs, link from inputs to target is not fixed by learnt weights but captured through convolution kernels, and probably more flexible. In tested networks, we put one layer of convolution (with different kernel size) before a dense one for classification (see Figure 3.1).

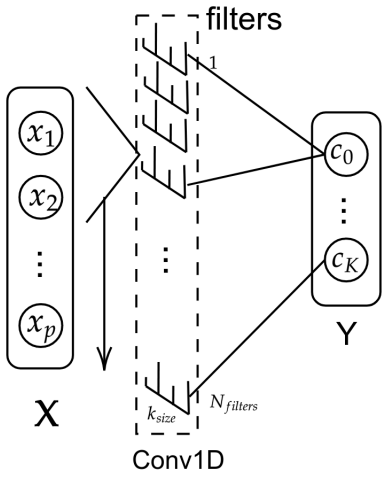

Figure 3.1: Scheme of CNNs used with one convolutional layer. All the N_{feat} feature maps are flattened and gathered in last dense classifying layer. On right, focus on one neuron h_j in hidden layer.

Different configuration settings for learning CNN classifier

CNN Series: 2 class

FIXED SETTINGS
input: rain
learning rate: 0.001
batchsize: 64
epochs: 60
hidden: 64
nb filters: 64 (100 for one)

TESTED SETTINGS
nb-lags: 10,15
kernel size: 3, 5, 9, 13
weighting: True, False
dropout: 0.0, 0.2, 0.4
rescale: True, False

Over 11 tested networks, best results are obtained with 100 convolutional filters of 3 days size. Balanced accuray is 0.65, recall at 0.44 and precision 0.32. Compare to DNN, with quite same level of recall, they have less over-detection (so better precision) and confusion matrix:

		Predicted	
		0	1
	0	805	133
GT	1	80	62

3.2.4 Regression Models: Generalised Linear Model (GLM)

We test some GLM with either Poisson Distribution or Binomial (with normalising target Y by their maximum values). All this fitted models have difficulties to model seasonal activities: bursting period followed by calm one. For example, you can see predictions on train dataset in Figure 3.2 for the estimated model (rain lagged to 5 days) below:

```
==============================================================================
Dep. Variable:                     nb   No. Observations:                 1512
Model:                            GLM   Df Residuals:                     1506
Model Family:                Binomial   Df Model:                            5
Link Function:                  logit   Scale:                          1.0000
Method:                          IRLS   Log-Likelihood:                -92.975
Date:                Mon, 08 Nov 2021   Deviance:                       83.445
Time:                        16:51:28   Pearson chi2:                     149.
No. Iterations:                     7
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
rain1          0.0183      0.005      3.420      0.001       0.008       0.029
rain2          0.0028      0.006      0.515      0.606      -0.008       0.014
rain3          0.0149      0.005      2.759      0.006       0.004       0.025
rain4         -0.0046      0.014     -0.335      0.738      -0.031       0.022
```

```
rain5          0.0033      0.011      0.286      0.775      -0.019       0.026
const         -4.4517      0.246    -18.123      0.000      -4.933      -3.970
==============================================================================
```

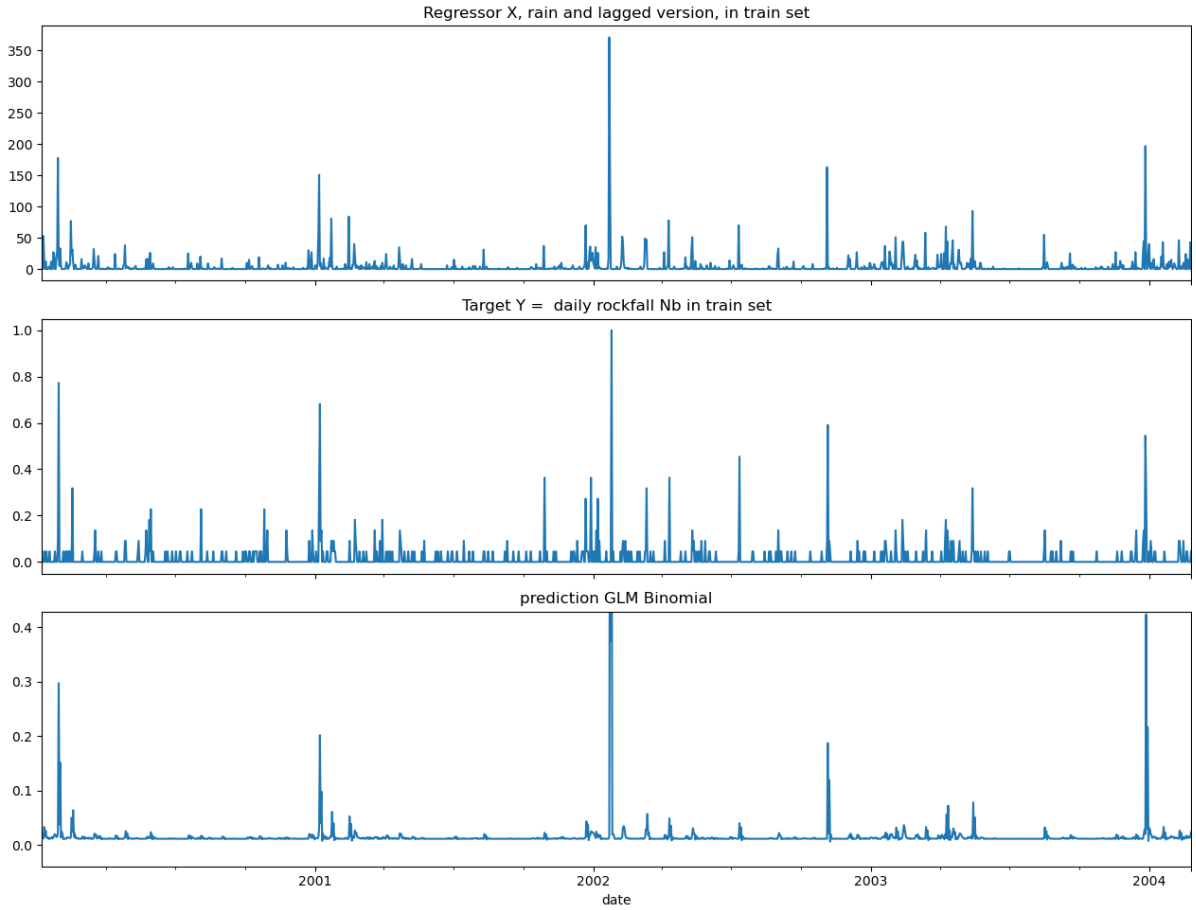

Figure 3.2: GLM regression with Binomial law estimated on period 2000-2004 (train set of first dataset)

3.3 Conclusion

Here, we sum up these preliminary studies to build predictive models for rockfall hazard based on some dependencies like rains quantities. These studies focuses at temporal level without taking into account spatial information (location PR, geological and topological factors ...).

