Tècniques d'aprenentatge automàtic com a suport per a la gestió del risc d'allaus de neu

Matemàtica Computacional i Analítica de Dades

Universitat Autònoma de Barcelona
Facultat de Ciències

Autor: Àlex Correa Orri

Tutors: Isabel Serra Mochales i Alvaro Corral Cano

Curs 2022-2023

Índex

1 Introducció

Les allaus de neu són fenòmens naturals que es produeixen sovint a moltes zones muntanyoses i presenten una amenaça per a les persones. Per tant, la previsió d'allaus i l'estimació del seu potencial perill són crucials per a garantir la seguretat i mobilitat en aquest terreny.

Durant l'època hivernal les diferents autoritats públiques de cada regió donen informació a la població sobre l'estat del mantell nival a diferents zones i el nivell de perill del desencadenament d'allaus. Avui en dia aquesta informació es presenta en butlletins, confeccionats principalment per experts, que analitzen les condicions sobre el terreny i fent servir dades meteorològiques.

En diversos articles (Edward R. LaChapelle, Jürg Schweizer) s'exposen estudis sobre l'incertesa de les previsions i les probabilitats de que succeeixin aquests esdeveniments, i la dificultat o impossibilitat de mesurar amb precisió el risc real. Tot i això sí que hi ha evidències que mostren que les eines d'aprenentatge automàtic conjuntament amb una bona preparació de les dades, poden oferir informació rellevant per a la gestió del risc d'allaus.

Durant el meu treball tractaré de mostrar evidències de la possibilitat de fer servir eines d'aprenentatge computacional i donar explicabilitat als models i les variables utilitzades. Per fer-ho, prendré com a referència principal l'article `https://doi.org/10.5194/nhess-22-2031-2022`. Aquest estudi descriu el procés d'un enfocament basat en dades, simulacions i classificadors automàtics (Random Forest el més destacat a l'article) per avaluar el nivell de perill d'allaus de neu seca als Alps Suïssos.

En el treball també mencionaré en diferents punts el programa Copernicus. Es tracta d'un programa d'observació i monitoratge de la Unió Europea que proporciona dades de manera oberta per a aplicacions ambientals i de seguretat climàtica. La principal utilitat, en referència a la temàtica i objectius del projecte, és la de poder extreure dades meteorològiques molt específiques de la majoria de les regions d'Europa. Cal destacar que Copernicus és un programa molt important a nivell estratègic per a la Unió Europea, que ha suposat una gran inversió, però pot donar lloc a moltíssimes

fonts d'informació per a la investigació i fins i tot oportunitats de comercialització, sobretot en l'anàlisi del risc.

1.1 Motivació

El meu treball de final de grau i la idea de fer servir un model de *machine learning* per a predir el nivell de perill d'allaus va començar durant el meu segon curs del grau. A l'assignatura d'Anàlisi de Dades Complexes vaig fer un treball, amb tècniques relacionades principalment amb el camp de l'estadística, sobre les diferents característiques que provocaven els diferents tipus d'allaus (anàlisi del pendent, orientacions, longituds, etc.). Al tercer curs, després de cursar l'assignatura d'Aprenentatge Computacional, vaig veure més en profunditat diferents models de regressió i classificació.

Amb aquest context juntament amb la meva afició personal als esports de muntanya vaig pensar que amb les dades suficients (qualitat i quantitat) seria possible fer un model de predicció que "fes el mateix" que fan els experts quan confeccionen els butlletins d'allaus.
Un cop decidit que m'agradaria investigar sobre aquest tema, durant l'estiu de tercer i quart curs vaig fer una gran cerca de conjunts de dades i articles que em fessin veure si era possible o no. Aquesta cerca em va portar a l'article que prenc com a referència.

1.2 Objectius

Donat aquest context els objectius del meu treball són:

- Reproduir els resultats suggerits a l'article esmentat. Aquest primer pas implica ser capaç de descarregar el conjunt de dades amb, per una banda, variables meteorològiques i d'altres simulades amb informació del mantell nival i, per altra banda, el nivell de perill d'allaus. També s'inclou en aquest punt construir un model d'aprenentatge computacional per a obtenir mètriques properes al 70-80% de precisió.

- Donar explicabilitat als models i les variables utilitzades. L'objectiu en aquest pas és el d'entendre com funciona el model i quines són les principals variables per a donar una millor predicció. Aquest punt és decisiu per a la confiança del model.

- Mostrar la possibilitat de realitzar el mateix estudi i resultats fent servir una base de dades extreta directament des de Copernicus (amb el que implica descarregar d'aquest programa i combinar les dades amb les del nivell de perill d'allaus). En aquest últim apartat un dels passos importants és el d'entendre com funciona l'accés a les dades de Copernicus i quin és el nivell de detall i resolució d'aquestes.

2 Procediment per a la quantització del risc d'allaus de neu

Un cop estan clars els objectius del treball serà rellevant donar un context teòric sobre les allaus de neu. Concretament, quins són els seus potencials factors desencadenants, explicar amb detall com són els butlletins d'allaus i com els confeccionen els especialistes.

2.1 Tipus d'allaus i els seus factors desencadenants

Per començar, cal definir què són les allaus: Segons el DIEC (diccionari de l'Institut d'Estudis Catalans), **massa de neu o de glaç que es desprèn i es precipita muntanya avall**. Tot i que aquesta definició és molt simplista, d'allaus de neu n'hi ha de molts tipus i cadascun d'ells es forma i provoca de diferents maneres.
L'EAWS (European Avalanche Warning Services) especifica **5 problemes** diferents:

- **Neu recent**: Relacionat amb la nevada més recent. La magnitud de sobrecàrrega dependrà de la neu nova sobre el mantell preexistent. El problema es presenta de forma generalitzada i sovint en totes les orientacions, fins pocs dies després (quan el mantell s'estabilitza).

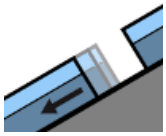
- **Neu ventada**: Relacionat amb el transport de neu pel vent. Sovint es troben plaques formades pel vent a sotavent en canals, depressions, prop de canvis de pendents o darrere de carenes.

- **Capes febles persistents**: Aquest problema està relacionat amb la presència de capes febles persistents en un mantell vell. Habitualment són allaus accidentals, provocats pel pas d'individus i no de manera natural. Les capes febles poden persistir de setmanes a mesos, amb possibilitat que inclús persisteixin durant tota la temporada hivernal. Les capes febles solen ser: gebre de superfície enterrat, gobelets o cristalls facetats.

- **Neu humida**: Es tracta d'un problema associat amb l'afebliment del mantell a causa de la presència d'aigua líquida dins d'aquest. L'aigua s'infiltra a les capes de neu per la fusió (condicionat a la temperatura i/o la radiació solar) o per la pluja. Aquest tipus d'allau es pot desencadenar sense la presència de cap individu i, quan la seva causa és el sol, depèn de l'orientació i altitud.

- **Lliscaments basals**: són aquelles allaus en què tot el mantell nival llisca sobre el terra. Normalment per a donar-se cal que el mantell sigui bastant homogeni o amb poques capes i el terreny sigui llis (superfícies herboses o roca llisa). Aquest fenomen es pot donar amb un mantell fred i sec i amb un mantell càlid

i humit. Generalment es produeixen de manera natural.



2.2 Nivells de perill i butlletins

Considerant tots aquests problemes i factors desencadenants, els experts confeccionen els butlletins d'allaus considerant factors meteorològics, simulacions i prospeccions de neu sobre el terreny. Aquests butlletins es publiquen regularment durant l'època hivernal a la majoria de serralades d'Europa. Tot i que generalment cada regió o àrea té el seu propi butlletí informatiu, mai és prou específic ja que les condicions poden variar a dins d'una mateixa regió (canvi de vall, orientació solar i del vent, tipus de terreny, temperatures, etc.).
Els butlletins d'allaus contenen la següent informació:

- **Nivell de perill**: Aquest és el factor més important del butlletí. S'estableixen 5 nivells de risc associats a la probabilitat i potencials conseqüències de provocar una allau a l'hora de desplaçar-se per terreny d'allaus. Considerant els objectius del treball, aquest serà el principal element a predir.