The objective was to investigate different models, their limitations and their performances. Here, we have simplified the initial problem by considering classifying target in 2 or 3 groups. On Table 3.4, we sum up best performances models for each type of relevant models:

- Operational rules standing for baseline is a binary classifier that have a good accuracy (84%) but with less True Positive (TP is rockfall correctly detected) than other models. As classes are unbalanced, it predicts mostly the frequent one (no rockfall). Accuracy is not relevant metric in that case.
- 3-classes problem is much more expressive but in practice, even if classes are weighted, it's difficult to separate positive classes "1" and "2" while keeping good performances.
- 2-classes models have more good predictions (TP) but also over-detection (False positives FP) than Operational Rules. In Table 3.4, we point out the two best 2-classes DNN models, one with only rain as input and other one with rain/number of rockfall. Latter one gives the best recall and TP from all models (54% and 77 TP) but also more over-detections (237 FP)
- 2-classes DNN models shows a good compromise between good-detections (62 TP) and over-detections (133 FP) and so the best balanced accuracy (65%).

Globally, these Neural Networks have slightly better performance than Operational Rules. Nonetheless, all this performances should be seen from an operational point of view because over-detections (FP : predicting rockfall while there is nothing) and miss-detections (FN: predicting nothing while there is a rockfall) have not the same consequences.

After the potential perspectives written in previous chapter conclusion, one could add:

- rockfall hazards seem to be a mix of "natural rockfall event" hardly predictable and "forced rockfall event" caused by exogenous factors (like rain, temperature, thaw-thin cycle..). If size of events could discriminate them (thresholding size), performances would be better.
- dynamic of rockfall is very different in low or high activity period. Here, we consider global model but it could better to investigate switching models (from dry to rainy periods) with their owns parameters.

Model	inputs	recall in %	precision in %	balanced acc in%	accuracy in %	TP	FP	FN	TN
Oper Rules	from 2004	35	27	60	84	38	104	71	872
3-class DNN	rain-10	29-35	14-29	47	73				
2-class DNN	rain-10	42	28	59	78	60	158	82	780
2-class DNN	rain-10 nb-10	54	25	64.5	72	77	237	65	701
2-class CNN	rain-10	44	32	65	80	62	133	80	805

Table 3.4: Best Performance in each type of Neural Networks (DNN, CNN 2 or 3 classes) compared to Operational Rules (TP, FP, FN, TN, standing for True Positive, False Positive, False Negative, True Negative)

Appendix A

Results of series of experiments using DNN

TEST

var_shift	dropout	lr	nb_lags	weighting_class	accuracy	precision1	recall1
('rain', 'nb')	0.0	0.001	-10,1	False	87.0	58.0	11.0
('rain',)	0.0	0.001	-15	False	86.0	40.0	13.0
('rain',)	0.0	0.001	-10	False	87.0	50.0	10.0
('rain',)	0.0	0.001	-15	False	85.0	30.0	11.0
('rain', 'nb')	0.0	0.001	-10,1	False	86.0	44.0	13.0
('rain', 'nb')	0.0	0.001	-5,5	False	87.0	52.0	8.0
('rain',)	0.0	0.001	-5	False	87.0	59.0	11.0
('rain',)	0.0	0.001	-10	False	87.0	46.0	12.0
('rain', 'nb')	0.2	0.001	-10,1	False	87.0	49.0	13.0
('rain',)	0.2	0.001	-10	False	87.0	47.0	14.0
('rain',)	0.0	0.001	-5	False	87.0	47.0	12.0
('rain',)	0.2	0.001	-5	False	87.0	50.0	16.0
('rain', 'nb')	0.0	0.001	-5,5	False	87.0	48.0	14.0
('rain', 'nb')	0.2	0.001	-5,5	False	87.0	52.0	17.0
('rain',)	0.2	0.001	-15	False	86.0	46.0	17.0
('rain', 'nb')	0.2	0.001	-10,1	False	87.0	52.0	16.0
('rain',)	0.2	0.001	-15	False	87.0	55.0	15.0
('rain', 'nb')	0.0	0.001	-10,1	False	87.0	61.0	10.0
('rain', 'nb')	0.0	0.001	-5,5	False	87.0	62.0	9.0
('rain',)	0.2	0.001	-10	False	87.0	57.0	18.0
('rain', 'nb')	0.0	0.001	-10,1	False	87.0	49.0	18.0
('rain', 'nb')	0.2	0.001	-10,1	False	87.0	59.0	12.0
('rain', 'nb')	0.2	0.001	-5,5	False	87.0	51.0	13.0