  1. Feble o Baix
  2. Moderat
  3. Marcat o Considerable
  4. Fort o Alt
  5. Molt fort o Molt Alt

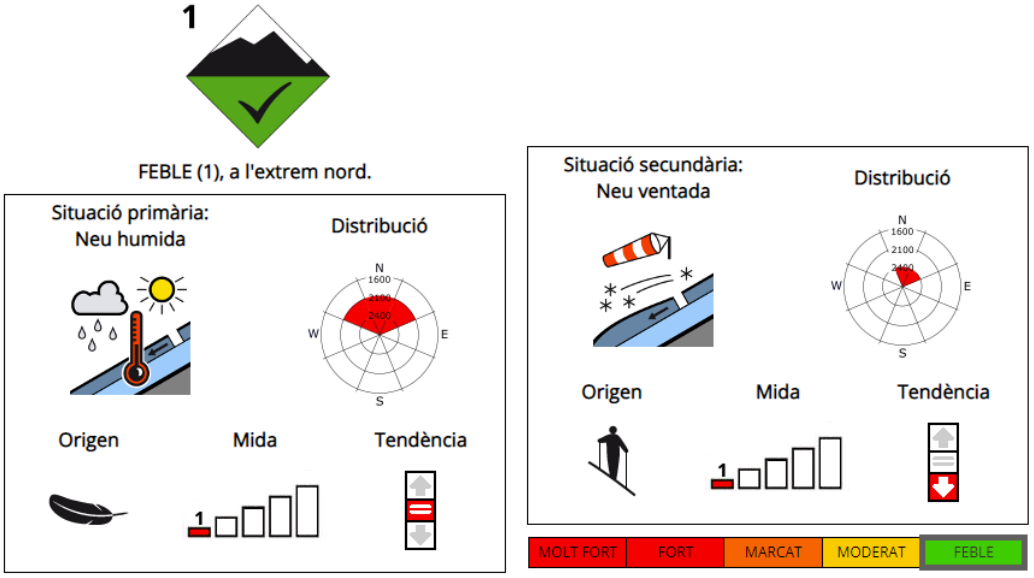
  Tot i l'existència de l'escala establerta, es considera que el risc 0 no existeix (o només existeix en l'absència de neu).

- **Localitzacions propenses a allaus o distribució**: S'indiquen les orientacions i cotes d'altitud a les que les allaus tindran més possibilitats de desencadenar-se.

- **Problema**: Tipus d'allau (veure 2.1)

- **Descripció del perill**
- Altra informació **estat del mantell, meteorologia, etc.**
- **Dades mesurades** per estacions meteorològiques (no verificades)

A continuació es mostra un exemple de butlletí (de l'ICGC):



A aquesta informació se li afegeix, en un text, la descripció de l'estat del mantell nival, la seva distribució i la seva tendència.

3 Fonaments d'aprenentatge computacional

En aquesta secció s'expliquen, de manera teòrica, algunes de les tècniques i models utilitzats durant el meu treball. Principalment, s'especifiquen models, mètriques i altres tècniques relacionades amb l'explicabilitat dels models.

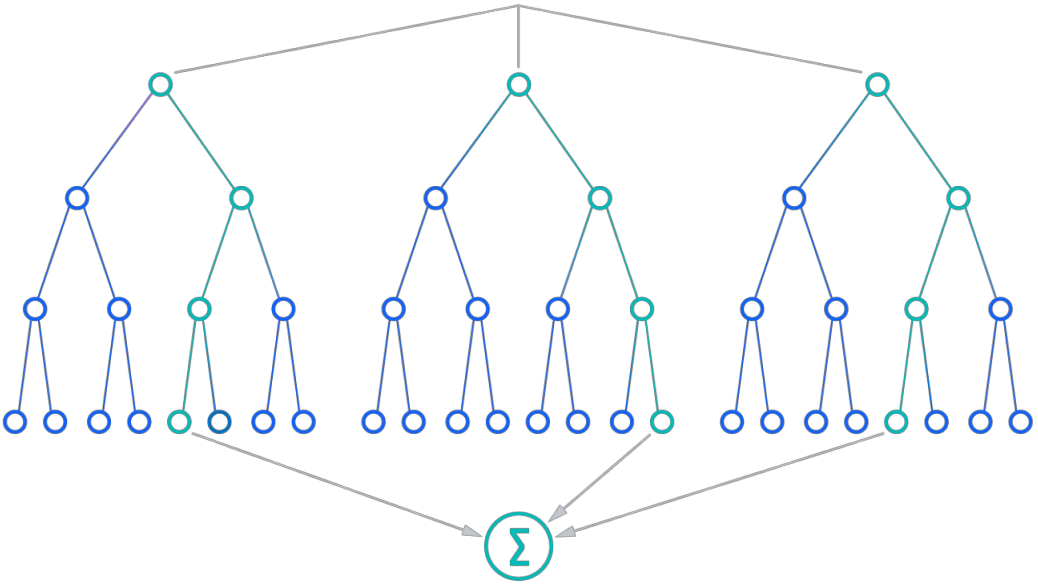
3.1 Models d'aprenentatge automàtic

A continuació explico els models que he fet servir durant la part pràctica del meu treball. Tots són models d'aprenentatge automàtic supervisat que compten amb la seva implementació en Python en llibreries com

Scikit-Learn. I també serveixen per tasques tant de classificació com de regressió.

3.1.1 Random Forest

El Random Forest (RF) és un model d'aprenentatge computacional basat en arbres de decisió que permet realitzar classificació i regressió.
El RF selecciona un subconjunt aleatori de les dades d'entrenament (amb reemplaçament) i un subconjunt de les variables per a cada branca de decisió de l'arbre, utilitzant l'índex *gini* per a establir cada decisió. Aquest procés es repeteix fins al nombre especificat d'arbres a crear.



L'índex *Gini* mesura la probabilitat de classificar de manera incorrecta un element de manera aleatòria. D'aquesta manera, ajuda a seleccionar la millor variable per separar les dades en cada branca de decisió. Aquesta mesura també permet calcular la importància de cada variable al model.

Alhora de crear aquest model, hi ha diferents paràmetres importants a considerar:

- Quantitat d'arbres: Determina el nombre d'arbres de decisió del model. En general augmentar aquest valor donarà un millor rendiment (més robust i menor variància). Tot i això, el fet de tenir més arbres augmenta el cost computacional.

- Màxima profunditat: Especifica la profunditat màxima dels arbres de decisió. Una major profunditat captarà relacions més complexes a les dades, però pot arribar a causar *overfitting*. Això vol dir que el model seria massa bo sobre el conjunt d'entrenament i no donaria bons resultats amb dades diferents (no garanteix la generalització).

- Mostres mínimes per fer divisió: Aquest paràmetre indica el mínim nombre de mostres necessàries perquè un node d'un arbre de decisió faci una divisió. Incrementar-lo pot reduir l'*overfitting*.

- Mostres mínimes per fulla: És el valor que indica el mínim de mostres per a crear un node fulla de l'arbre. Igual que a l'anterior paràmetre, incrementar-lo ajuda a reduir l'*overfitting*. Tot i això, un valor massa elevat pot provocar *underfitting* (el contrari a *overfitting*).

- Màxim de característiques: El RF, durant l'entrenament, per a fer cada divisió selecciona un subconjunt aleatori de característiques. Aquest paràmetre indica el màxim de variables del subconjunt. En general un valor petit ofereix més diversitat d'arbres, però pot portar a una menor precisió.

3.1.2 AdaBoost

Es tracta d'un model que també es basa en arbres de decisió. Generalment, i com a idea principal d'aquest model, es crea un bosc d'arbres que només tenen un node i dues fulles. La diferència amb el Random Forest és que aquest té el mateix pes de decisió per cada arbre mentre que el pes de cada arbre a l'AdaBoost pot variar.
El nom d'AdaBoost ve del concepte *adaptative boosting*. Es parla de *boosting* perquè crea models de manera seqüencial corregint (o reduint) els errors de l'anterior. És a dir, que aquest model fa servir classsificadors "dèbils" (funcionen millor que una predicció aleatòria, però sense una bona precisió) combinats per a obtenir bons resultats.
Els passos que es segueixen per crear el model:

1. Es crea un classificador dèbil (un arbre amb només dues fulles) amb les dades

d'entrenament per a cada variable. S'observa quantes mostres es classifiquen bé a cada arbre.