('rain',)	0.0	0.001	-10	False	88.0	68.0	11.0
('rain', 'nb')	0.2	0.001	-5,5	False	87.0	57.0	11.0
('rain',)	0.0	0.001	-15	False	88.0	68.0	12.0
('rain',)	0.2	0.001	-10	False	87.0	65.0	9.0
('rain', 'nb')	0.2	0.001	-10,1	False	87.0	50.0	17.0
('rain',)	0.0	0.001	-10	False	87.0	63.0	8.0
('rain',)	0.2	0.001	-10	False	88.0	68.0	11.0
('rain', 'nb')	0.0	0.001	-5,5	False	87.0	64.0	10.0
('rain',)	0.2	0.001	-15	False	87.0	67.0	10.0
('rain',)	0.2	0.001	-5	False	87.0	65.0	8.0
('rain',)	0.0	0.001	-15	False	87.0	63.0	12.0
('rain',)	0.2	0.001	-15	False	87.0	63.0	12.0
('rain',)	0.0	0.001	-5	False	87.0	65.0	9.0
('rain', 'nb')	0.2	0.001	-5,5	False	87.0	67.0	8.0
('rain',)	0.2	0.001	-5	False	87.0	51.0	13.0
('rain',)	0.0	0.001	-5	False	88.0	67.0	10.0
('rain', 'nb')	0.0	0.001	-10,1	True	81.0	31.0	37.0
('rain',)	0.2	0.001	-5	False	87.0	65.0	9.0
('rain',)	0.0	0.001	-15	True	78.0	27.0	38.0
('rain', 'nb')	0.0	0.001	-10,1	True	80.0	30.0	38.0
('rain',)	0.0	0.001	-10	True	80.0	31.0	40.0
('rain',)	0.0	0.001	-10	True	79.0	28.0	39.0
('rain', 'nb')	0.0	0.001	-5,5	True	80.0	28.0	35.0
('rain', 'nb')	0.0	0.001	-5,5	True	82.0	30.0	32.0
('rain',)	0.0	0.001	-5	True	82.0	32.0	35.0
('rain',)	0.0	0.001	-10	True	74.0	24.0	46.0
('rain',)	0.0	0.001	-15	True	77.0	26.0	39.0
('rain',)	0.0	0.001	-15	True	73.0	22.0	42.0
('rain',)	0.0	0.001	-10	True	72.0	21.0	40.0
('rain',)	0.2	0.001	-10	True	72.0	21.0	41.0
('rain',)	0.0	0.001	-5	True	82.0	32.0	33.0
('rain', 'nb')	0.0	0.001	-10,1	True	72.0	24.0	51.0
('rain', 'nb')	0.0	0.001	-10,1	True	72.0	24.0	51.0
('rain', 'nb')	0.2	0.001	-5,5	True	78.0	26.0	38.0
('rain', 'nb')	0.2	0.001	-10,1	True	80.0	30.0	41.0
('rain',)	0.2	0.001	-10	True	72.0	21.0	41.0
('rain',)	0.0	0.001	-5	True	76.0	24.0	36.0
('rain', 'nb')	0.2	0.001	-10,1	True	72.0	24.0	51.0
('rain',)	0.0	0.001	-5	True	75.0	23.0	38.0
('rain',)	0.2	0.001	-5	True	79.0	28.0	39.0

('rain', 'nb')	0.2	0.001	-5,5	True	80.0	28.0	37.0
('rain', 'nb')	0.0	0.001	-5,5	True	72.0	23.0	48.0
('rain', 'nb')	0.2	0.001	-10,1	True	72.0	25.0	54.0
('rain',)	0.2	0.001	-15	True	72.0	22.0	43.0
('rain',)	0.2	0.001	-5	True	80.0	29.0	37.0
('rain', 'nb')	0.0	0.001	-5,5	True	71.0	22.0	46.0
('rain',)	0.0	0.001	-15	True	72.0	21.0	42.0
('rain', 'nb')	0.2	0.001	-5,5	True	72.0	23.0	48.0
('rain',)	0.2	0.001	-5	True	73.0	22.0	40.0
('rain',)	0.2	0.001	-5	True	75.0	23.0	38.0
('rain',)	0.2	0.001	-10	True	80.0	30.0	40.0
('rain', 'nb')	0.2	0.001	-10,1	True	78.0	28.0	42.0
('rain',)	0.2	0.001	-15	True	76.0	24.0	39.0
('rain', 'nb')	0.2	0.001	-5,5	True	72.0	22.0	45.0
('rain',)	0.2	0.001	-15	True	71.0	21.0	43.0
('rain',)	0.2	0.001	-15	True	77.0	26.0	41.0
('rain',)	0.2	0.001	-10	True	78.0	28.0	42.0

Table A.1: First Series of experiments for DNN with 3 classes

---

```
->binary_class-bsize:64-epochs:60-lr:0.001-hidden_l:64-weighting_class:True-dropout:0
-var_shift:('rain','nb')-nb_lags:(10,10)-rescale:True-date:13_07_08_31
All params
-> bsize : 64
-> epochs : 60
-> lr : 0.001
-> hidden_l : 64
-> weighting_class : True
-> dropout : 0
-> var_shift : ('rain', 'nb')
-> nb_lags : (10, 10)
-> rescale : True
-> date : 13_07_08_31
Model: "binary_class"
_________________________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================================
input_17 (InputLayer)        [(None, 20)]              0
_________________________________________________________________________________
```

```
dense_16 (Dense)                (None, 64)                          1344
_________________________________________________________________________________
dropout_9 (Dropout)             (None, 64)                          0
_________________________________________________________________________________
main (Dense)                    (None, 2)                           130
=================================================================================
Total params: 1,474
Trainable params: 1,474
Non-trainable params: 0
_________________________________________________________________________________
-> confusion matrix:
     0    1
0  704  234
1   70   72
-> accuracy:72.0
-> precision:[91. 24.]
-> recall:[75. 51.]
```

```
->binary_class-bsize:64-epochs:60-lr:0.001-hidden_l:32-weighting_class:True-dropout:0
-var_shift:('rain','nb')-nb_lags:(10,10)-rescale:True-date:13_07_08_30
All params
-> bsize : 64
-> epochs : 60
-> lr : 0.001
-> hidden_l : 32
-> weighting_class : True
-> dropout : 0
-> var_shift : ('rain', 'nb')
-> nb_lags : (10, 10)
-> rescale : True
-> date : 13_07_08_30
Model: "binary_class"
_________________________________________________________________________
Layer (type)                 Output Shape              Param #
=========================================================================
input_12 (InputLayer)        [(None, 20)]              0
_________________________________________________________________________
dense_11 (Dense)             (None, 32)                672
_________________________________________________________________________
```

```
dropout_6 (Dropout)          (None, 32)                              0
_________________________________________________________________________________
main (Dense)                 (None, 2)                               66
=================================================================================
Total params: 738
Trainable params: 738
Non-trainable params: 0
_________________________________________________________________________________
RESULTS on VALIDATION SET

-> confusion matrix:
     0    1
0  710  228
1   69   73
-> accuracy:72.0
-> precision:[91. 24.]
-> recall:[76. 51.]
```

```
Expname:
->conv_binary_class-bsize:64-epochs:60-lr:0.001-hidden_l:64-weighting_class:True
-dropout:0.4-k_size:3-filters:100-pool:False-var_shift:('rain',)-nb_lags:(10,)-
rescale:False-date:18_07_17_50
All params
-> bsize : 64
-> epochs : 60
-> lr : 0.001
-> hidden_l : 64
-> weighting_class : True
-> dropout : 0.4
-> k_size : 3
-> filters : 100
-> pool : False
-> var_shift : ('rain',)
-> nb_lags : (10,)
-> rescale : False
-> date : 18_07_17_50
Model: "conv_binary_class"
_________________________________________________________________________
Layer (type)                 Output Shape              Param #
```

```
=================================================================
input_6 (InputLayer)        [(None, 10)]                      0
_________________________________________________________________
tf.expand_dims           (None, 10, 1)                        0
_________________________________________________________________
conv1d (Conv1D)          (None, 8, 100)                     400
_________________________________________________________________
batch_normalization      (None, 8, 100)                     400
_________________________________________________________________
dropout_3 (Dropout)      (None, 8, 100)                       0
_________________________________________________________________
flatten (Flatten)        (None, 800)                          0
_________________________________________________________________
main (Dense)             (None, 2)                         1602
=================================================================
Total params: 2,402
Trainable params: 2,202
Non-trainable params: 200
_________________________________________________________________
RESULTS on VALIDATION SET

-> confusion matrix:
     0    1
0  805  133
1   80   62
-> precision:[0.91 0.32]
-> recall:[0.86 0.44]
-> accuracy:0.8
-> balanced_accuracy:0.65
-> fbeta_score:0.37
```

epochs	lr	var_shift	bsize	rescale	drop out	Nb lags	weightin g_class	hidden	accuracy	precisi on1	precisi on2	recall1	recall2	avgPrecision	avgRecall
60	0,001	('rain',)	64	FAUX	0,2	(15,)	FAUX	32	87	100	44	1	19	72	10
60	0,001	('rain',)	64	FAUX	0,2	(10,)	FAUX	32	87	17	58	1	26	37,5	13,5
60	0,001	('rain',)	64	FAUX	0,2	(5,)	FAUX	32	87	0	67	0	23	33,5	11,5
60	0,001	('rain',)	64	VRAI	0	(15,)	FAUX	64	87	0	62	0	23	31	11,5
60	0,001	('rain',)	64	FAUX	0,2	(5,)	FAUX	64	87	0	61	0	26	30,5	13
60	0,001	('rain',)	64	VRAI	0	(10,)	FAUX	64	87	0	60	0	21	30	10,5
60	0,001	('rain',)	64	VRAI	0	(10,)	FAUX	32	87	0	58	0	16	29	8
60	0,001	('rain',)	64	VRAI	0,2	(10,)	FAUX	64	87	0	57	0	19	28,5	9,5
60	0,001	('rain',)	64	VRAI	0	(5,)	FAUX	64	87	0	57	0	19	28,5	9,5
60	0,001	('rain',)	64	VRAI	0,2	(5,)	FAUX	64	87	0	57	0	19	28,5	9,5
60	0,001	('rain',)	64	VRAI	0,2	(10,)	FAUX	32	87	0	56	0	21	28	10,5
60	0,001	('rain',)	64	VRAI	0,2	(5,)	FAUX	32	87	0	56	0	21	28	10,5
60	0,001	('rain',)	64	VRAI	0	(5,)	FAUX	32	87	0	55	0	14	27,5	7
60	0,001	('rain',)	64	FAUX	0	(5,)	FAUX	32	87	0	54	0	33	27	16,5
60	0,001	('rain',)	64	VRAI	0,2	(15,)	FAUX	64	87	0	54	0	16	27	8
60	0,001	('rain',)	64	VRAI	0,2	(15,)	FAUX	32	87	0	54	0	16	27	8
60	0,001	('rain',)	64	VRAI	0	(15,)	FAUX	32	87	0	47	0	16	23,5	8
60	0,001	('rain',)	64	FAUX	0	(10,)	FAUX	32	86	17	45	4	21	31	12,5
60	0,001	('rain',)	64	FAUX	0,2	(10,)	FAUX	64	86	11	50	1	23	30,5	12
60	0,001	('rain',)	64	FAUX	0	(10,)	FAUX	64	85	14	43	3	14	28,5	8,5
60	0,001	('rain',)	64	FAUX	0	(15,)	FAUX	32	85	21	31	7	23	26	15
60	0,001	('rain',)	64	FAUX	0	(15,)	FAUX	64	85	12	38	3	26	25	14,5
60	0,001	('rain',)	64	FAUX	0,2	(15,)	FAUX	64	85	0	41	0	26	20,5	13
60	0,001	('rain',)	64	FAUX	0	(5,)	VRAI	64	80	15	27	13	51	21	32
60	0,001	('rain',)	64	FAUX	0	(5,)	VRAI	32	80	13	23	6	63	18	34,5
60	0,001	('rain',)	64	FAUX	0	(10,)	VRAI	32	77	12	21	11	47	16,5	29
60	0,001	('rain',)	64	FAUX	0	(10,)	VRAI	64	77	6	21	4	58	13,5	31
60	0,001	('rain',)	64	FAUX	0,2	(5,)	VRAI	32	75	12	26	17	56	19	36,5
60	0,001	('rain',)	64	FAUX	0,2	(10,)	VRAI	32	75	12	24	17	58	18	37,5
60	0,001	('rain',)	64	FAUX	0,2	(5,)	VRAI	64	75	12	24	15	56	18	35,5
60	0,001	('rain',)	64	FAUX	0,2	(15,)	VRAI	32	75	11	23	13	58	17	35,5
60	0,001	('rain',)	64	FAUX	0	(15,)	VRAI	64	73	14	29	29	35	21,5	32
60	0,001	('rain',)	64	VRAI	0	(5,)	VRAI	32	73	3	14	1	65	8,5	33
60	0,001	('rain',)	64	FAUX	0,2	(10,)	VRAI	64	72	12	22	19	53	17	36
60	0,001	('rain',)	64	FAUX	0	(15,)	VRAI	32	72	13	17	20	49	15	34,5
60	0,001	('rain',)	64	VRAI	0,2	(5,)	VRAI	32	72	10	16	9	63	13	36
60	0,001	('rain',)	64	VRAI	0,2	(5,)	VRAI	64	71	9	17	11	63	13	37
60	0,001	('rain',)	64	VRAI	0	(5,)	VRAI	64	71	8	17	9	63	12,5	36
60	0,001	('rain',)	64	VRAI	0	(10,)	VRAI	32	70	4	15	3	65	9,5	34
60	0,001	('rain',)	64	VRAI	0,2	(10,)	VRAI	32	70	3	14	2	63	8,5	32,5
60	0,001	('rain',)	64	FAUX	0,2	(15,)	VRAI	64	69	10	21	18	49	15,5	33,5
60	0,001	('rain',)	64	VRAI	0	(10,)	VRAI	64	69	5	16	6	65	10,5	35,5
60	0,001	('rain',)	64	VRAI	0,2	(15,)	VRAI	64	67	8	18	14	63	13	38,5
60	0,001	('rain',)	64	VRAI	0,2	(10,)	VRAI	64	67	8	17	13	60	12,5	36,5
60	0,001	('rain',)	64	VRAI	0	(15,)	VRAI	32	67	6	17	9	63	11,5	36
60	0,001	('rain',)	64	VRAI	0,2	(15,)	VRAI	32	67	7	16	10	60	11,5	35
60	0,001	('rain',)	64	VRAI	0	(15,)	VRAI	64	66	8	18	16	63	13	39,5