2. Seguint els resultats del pas anterior, es modifiquen els pesos de cada mostra i classificador per assignar més mostres amb l'etiqueta corresponent. Si un classificador assoleix millors resultats, aquest rep més pes.

3. Es repeteixen els dos primers passos fins que tots els punts estan ben classificats o s'arriba al límit d'iteracions.

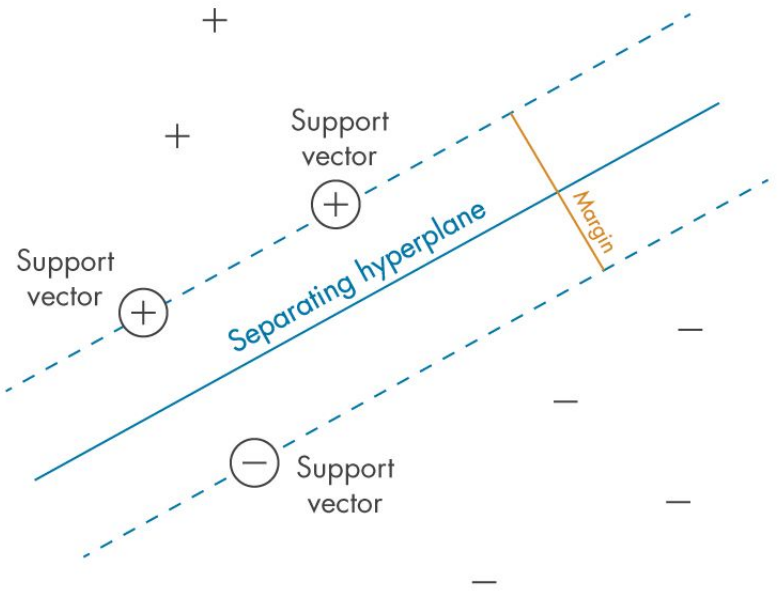
3.1.3 XGBoost

El XGBoost és un model que es basa en arbres de decisió i destaca pel bon funcionament amb grans conjunts de dades i complicats en fer servir diversos mètodes d'optimització. Igual que l'AdaBoost, es basa en el *boosting*, fa servir un procés iteratiu per crear i combinar models febles (arbres de decisió) per formar-ne un de més robust. El procés comença amb un model inicial simple que serveix com a estimació de les classes o valors de regressió. Després, es calcula l'error residual entre les prediccions del model inicial i els valors reals. Aquest error indica la quantitat d'informació que encara no ha estat capturada pel model. Llavors es construeix un nou arbre de decisió que intenta aprendre i modelar aquest error residual. L'arbre de decisió es crea de manera incremental, afegint cada nou node amb l'objectiu de minimitzar l'error restant. A més, s'utilitzen tècniques d'optimització per trobar la millor manera de dividir els nodes de l'arbre amb l'objectiu de reduir l'error. Aquest procés es repeteix diverses vegades.
A més a més, el XGBoost empra un procés de regularització per evitar l'*overfitting*. El que fa per evitar-ho és fer servir un sistema de penalitzacions per reduir la complexitat i el sobreajustament de les dades d'entrenament.

3.1.4 SVM

Aquest és segurament el model més diferent dels altres, ja que no es basa en arbres de decisió. Generalment, treballa bé amb conjunts de dades de dimensions elevades. El SVM o Suport Vector Machine busca crear un hiperplà que separi les dades de

la millor manera possible en les diferents classes o que estimi els valors de regressió. L'objectiu de l'hiperplà és trobar una frontera que maximitzi la distància entre les mostres més properes de les diferents classes. Aquestes distàncies es representen en els vectors de suport.



Un SVM pot triar fer servir d'entre diferents funcions com a transformacions de les dades per facilitar la tasca d'obtenir un hiperplà que maximitzi la separació de les classes. Algunes d'aquestes (també anomenades funció Kernel) són la lineal, la polinòmica, la funció de base radial (o gaussiana) o la sigmoide.

3.2 Validació dels models

Per a verificar si el model prediu adequadament l'objectiu he fet servir les mètriques *Accuracy*, precisió, *Recall* i *F1 Score*. Per a fer la notació de les fòrmules més simple:

- TP: nombre de veritables positius (*true positives*)
- TN: nombre de veritables negatius (*true negatives*)
- FP: nombre de falsos positius (*false positives*)
- FN: nombre de falsos negatius (*false positives*)

3.2.1 *Accuracy*

Aquesta mètrica representa el nombre de prediccions correctes entre el nombre total de prediccions:

$$\frac{TP + TN}{TP + TN + FP + FN}$$

Tot i que aquesta és una de les mesures més utilitzades (i també senzilles d'entendre), cal tenir en compte que no és la millor per a conjunts de dades no balancejats.

3.2.2 *Precision*

Mostra la qualitat del model, quin percentatge de casos que es prediuen de cada classe són realment d'aquella classe.

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

3.2.3 *Recall*

Fa referència a la quantitat de casos predits correctament de cada classe d'entre tots els que són d'aquella classe. Es basa en el concepte d'exhaustivitat, dona informació de la quantitat que el model és capaç d'identificar.

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

3.2.4 *F1 Score*

Ens dona l'informació combinada del *recall* i la precisió fent servir la mitjana harmònica de les dues mètriques. D'aquesta manera en cas que un dels dos valors sigui petit (dolent) llavors el resultat f1 no serà bo.

$$2 \cdot \frac{precision \cdot recall}{precision + recall}$$

3.3 Explicabilitat dels models

L'explicabilitat dels models d'aprenentatge computacional i intel·ligència artificial és crucial per a entendre com funcionen i sobretot per poder confiar en els seus resultats. Sol ser utilitzat per descriure el model,

l'impacte esperat i el seu potencial biaix. S'evita el concepte de "caixa negra".

Les tècniques descrites permeten entendre com el model pren les decisions. Això és molt important ja que en cas d'obtenir resultats no desitjats, hi ha la possibilitat de detectar d'on venen i poder-los evitar. També ajuda a donar suport als resultats o prediccions encertades, serien respostes més robustes.
En concret, per a problemes de classificació, l'explicabilitat dels models és important per a identificar quines són les característiques o variables del conjunt de dades més rellevants.

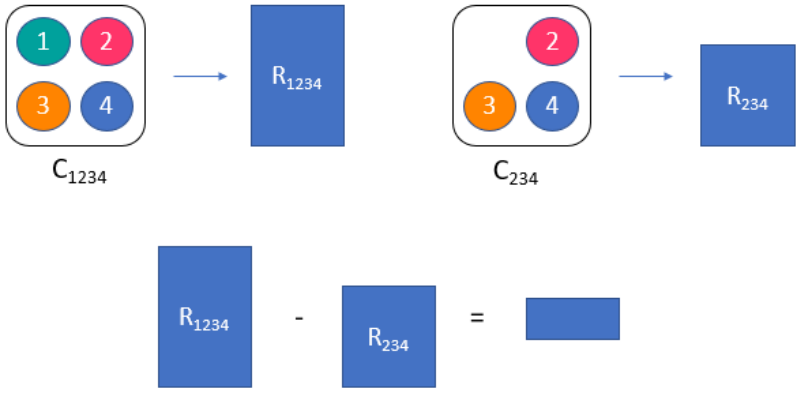
3.3.1 Importància de les variables en el Random Forest

Com he esmentat en l'apartat del Random Forest, aquest model conté l'índex Gini a les seves branques de decisió dels diferents arbres. Aquest índex mesura la puritat de les particions.
A cada arbre es pot calcular la importància de cada variable, però per donar un resultat més robust i general del model es sol calcular la mitjana de cada arbre del RF.
En Python, fent servir la llibreria Scikit-Learn, el RF té de manera nativa l'opció de mostrar la importància de les variables després de l'entrenament del model.