Table A.2: Second Series of experiments for DNN(3 classes), with fixed parameters in blue, tested ones in green, and metrics

lr	bsize	Weighting class	hidden	rescale	nb_lags	drop out	var_shift	accuracy	precision1	recall1
0,001	64	FAUX	64	FAUX	(10, 10)	0	('rain', 'nb')	87	58	11
0,001	64	FAUX	64	FAUX	(15,)	0	('rain',)	86	40	13
0,001	64	FAUX	64	FAUX	(10,)	0	('rain',)	87	50	10
0,001	64	FAUX	32	FAUX	(15,)	0	('rain',)	85	30	11
0,001	64	FAUX	32	FAUX	(10, 10)	0	('rain', 'nb')	86	44	13
0,001	64	FAUX	64	FAUX	(5, 5)	0	('rain', 'nb')	87	52	8
0,001	64	FAUX	64	FAUX	(5,)	0	('rain',)	87	59	11
0,001	64	FAUX	32	FAUX	(10,)	0	('rain',)	87	46	12
0,001	64	FAUX	64	FAUX	(10, 10)	0,2	('rain', 'nb')	87	49	13
0,001	64	FAUX	64	FAUX	(10,)	0,2	('rain',)	87	47	14
0,001	64	FAUX	32	FAUX	(5,)	0	('rain',)	87	47	12
0,001	64	FAUX	32	FAUX	(5,)	0,2	('rain',)	87	50	16
0,001	64	FAUX	32	FAUX	(5, 5)	0	('rain', 'nb')	87	48	14
0,001	64	FAUX	32	FAUX	(5, 5)	0,2	('rain', 'nb')	87	52	17
0,001	64	FAUX	64	FAUX	(15,)	0,2	('rain',)	86	46	17
0,001	64	FAUX	32	FAUX	(10, 10)	0,2	('rain', 'nb')	87	52	16
0,001	64	FAUX	32	FAUX	(15,)	0,2	('rain',)	87	55	15
0,001	64	FAUX	64	VRAI	(10, 10)	0	('rain', 'nb')	87	61	10
0,001	64	FAUX	64	VRAI	(5, 5)	0	('rain', 'nb')	87	62	9
0,001	64	FAUX	32	FAUX	(10,)	0,2	('rain',)	87	57	18
0,001	64	FAUX	32	VRAI	(10, 10)	0	('rain', 'nb')	87	49	18
0,001	64	FAUX	64	VRAI	(10, 10)	0,2	('rain', 'nb')	87	59	12
0,001	64	FAUX	64	FAUX	(5, 5)	0,2	('rain', 'nb')	87	51	13
0,001	64	FAUX	64	VRAI	(10,)	0	('rain',)	88	68	11
0,001	64	FAUX	64	VRAI	(5, 5)	0,2	('rain', 'nb')	87	57	11
0,001	64	FAUX	64	VRAI	(15,)	0	('rain',)	88	68	12
0,001	64	FAUX	64	VRAI	(10,)	0,2	('rain',)	87	65	9
0,001	64	FAUX	32	VRAI	(10, 10)	0,2	('rain', 'nb')	87	50	17
0,001	64	FAUX	32	VRAI	(10,)	0	('rain',)	87	63	8
0,001	64	FAUX	32	VRAI	(10,)	0,2	('rain',)	88	68	11
0,001	64	FAUX	32	VRAI	(5, 5)	0	('rain', 'nb')	87	64	10
0,001	64	FAUX	32	VRAI	(15,)	0,2	('rain',)	87	67	10
0,001	64	FAUX	64	VRAI	(5,)	0,2	('rain',)	87	65	8
0,001	64	FAUX	32	VRAI	(15,)	0	('rain',)	87	63	12
0,001	64	FAUX	64	VRAI	(15,)	0,2	('rain',)	87	63	12
0,001	64	FAUX	64	VRAI	(5,)	0	('rain',)	87	65	9
0,001	64	FAUX	32	VRAI	(5, 5)	0,2	('rain', 'nb')	87	67	8
0,001	64	FAUX	64	FAUX	(5,)	0,2	('rain',)	87	51	13
0,001	64	FAUX	32	VRAI	(5,)	0	('rain',)	88	67	10
0,001	64	VRAI	64	FAUX	(10, 10)	0	('rain', 'nb')	81	31	37
0,001	64	FAUX	32	VRAI	(5,)	0,2	('rain',)	87	65	9
0,001	64	VRAI	64	FAUX	(15,)	0	('rain',)	78	27	38
0,001	64	VRAI	32	FAUX	(10, 10)	0	('rain', 'nb')	80	30	38
0,001	64	VRAI	64	FAUX	(10,)	0	('rain',)	80	31	40
0,001	64	VRAI	32	FAUX	(10,)	0	('rain',)	79	28	39
0,001	64	VRAI	32	FAUX	(5, 5)	0	('rain', 'nb')	80	28	35
0,001	64	VRAI	64	FAUX	(5, 5)	0	('rain', 'nb')	82	30	32
0,001	64	VRAI	32	FAUX	(5,)	0	('rain',)	82	32	35
0,001	64	VRAI	64	VRAI	(10,)	0	('rain',)	74	24	46
0,001	64	VRAI	32	FAUX	(15,)	0	('rain',)	77	26	39
0,001	64	VRAI	64	VRAI	(15,)	0	('rain',)	73	22	42

Table A.3: Third Series of experiments for DNN(2 classes)

0,001	64	VRAI	32	VRAI	(10,)	0	('rain',)	72	21	40
0,001	64	VRAI	64	VRAI	(10,)	0,2	('rain',)	72	21	41
0,001	64	VRAI	64	FAUX	(5,)	0	('rain',)	82	32	33
0,001	64	VRAI	64	VRAI	(10, 10)	0	('rain', 'nb')	72	24	51
0,001	64	VRAI	32	VRAI	(10, 10)	0	('rain', 'nb')	72	24	51
0,001	64	VRAI	32	FAUX	(5, 5)	0,2	('rain', 'nb')	78	26	38
0,001	64	VRAI	64	FAUX	(10, 10)	0,2	('rain', 'nb')	80	30	41
0,001	64	VRAI	32	VRAI	(10,)	0,2	('rain',)	72	21	41
0,001	64	VRAI	64	VRAI	(5,)	0	('rain',)	76	24	36
0,001	64	VRAI	64	VRAI	(10, 10)	0,2	('rain', 'nb')	72	24	51
0,001	64	VRAI	32	VRAI	(5,)	0	('rain',)	75	23	38
0,001	64	VRAI	32	FAUX	(5,)	0,2	('rain',)	79	28	39
0,001	64	VRAI	64	FAUX	(5, 5)	0,2	('rain', 'nb')	80	28	37
0,001	64	VRAI	64	VRAI	(5, 5)	0	('rain', 'nb')	72	23	48
0,001	64	VRAI	32	VRAI	(10, 10)	0,2	('rain', 'nb')	72	25	54
0,001	64	VRAI	64	VRAI	(15,)	0,2	('rain',)	72	22	43
0,001	64	VRAI	64	FAUX	(5,)	0,2	('rain',)	80	29	37
0,001	64	VRAI	32	VRAI	(5, 5)	0	('rain', 'nb')	71	22	46
0,001	64	VRAI	32	VRAI	(15,)	0	('rain',)	72	21	42
0,001	64	VRAI	64	VRAI	(5, 5)	0,2	('rain', 'nb')	72	23	48
0,001	64	VRAI	64	VRAI	(5,)	0,2	('rain',)	73	22	40
0,001	64	VRAI	32	VRAI	(5,)	0,2	('rain',)	75	23	38
0,001	64	VRAI	64	FAUX	(10,)	0,2	('rain',)	80	30	40
0,001	64	VRAI	32	FAUX	(10, 10)	0,2	('rain', 'nb')	78	28	42
0,001	64	VRAI	64	FAUX	(15,)	0,2	('rain',)	76	24	39
0,001	64	VRAI	32	VRAI	(5, 5)	0,2	('rain', 'nb')	72	22	45
0,001	64	VRAI	32	VRAI	(15,)	0,2	('rain',)	71	21	43
0,001	64	VRAI	32	FAUX	(15,)	0,2	('rain',)	77	26	41
0,001	64	VRAI	32	FAUX	(10,)	0,2	('rain',)	78	28	42

Table A.4: Third Series of experiments for DNN(2 classes), second part

Appendix B

Bibliography about point process related to Deep Neural Network

Only recent works (since 2016) investigated DNN for point process. Several sub-family are used in such task, mainly Recurrent Neural Network (RNN) with an hidden layer that endorse memory through time (papers [5] [8] [9] [11]), NARX model [1], Spiking Neural Network (convolutional network ??) and . Hereafter, we enumerate and sum up some related papers with citation of meaning-full part.

Paper Boussaada 2018 [1]NARX models

"Actually, NARX concept is a nonlinear generalization of the Autoregressive Exogenous (ARX), which is a standard instrument in linear black-box system identification. There are two different architectures of NARX neural network model, series-parallel architecture (named also open-loop) and parallel architectures (named also close-loop) given by equations below (see Figure B.1 for illutration)"

$$\hat{y}(t+1) = F[y(t), y(t-1), ...y(t-n_y), x(t+1), x(t), ...x(t-n_x)] \tag{B.1}$$

$$\hat{y}(t+1) = F[\hat{y}(t), \hat{y}(t-1), ...\hat{y}(t-n_y), x(t+1), x(t), ...x(t-n_x)] \tag{B.2}$$

where $F(.)$ is the mapping function of the neural network, $\hat{y}(t+1)$ is the predicted output of the NARX for the time $t+1$. $\hat{y}(t)$, ..., $\hat{y}(t-n_y)$ are past outputs of the NARX. $y(t)$,..., $y(t-n_y)$ are the true past values of the times series. $x(t+1)$, ..., $x(t-n_x)$ are the inputs of the NARX. n_x and n_y are numbers of input/output delays.