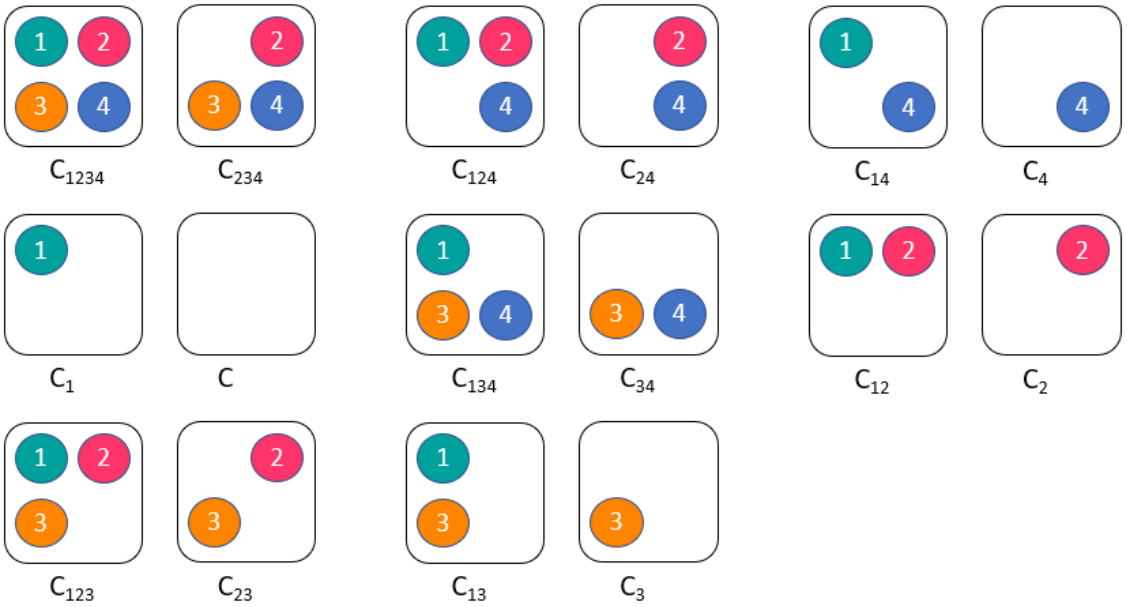
3.3.2 Shapley values

L'ús de Shapley Values per a l'explicabilitat d'un model d'aprenentatge computacional és una solució, basada en el concepte utilitzat en Teoria de Jocs, que tracta de trobar una distribució justa dels guanys depenent de la contribució de cada membre d'una coalició.
Es tracta d'una solució *post hoc*, és a dir, és aplicable per a models complexos fent servir només la sortida després que aquest hagi estat entrenat. També es tracta d'una solució agnòstica ja que no depèn de l'estructura concreta del model. Els Shapley Values ens permeten obtenir una intrepetació local (informació sobre una sola predicció del model) i global (la contribució general de cada variable).

Per entendre més el concepte basat en Teoria de Jocs, podem imaginar l'exemple en què 4 jugadors d'un mateix equip guanyen un torneig i volen saber quina és la repartició justa del premi econòmic per cada membre. Per fer-ho cal buscar la contribució de cadascú. La idea es basa a veure quin és el resultat o puntuació amb i sense un jugador per veure'n la diferència i així obtenir el que ha aportat. A continuació es mostra el concepte per al jugador 1 (considerant que l'equip disposa de 4 membres). R representa el resultat de la combinació C:



Encara que sembli trivial, cal fer aquest procés per cadascuna de les combinacions en què el jugador 1 participa. És a dir, a l'exemple es mostren tots els casos en què el jugador 1 contribueix i, a la dreta (de cadascun dels casos), la mateixa combinació sense aquest jugador (1):



Amb totes aquestes parelles de combinacions es calculen les diferències i la mitjana d'aquestes és el Shapley Value del jugador 1. Ara caldria repetir-ho pels 3 jugadors restants, és a dir, fer totes les combinacions amb i sense els jugadors 2, 3 i 4 (cal destacar que algunes combinacions es repetirien en aquest procés, si es fa jugador per jugador i no s'aprofiten les combinacions anteriors ja calculades).

Si suposem aquest joc com un model, els jugadors de l'equip com les variables de les dades i el resultat o puntuació com la variable objectiu del model, podem aplicar aquest concepte per entendre la contribució de cada característica i interpretar el comportament del model.

Si tenim el model f sobre $x = (x_1, \cdots, x_n)$, calculant els Shapley Values, ϕ_i, per cada variable x_i obtenim:

$$g(x') = \phi_0 + \sum_{i=1}^{n} \phi_i x'_i,$$

amb ϕ_0 constant i x' aproximadament x, $x \approx x'$. Aleshores diem que g és el model explicatori de f i denotem $g(x') \approx f(x)$.

Un cop explicat el concepte en Teoria de Jocs cal veure si aquesta solució és aplicable computacionalment a un cas real d'un model aprenentatge automàtic. Si es calcula la quantitat de combinacions a mostrejar per part del model entrenat es pot veure que a l'augmentar el nombre de variables es torna un problema inviable. A la següent taula

es mostren dues possibles maneres de mostrejar totes les combinacions. La primera ($n \cdot 2^{n-1}$) seria el cas en què iterem per cada variable i provem totes les combinacions on aquesta es fa servir (i es compararia amb les que no es pot fer servir). Seria més eficient pel que fa a memòria però molt menys en l'àmbit computacional. En la segona (2^n) es calculen totes les combinacions possibles considerant que podem guardar-les en memòria. Per tant, tindria major cost en memòria i menor cost computacional.

# Variables	$n \cdot 2^{n-1}$	2^n
4	32	16
10	5120	1024
32	$6.9 \cdot 10^{10}$	$4.3 \cdot 10^9$

En aquesta taula es pot veure clarament que quan s'augmenta el nombre de variables del model es fa impossible d'executar. Per tant, cal trobar una solució que redueixi clarament aquesta complexitat.

És per això que la llibreria de Python SHAP, que implementa els Shapley Values com a tècnica per donar explicabilitat als models, disposa de diferents *explainers*, o explicadors, amb menor complexitat computacional.

En especial, a la llibreria SHAP, s'ha implementat el Kernel Explainer que és aplicable a gairebé qualsevol tipus de model de Scikit-Learn, Torch, Keras i altres. Es tracta d'una eina d'explicació agnòstica en què aproxima els valors Shapley fent servir un mètode basat en Kernels.

Aquesta implementació (KernelExplainer) comença per crear un objecte KernelExplainer amb dos paràmetres d'entrada, el predictor del model que volem explicar i un subconjunt explicatori de les dades. Fent servir aquestes mostres es crea un model "substitut" fent servir una regressió basada en kernels ponderats (diferents pesos per a cada mostra), que aproxima el comportament del model original. Els Shapley Values es calculen fent servir subconjunts fixos i permutacions aleatòries de les característiques avaluant les prediccions del model "substitut". Com s'ha explicat en l'exemple de Teoria de Jocs, es calculen les diferències entre les prediccions quan s'inclouen les característiques fixes i quan no s'inclouen. Donat que calcular els

Shapley Values és molt costós, el KernelExplainer utilitza el mètode de Monte Carlo per a la generació de mostres.
Un cop generats els valors per a cada variable, es poden entendre com les contribucions de cada característica a la predicció del model. Un valor positiu indica que la variable fa augmentar el valor de la predicció (un de negatiu la fa disminuir).
Tot i que aquest és l'explicador més versàtil, n'hi ha d'altres més específics: TreeExplainer per models basats en arbres de decisió, DeepExplainer per a models d'aprenentatge profund (en especial els implementats en TensorFlow i Keras) i el LinearExplainer per a models lineals.

4 Aplicació de la predicció del nivell de perill a Suïssa

L'objectiu principal d'aquest treball consisteix en trobar evidències de les capacitats dels algorismes d'aprenentatge automàtic per a la predicció dels riscs d'allaus. Per assolir aquest objectiu ens calen evidències de la capacitat predictiva i ens calen mecanismes d'explicabilitat que permetin la comprensió dels resultats. En aquest capítol, reproduiré el treball desenvolupat a l'article "Data-driven automated predictions of the avalanche danger level for dry-snow conditions in Switzerland" en l'entorn de Suissa on es mostrava l'alta capacitat predictiva del model de Random Forest per a la predicció del risc d'allaus. Seguidament, he utilitzat altres models de predicció i he fet un anàlisis d'explicabilitat.

En aquest apartat faig una descripció precisa de l'origen de les dades i el seu contingut i una explicació del model provat, les variables utilitzades i els obtinguts.

4.1 Registre històric del nivell de perill

A l'article es fan servir dos conjunts de dades que contenen variables meteorològiques mostrejades cada 24 hores i dades simulades sobre l'estat del mantell nival. Aquestes dades han estat combinades amb el registre

històric de risc d'allaus. Les dades cobreixen des de l'hivern de 1997/1998 fins al de 2019/2020.
A les dades de l'article només es fan servir registres relacionats amb allaus de neu seca i no de neu humida. Això és degut al fet que els factors desencadenants són molt diferents i caldria fer models diferents de predicció. A més, cal destacar que en el cas de neu seca els butlletins es fan diàriament mentre que per neu humida només quan es donen aquestes condicions (a Suïssa).

Les dades meteorològiques provenen de la xarxa IMIS (Intercantonal Measurement and Information System). Aquesta va començar el 1996, i a l'hivern de 1997/1998 tenia 50 estacions operatives, el 2020 ja en tenia 182. 124 d'aquestes estacions es troben refugiades del vent, situades en terreny pla, són estacions de neu. Sobre aquestes últimes, per cadascuna d'elles, fent servir dades meteorològiques, s'executa el model Snowpack que dona informació detallada del mantell nival així com de la microestructura de la neu. Pel que fa a les estacions de vent, aquestes estan generalment situades en zones de més altitud i exposades (cims o crestes entre muntanyes).

Per tant, es diferencia entre estacions de neu i estacions de vent en funció de la posició estratègica en què han estat situades per captar millor unes dades concretes. En el cas de les de vent, es captarà millor la velocitat i intensitat del vent a zones més exposades a aquest factor. Mentre que les de neu necessiten estar més aïllades del vent per mesurar exactament quina és la precipitació local.

Pel que fa a les dades de risc d'allaus, aquestes han estat extretes dels butlletins publicats pel servei nacional d'alerta d'allaus de Suïssa (a l'institut de recerca de neu i allaus). El nivell de perill en el conjunt de dades serà la variable a predir.

4.1.1 Descripció del conjunt de dades

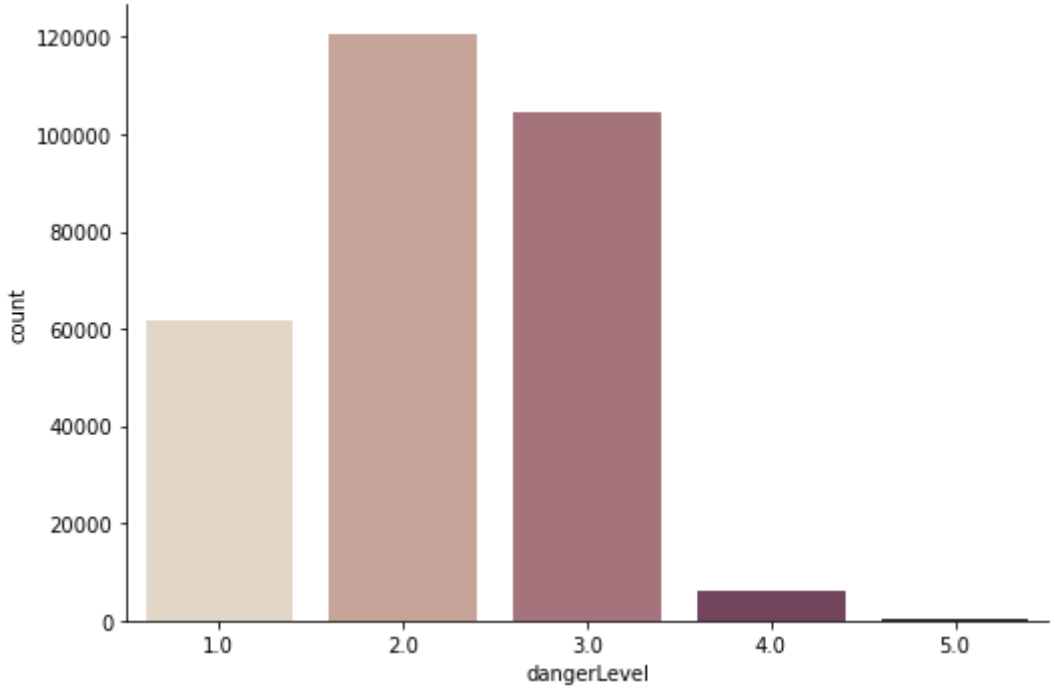
Amb tota aquesta informació els autors de l'article han creat dos fitxers CSV preparats per a fer un model de predicció del risc d'allaus, RF1 i RF2. El primer conté tots els registres de les dades mentre que el segon és un subconjunt filtrat on s'han seleccionat els registres dels quals s'ha considerat que el nivell de perill s'ha proposat de manera més correcta i precisa.
A l'Annex I - Variables del conjunt de dades - es mostren i expliquen totes les columnes del conjunt de dades.

4.1.2 Anàlisi exploratòria de les dades

Per a la meva anàlisi faré servir el dataset RF1, que és el conjunt de dades menys preprocessat.

Recomptes

Conté 77 columnes (corresponents a les característiques de l'apartat anterior 4.1.1) i 292837 files. En total hi ha 3820 dies registrats amb una mediana de 85 registres per cadascun i una mitjana de 76.7 (la quantitat de registres per dia depèn de la quantitat d'estacions en funcionament). Aquesta quantitat de dades a priori sembla més que suficient per entrenar i garantir un model robust. D'aquestes files, un 88% està dedicat al *train* i el 12% restant al *test*. El primer conjunt serà utilitzat per a l'entrenament del model i la validació, per ajustar els hiperparàmetres, mentre que la part de *test* serà per veure els resultats finals.
En aquest treball és essencial entendre la prevalença de cada nivell de perill. Del total de files un 21.1% tenen assignat nivell 1, 41.2% nivell 2, 35.6% nivell 3, 2.0% nivell 4 i 0.1% nivell 5. Per tant, es pot observar clarament que els nivells 2 i 3 són més prevalents i que, en especial, els nivells 4 i 5 són poc habituals.



Estadístics per a diferents variables

El primer estadístic important, i que mostra un resultat similar al del gràfic anterior, és la mitjana del nivell de perill que és 2.19 (amb desviació estàndard 0.79). És a dir, la majoria dels registres del conjunt de dades es troben propers a aquest valor (nivell 2).

Pel que fa a les variables pel model, la mitjana de neu recent en les últimes 72 hores és de 14.95 i pels 5 nivells de perill (de l'1 al 5) són 6.06, 7.67, 28.05, 67.81 i 113.7. Per tant, hi ha un clar increment de la neu recent que pot provocar un increment del nivell de perill. Altres variables com les relacionades amb el vent o la humitat relativa també presenten un augment conjuntament amb el nivell de perill. Aquestes segurament seran variables útils per al model.

Per altra banda, n'hi ha algunes que es mantenen constants independentment del nivell a causa de la manera en què s'han capturat les dades, com l'alçada de les estacions meteorològiques.

Correlacions

En aquest apartat, pot ser interessant veure la correlació entre les diferents columnes amb la columna a predir (*dangerLevel*). Això donarà una visió de quines seran

possiblement les columnes importants per al model. Les variables més correlacionades (valor absolut > 0.5) amb el nivell de perill són Pen_depth, HN72_24 i HN24_7d. És a dir, que la profunditat de penetració en el mantell i la neu recent dels últims dies són els factors que, a priori, semblen més rellevants per a predir el nivell de perill per al desencadenament d'allaus. D'altres amb una correlació significativa són les relacionades amb el vent, l'elevació, la temperatura de l'aire, la radiació solar i la humitat relativa, entre d'altres.