Paper N. Du 2016 [5]

Du et al. [5] proposed to use an RNN to model the conditional intensity function. In this approach, an input vector x_i , which extracts the information of the event time t_i , is first fed into

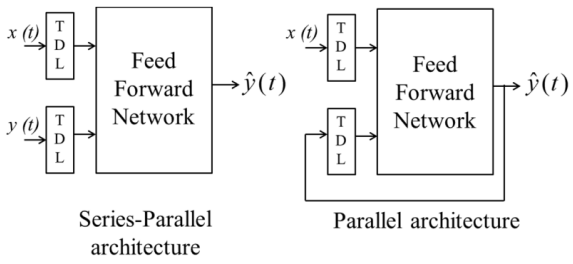

Figure B.1: NARX model illustration from paper [1]

the RNN. A simple form of the input is the inter-event interval as $x_i = (t_i - t_{i-1})$ or its logarithm as $x_i = (log(t_i - t_{i-1}))$. A hidden state h i of the RNN is updated as follows:

$$h_i = f(W^h h_{i-1} + W^x x_i + b^h)$$

where W^h, W^x , and b^h denote the recurrent weight matrix, input weight matrix, and bias term, respectively, and f is an activation function. We here treat the hidden state of the RNN as a compact vector representation of the event history.The conditional intensity function is then formulated as a function of the elapsed τ time from the most recent event and the hidden state of the RNN, given as follows:

$$\lambda(t|H_t) = \phi(t - t_i|h_i)$$

where $\phi(.)$ is a non-negative function referred to as *hazard function* Du et al. (2016) assumed the following form for the hazard function

$$\phi(\tau|h_i) = \exp(w^t\tau + v^\phi \times h_i + b^\phi)$$

The exponential function in the above equation is used to ensure the non-negativity of the intensity. In this model, the conditional intensity function exponentially decreases or increases with the elapsed time τ from the most recent event until the next event *" Since the occurrence of an event may be triggered by what happened in the past, we can essentially specify models for the timing of the next event given what we have already known so far. More formally, a marked temporal point process is a random process of which the realization consists of a list of discrete events localized in time,t_j, y_j with the timing $t_j \in \mathbb{R}^+$, the marker y_j and $j \in \mathbb{Z}^+$ "*
See Figure B.2 for architecture precision on RMTPP model. *" Our key idea is to let the RNN (or its modern variant LSTM [23], GRU [5], etc.) model the nonlinear dependency over both of the markers and the timings from past events As shown in Figure 2, for the event occurring at the time t_j of type y_j, the pair (t_j, y_j) is fed as the input into a RNN unfolded up to the j+1 th event. The embedding h_{j-1} represents the memory of the influence from the timings and the markers of past events. The neural network updates h_{j-1} to h_j by taking into account the effect of the current event (t_j, y_j).The advantage of this formulation is that we explicitly embed the event history into a latent vector space, and by the elegant relation (4), we are now able to capture a general form of the*

conditional intensity function $\lambda^{\star}(t)$ without the need of specifying a fixed parametric specification for the dependency structure over the history "

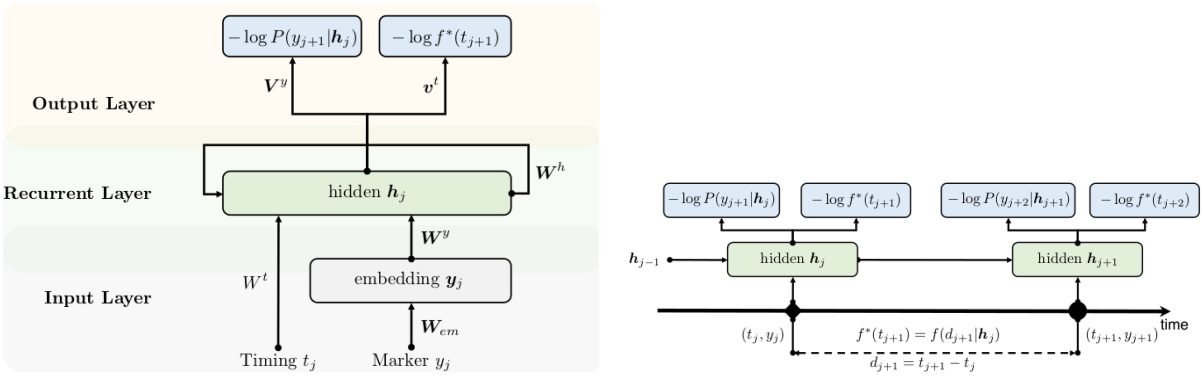

Figure B.2: RMTPP from paper [5], overview on left and zoom on unfold hidden state h_j on right

Paper O. Takahiro 2019 [8]

Specialisation of RNN model proposed by Du et al [5]. source code available *"In this approach, an RNN is used to obtain a compact representation of the event history. The conditional intensity function is then modeled as a function of the hidden state of the RNN. Consequently, the RNN based models outperform the parametric models in prediction performance."*

Paper A. Soen 2020 [9]

it give some theoretical results about point process approximation, some reference about RNN models at most, and propose a RNN model call UNIPoint.

"A known method for approximating a wide variety of functions is the use of neural networks Hornik et al. (1989); Hornik (1991). Recent work has utilised recurrent neural networks to define point process intensity functions Du et al. (2016); Mei and Eisner (2017); Omi et al. (2019); Shchur et al. (2020). A central concept amongst these models is the usage of a recurrent neural network to encode the past events in a sequence, initially proposed by by the Recurrent Marked Temporal Point Process model Du et al. (2016) "

"There are many different types of point processes, and a variety of applications in which they are employed, for example analysing social networks Mishra et al. (2016); Wilhelm et al. (2018). The homogeneous Poisson process is often considered the simplest type of point process, where the number of events over a finite time interval is a random variable with a Poisson distribution Kingman (2005); Daley and Vere-Jones (2007). For the homogeneous Poisson process, the distribution of points is independent of the history and the distribution does not change over time.

The inhomogeneous Poisson process extends the homogeneous version by relaxing the condition that the distribution is fixed over time. A particular type of inhomogenous Poisson process, the Hawkes process, aims to model self-excitation and so has a distribution of points that depends on the history Bacry et al. (2015); Laub et al. (2015) "

"Recurrent Marked Temporal Point Process (RMTPP) Du et al. (2016) was among the first models to employ an recurrent neural network (RNN) to encode the event history and generate parameters defining the intensity function. RMTPP uses an exponential intensity function, a choice which has also been adopted by (Upadhyay et al., 2018). Other methods have employed piecewise constant functions Li et al. (2018); Huang et al. (2019)."