4.2 Modelització i validació

El model utilitzat en primer lloc és el Random Forest (RF). És un model d'aprenentatge computacional basat en arbres de decisió que permet realitzar classificació i regressió. En aquest cas, al voler predir una variable categòrica (5 nivells), es fa servir per classificar.

Les característiques fetes servir en el model són les següents: HN24, HN24_7d, HN72_24, HS_mod, ILWR, ISWR_diff, ISWR, LWR_net, MS_Snow, Qg0, Qs, Qw, RH, S4, Sn, Ss, TA TSS_mod, VW_drift, VW, ccl_pwl_100, min_ccl_pen, pAlbedo, Pen_depth, sn38_pwl_100, wind_trans24, wind_trans24_3d, wind_trans24_7d, zSn i zSs. Aquestes són les relacionades amb la precipitació dels darrers dies, la radiació solar, el vent local dels darrers dies, l'estabilitat del mantell i la temperatura. A l'Annex I - Variables del conjunt de dades - s'expliquen cadascuna de les variables utilitzades.

Aquestes han estat triades buscant mantenir les millors mètriques del model sense comprometre la complexitat computacional. És a dir, en la mesura del possible, s'han triat el mínim de variables mantenint els millors resultats.

Un detall important és que els autors de l'article, al veure que el nivell de perill 5 és tan escàs a les dades, van decidir ajuntar-lo amb el nivell 4. D'aquesta manera i hi hauria només 4 classes a predir (i no 5). Tot i això, jo he preferit no fer-ho i a l'apartat de resultats realment es veurà si afecta o no a les prediccions.

4.2.1 Entrenament i predicció

Durant la fase d'entrenament he ajustat manualment els paràmetres del model. Sí que he obtingut resultats molt similars als de l'article però amb hiperparàmetres diferents: màxima profunditat 200, nombre d'estimadors 1000 i nombre de tasques -1 (paràmetre per aprofitar tots els processadors i accelerar l'entrenament). Per la resta de paràmetres he deixat els predeterminats de sklearn (RF Sklearn).
A l'article però els paràmetres que fan servir són: màxima profunditat 40, mínim de mostres per fulla 6, mínim de mostres per fer un *split* 12 i nombre d'estimadors 1000. De la mateixa manera, també ajusten el nombre de tasques per fer servir tots els nuclis del processador.

4.2.2 Evaluació i resultats

Amb diverses execucions sobre la part de validació (30% de la part d'entrenament) es poden apreciar resultats molt estables al voltant de les següents mesures:

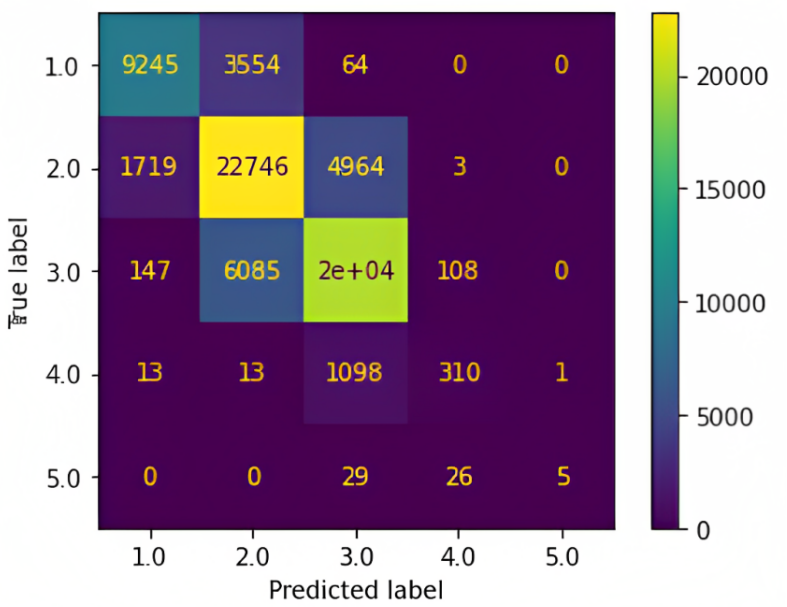
- *Accuracy*: 0.75
- *Precision*: 0.75
- *Recall*: 0.75
- *F1 Score*: 0.75

I si ens centrem en l'*accuracy* de cada nivell:

- Nivell 1: 0.83
- Nivell 2: 0.70
- Nivell 3: 0.77
- Nivell 4: 0.69
- Nivell 5: 0.83

En general els resultats són força estables, i amb diverses execucions la variabilitat és baixa excepte pel cas del nivell 5.

Per acabar de veure quins són els punts febles del model, es mostra la matriu de confusió:



Clarament es pot veure que els nivells més prevalents són els que es prediuen millor. Mentre que els nivells 4 i 5 tenen pitjors mètriques. Tot i això, cal destacar que la confusió en general es troba en els nivells consecutius.

També he provat altres models com l'AdaBoost, el SVM o el XGBoost, però només amb aquest últim he aconseguit resultats similars (lleugerament pitjor *accuracy*, 0.73 en el total del model). A continuació mostro alguns dels models fets sevir en la següent taula sobre el conjunt de validació (30% del conjunt inicial d'entrenament):

Model	***Accuracy***	***F1 Score***
Random Forest	0.75	0.74
XGBoost	0.73	0.73
SVM*	0.56	0.54
AdaBoost	0.45	0.44
Perceptron	0.57	0.50
Decision Tree	0.63	0.63
MLP	0.70	0.70

Aquesta és la taula de resultats amb el conjunt reservat al test:

Model	***Accuracy***	***F1 Score***
Random Forest	0.72	0.72
XGBoost	0.72	0.72
SVM*	0.53	0.51
AdaBoost	0.53	0.47
Perceptron	0.64	0.63
Decision Tree	0.60	0.60
MLP	0.72	0.72

Amb tots els classificadors la matriu de confusió és similar. Els tres primers nivells força ben classificats i 4 i 5 amb resultats més pobres. I en general sempre més confusió en els nivells consecutius.

* En el cas del model SVM, per culpa de la seva complexitat computacional, només he pogut fer l'entrenament amb un 10% del total de les dades.

Accuracy de cada regió

Prenent com a referència que cada zona correspon a les regions que la tenen com per primer nombre del codi (és a dir que la zona 1 compren totes les regions que el seu codi comença per 1: 1111, 1112, 1113,...), les zones i *accuracy* d'aquestes són:

- Zona 1: Part occidental del flanc nord dels Alps: 0.77
- Zona 2: Part central del flanc nord dels Alps: 0.75
- Zona 3: Part oriental del flanc nord dels Alps: 0.78
- Zona 4: Valais (sud-oest de Suïssa): 0.75
- Zona 5: Part nort i central dels Grisons: 0.74
- Zona 6: Part central del flanc sud dels Alps: 0.74
- Zona 7: Part oriental del flanc sud dels Alps: 0.72

Cal mencionar que a les dades no tenim zones ni 8 ni 9 (que sí que es mostren al mapa). També cal especificar que quan es parla d'Alps només fa referència als Alps Suïssos.

Amb aquestes mètriques no es destaquen grans variacions, en general els resultats (en diverses execurcions) són estables al voltant del 0.71-0.79.

4.3 Interpretabilitat i explicabilitat del model

En el context d'un model d'aprenentatge automàtic com el Random Forest per a donar suport a la predicció del nivell de perill d'allaus, és essencial entendre de quina manera es comporta el model i per quin motiu prediu cada nivell. En aquest apartat intento mostrar l'explicabilitat del model i d'algunes prediccions en concret amb dues tècniques diferents.

Per als següents dos mètodes només faré servir el model de Random Forest ja que és el que dona mètriques més elevades (i estables).