Paper S. Xiao 2017 [12]

" Our model interprets the conditional intensity function of a point process as a nonlinear mapping, which is synergetically established by a composite neural network with two RNNs as its building blocks. See Figure B.3

...

We first make an observation that many conditional intensity functions can be viewed as an integration of two effects: i) spontaneous background component inherently affected by the internal (time-varying) attributes of the individual and the event type; ii) effects from history events "

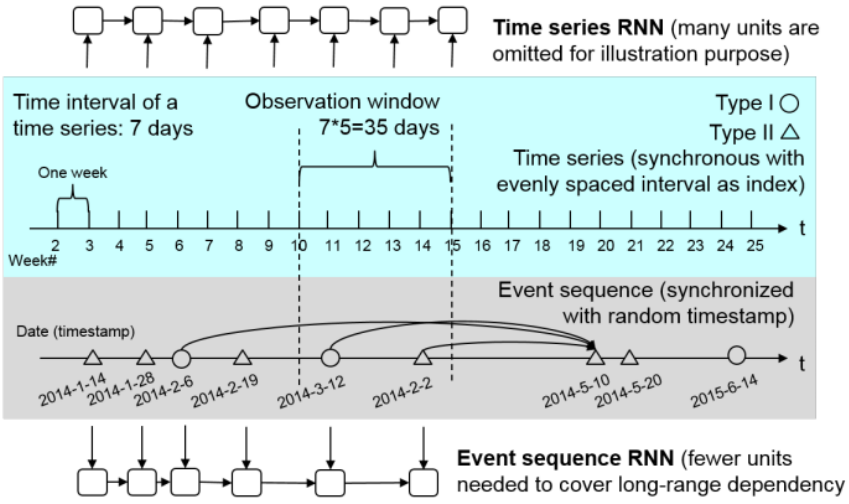

Figure B.3: RNN model from paper [12]

Paper Guen 2020 [6]

No point process modeling .No relevant

A.C Turkmen 2019 [11]

"Temporal point processes (TPP) are probabilistic models of such data, namely discrete event sets in continuous time. They have been extended widely to describe patterns through which events (points) interact, and to model side information available in the form of features (marks)."

" In this paper, we propose a novel model, FastPoint, for efficient learning and approximate inference in multivariate TPP. We combine the expressiveness of RNNs to model mutual excitation (between marks), with well-studied Hawkes processes to capture local (within marks) temporal relationships."

" In Recurrent Marked TPP (RMTPP), Du et al. [6] propose to model a multivariate point process via an approximation to the conditional intensity function. This is achieved by an RNN, in their experiments an LSTM [12]. Effectively, the LSTM embeds the event history $H_t = (t_i, y_i)|t_i < t$ *to a vector, on which the conditional intensity function and the conditional distribution of the mark of the next point are calculated. Concretely, they take the conditional intensity*

$$\lambda^*(t) = \exp(v^T h_j + \beta(t - t_j) + b)$$

where β*,* b *are scalar parameters,* v *is a vector parameter of appropriate dimension.* h_j *is the output of the LSTM for point* t_j *"*

Spiking Neural Network (SNN)

[10] [7] [4] + See website http://www.xavierdupre.fr/app/ensae_teaching_cs/helpsphinx/ml2a/td2a_mlplus_snn.html
A toolbox is available https://snntoolbox.readthedocs.io/en/latest/guide/intro.html

Other Papers to overview

Appendix C

Bibliography

[1] zina boussaada, Octavian Curea, Ahmed REMACI, Haritza Camblong, and Najiba Mrabet Bellaaj. A Nonlinear Autoregressive Exogenous (NARX) Neural Network Model for the Prediction of the Daily Direct Solar Radiation. *Energies*, 11(3), March 2018.

[2] J. D'Amato, D. Hantz, A. Guerin, M. Jaboyedoff, L. Baillet, and A. Mariscal. Influence of meteorological factors on rockfall occurrence in a middle mountain limestone cliff. *Natural Hazards and Earth System Sciences*, 16(3):719–735, 2016.

[3] A. Delonca, Y. Gunzburger, and T. Verdel. Statistical correlation between meteorological and rockfall databases. *Natural Hazards and Earth System Sciences*, 14(8):1953–1964, 2014.

[4] Antonio Di Crescenzo, Maria Longobardi, and Barbara Martinucci. On a spike train probability model with interacting neural units. *Mathematical biosciences and engineering : MBE*, 11:217–231, 04 2014.

[5] Nan Du, H. Dai, Rakshit S. Trivedi, U. Upadhyay, M. Gomez-Rodriguez, and Le Song. Recurrent marked temporal point processes: Embedding event history to vector. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[6] Vincent Le Guen and Nicolas Thome. Probabilistic Time Series Forecasting with Structured Shape and Temporal Diversity. In *NeurIPS 2020*, Vancouver, Canada, December 2020.

[7] Wolfgang Maass. Networks of spiking neurons: The third generation of neural network models. *Neural Networks*, 10(9):1659–1671, 1997.

[8] T. Omi, N. Ueda, and K. Aihara. Fully neural network based model for general temporal point processes. In *NeurIPS*, 2019.

[9] Alexander Soen, Alex Mathews, Daniel Grixti-Cheng, and Lexing Xie. Universal approximation with neural intensity point processes. *ArXiv*, abs/2007.14082, 2020.

[10] Wilson Truccolo, Uri T. Eden, Matthew R. Fellows, John P. Donoghue, and Emery N. Brown. A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *Journal of Neurophysiology*, 93(2):1074–1089, 2005.

[11] Ali Caner Türkmen, Yuyang Wang, and Alex Smola. Fastpoint: Scalable deep point processes. In *ECML/PKDD*, 2019.

[12] Shuai Xiao, Junchi Yan, Xiaokang Yang, H. Zha, and S. Chu. Modeling the intensity function of point process via recurrent neural networks. In *AAAI*, 2017.