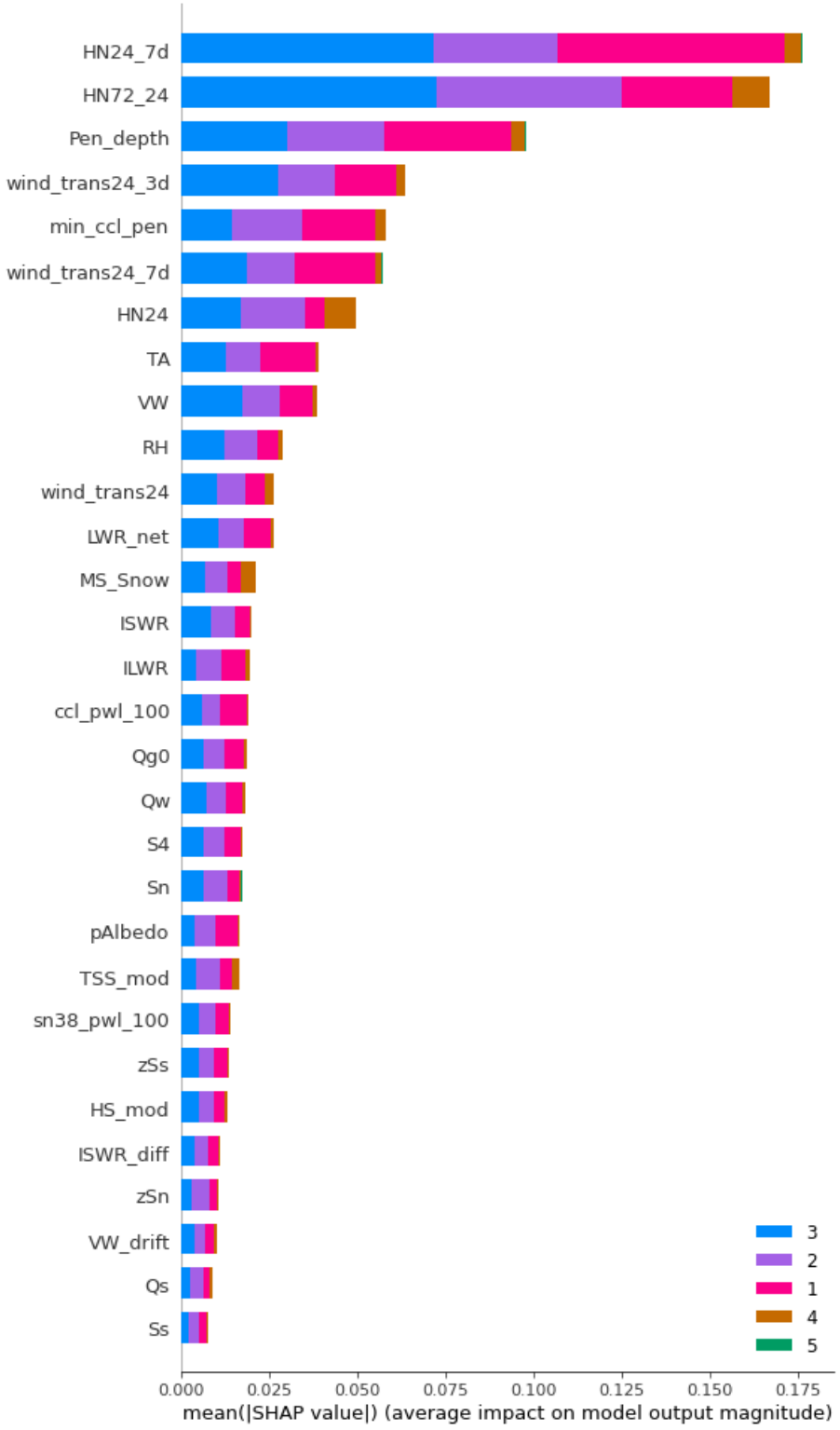
4.3.1 Proximitat i importància al Random Forest

Com està explicat en la secció 3.3.1, amb el Random Forest amb l'índex *Gini* és immediat donar la importància de les variables al model.

Les característiques més rellevants són les variables relacionades amb la neu nova dels darrers dies (HN24_7d i HN72_24), la profunditat de penetració (Pen_depth), la longitud mínima de tall crítica (min_ccl_pen) i les relacionades amb el vent dels darrers dies (wind_trans24_7d i wind_trans24_3d). Els valors, calculats amb el decreixement mitjà d'impuritat, són: 0.082, 0.080, 0.077, 0.051, 0.043 i 0.042 (seguint l'ordre en què s'han esmentat les variables prèviament).
La importància de les variables al Random Forest segueix la mateixa línia de resultats que l'anàlisi de correlacions.

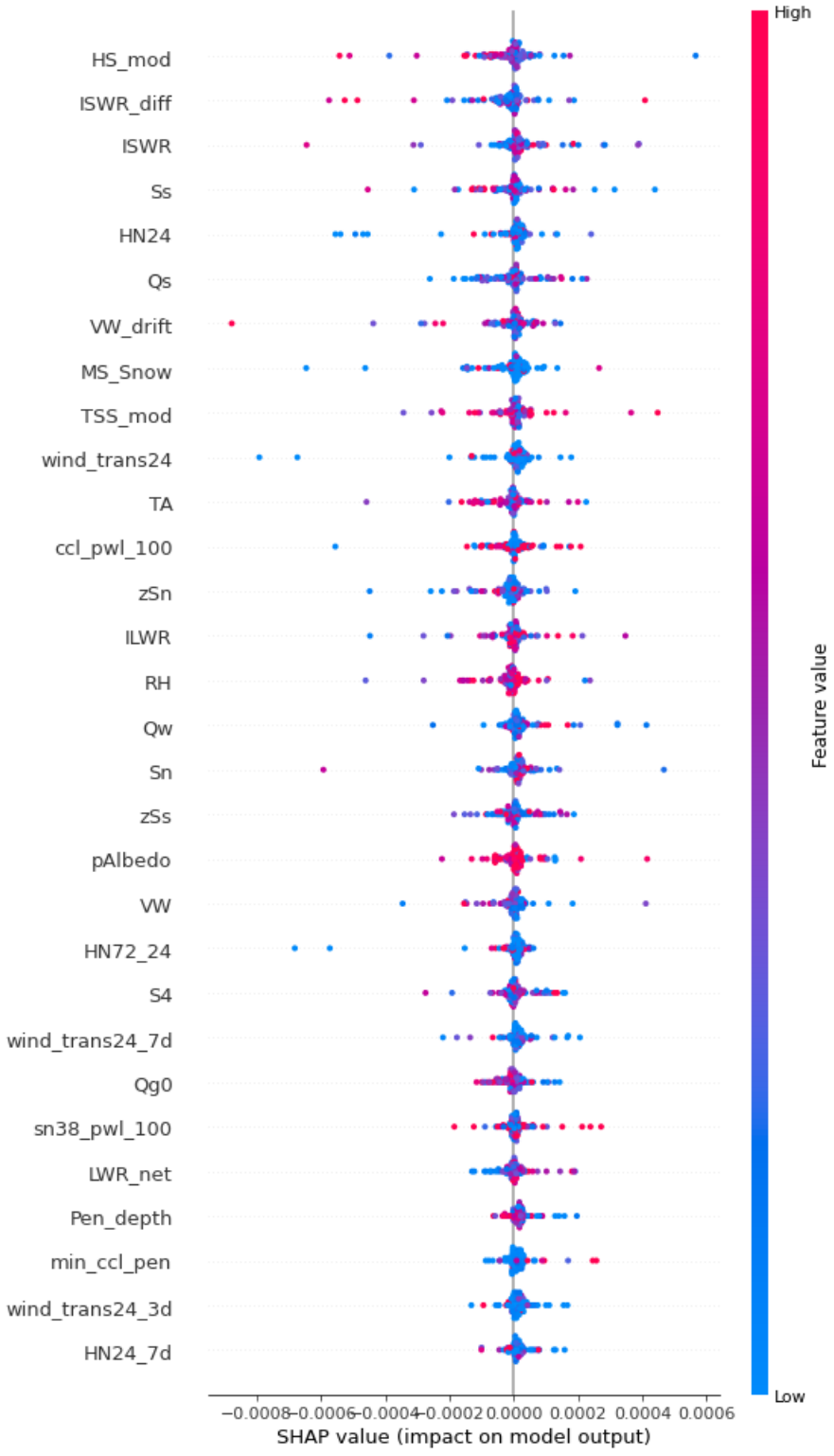
4.3.2 Shapley Values

Seguint l'implementació de la llibreria SHAP, i les tècniques explicades a l'apartat 3.3.2 la contribució de les variables utilitzades al model és la que es veu al següent gràfic, diferenciant per cada nivell de perill (colors de la llegenda):



Donada la complexitat computacional el KernelExplainer l'he hagut d'entrenar amb només 100 mostres del train de les dades, i la importància mitjana s'ha generat amb 100 mostres de la part de validació. En aquesta representació es mostra la mitjana de l'impacte de cada característica en el model, per a cada nivell de perill.

Ara, en el següent gràfic, es mostra la contribució de cada variable en funció del seu valor (si és alt - vermell o si és baix - blau):



En aquest cas, la interpretació de resultats és més difícil de fer visualment. Tot i

això, es veuen clares diferències entre algunes de les característiques. En el cas de les variables relacionades amb la precipitació en forma de neu i les relacionades amb el vent, que eren de les més rellevants pel model i les més correlacionades amb el nivell de perill, semblen no tenir cap distribució marcada (en cap dels sentits, ni positiu ni negatiu).

Per altra banda, sí que n'hi ha algunes que es poden analitzar. La profunditat de penetració de l'esquiador (Pen_depth) tot i que no té una distribució gaire clara, sembla que la majoria dels valors (inclosos els grans) semblen estar més concentrats a la part positiva de l'eix x, és a dir que es tracta d'una variable que en general contribueix positivament al nivell de perill. En el cas de la radiació entrant d'ones curtes (ISWR) sembla que hi ha més punts vermells a la dreta i més blaus a l'esquerra, és a dir, a més radiació més nivell de perill d'allaus. En general, la radiació solar, a curt termini, afecta l'estabilitat de les capes febles del mantell i pot donar lloc a allaus en zones amb capes febles persistents, de neu humida o fins i tot la possibilitat de provocar lliscaments de tot el mantell.

4.4 Procediment computacional

En aquest apartat explico de manera resumida el codi fet servir en els punts anteriors. Tot el codi està escrit en Python. Els scripts es poden trobar al repositori de GitHub: GitHub - Snow Avalanche Danger Level Forecast.

He separat el treball en cinc scripts diferents, quatre dels quals contenen les funcions que es criden al *main*. Aquests són el main.py, utils.py, upload_data.py, data_analysis.py i el train.py.

El primer pas és el de carregar les dades. Per fer-ho cal cridar la classe *get_data* en la que es pot triar el dataframe RF1 o RF2 amb les funcions *get_rf1* o *get_rf2* i si es vol la part d'entrenament o de test. També se seleccionen les característiques en un llistat.

```python
class get_data:

    def __init__(self, project_dir: str, set_t: str, features: list = []):
        self.project_dir = project_dir
        self.set = set_t
        self.rf1 = pd.DataFrame()
        self.rf2 = pd.DataFrame()
        self.cp_data = pd.DataFrame()
        self.features = features
```

Després es fa l'anàlisi de les dades que es mostra a la secció 4.1.2 i la selecció de característiques pel model (entrada com a llista).

Per a la separació de les dades d'entrenament (en *train* i *validation*) faig servir la funció de Sklearn:

```python
def train_valid_split(data: pd.DataFrame, target: str = 'dangerLevel',
                      train_size=0.7):
    X = data.drop(columns=target)
    y = data[[target]]
    if train_size == 0.0:
        return np.nan, X, np.nan, y
    return train_test_split(X, y, train_size=train_size)
```

Un cop preparats els conjunts (entrenament, validació i test) es fa la selecció i entrenament del model:

```python
def train(X, y, mod):
    if mod == 'xg':
        model = XGBClassifier(objective="multi:softprob")
    elif mod == 'ada':
        model = AdaBoostClassifier(n_estimators=100)
    elif mod == 'svm':
        model = SVC(kernel = 'linear', gamma = 'scale', shrinking = False,)
    elif mod == 'perceptron':
        model = Perceptron(max_iter=1000)
    elif mod == 'dt':
        model = DecisionTreeClassifier()
    elif mod == 'mlp':
        model = MLPClassifier(solver='lbfgs', alpha=1e-5,
                              hidden_layer_sizes=(100, 100))
    else:
        model = RandomForestClassifier(max_depth=200,
                                       n_estimators=1000,
                                       n_jobs=-1)
    model.fit(X, y)
    return model
```

Després es fa la predicció del model, es mostren les mètriques i la matriu de confusió. Finalment, es fa l'anàlisi de la importància de les variables del model. Es fa de

manera nativa amb el model (en cas que la implementació tingui l'opció, com és el cas del Random Forest) i amb la llibreria de Shap. Aquesta segona opció crea un *explainer* amb un subconjunt aleatori de l'entrenament i el model, i amb el test o validació es calculen els Shapley Values.

```
explainer = shap.KernelExplainer(model.predict_proba,
                                 X_train.sample(n=100))
# Generate SHAP values
X_val_red = X_val.sample(n=100)
shap_values = explainer.shap_values(X_val_red)

# Plot the SHAP summary plot
shap.summary_plot(shap_values, X_val, class_names=model.classes_,
                  max_display=X_val.shape[1])

# Calculate the mean SHAP values
mean_shap_values = np.mean(shap_values, axis=0)

shap.summary_plot(mean_shap_values, X_val_red, class_names=model.classes_,
                  max_display=X_val_red.shape[1])
```

5 Treball futur

Un dels darrers objectius del meu treball és el de veure la possibilitat de realitzar aquest mateix estudi i crear el model de predicció fent servir dades extretes de Copernicus. El concepte es basa en la mateixa idea inicial, ser capaç de predir el grau de perill que s'informa en els butlletins d'allaus fent servir dades meteorològiques locals. Aquesta implementació, en cas de donar bons resultats (*accuracy* > 70-90% i una explicabilitat corresponent a la realitat), seria aplicable a qualsevol regió en què hi ha el mateix detall de dades meteorològiques i butlletins registrats. Inclús es podria entrenar el model en una zona amb major detall de dades enregistrades per després ser aplicat a altres localitzacions (considerant que el model és prou robust i s'ha entrenat en condicions similars).

Tot i que durant el meu projecte no he tingut prou temps per a implementar aquesta part, sí que he pogut investigar quins serien alguns dels passos a seguir i d'on podria extreure aquestes dades.

En primer lloc, cal disposar de la periodicitat dels butlletins d'allaus i la manera d'extreure la informació de manera automàtica. Es podria fer servir el mateix que s'utilitza durant el meu treball (basant-me en el conjunt de dades de l'article de referència) que és de Suïssa o, per exemple, els de Catalunya (proporcionats per l'ICGC - Institut Cartogràfic i Geològic de Catalunya).

En segon lloc, un cop es disposa d'un dataframe amb data, zona (o regió) i nivell de perill (extrets del butlletí), cal aconseguir creuar-ho amb les dades meteorològiques específiques de la zona. El millor accés a aquest tipus d'informació, en l'àmbit Europeu, és el programa Copernicus. Cercant tots els conjunts de dades disponibles, el que sembla que dona més detall en referència a l'objectiu del treball és el de l'ERA5. Es tracta de la cinquena versió de l'ECMWF *climate reanalysis*, un projecte que combina dades modelades amb observacions de tot el planeta per obtenir un conjunt de dades consistent fent servir les lleis físiques. Aquest mètode està basat en l'assimilació de les dades, es combinen els pronòstics numèrics anteriors amb les noves observacions disponibles de la manera òptima per a donar una estimació de l'estat de l'atmosfera.
En aquest conjunt de dades es fa servir la projecció cartogràfica regular, amb referències de longitud i latitud. La resolució horitzontal és de 0.25° x 0.25° (atmosfera) i 0.5° x 0.5° (ones de l'oceà) en la reanàlisi. La cobertura temporal és del 1940 al present i la resolució és de cada hora. Les variables d'aquest conjunt de dades es troben explicades a la pàgina principal de l'ERA5 de Copernicus. Dins de l'enorme llistat de variables s'hi troben la majoria de les necessàries per al model, algunes de les disponibles són relacionades amb la temperatura, la precipitació (aigua i neu), el vent, la radiació solar, l'evaporació, l'albedo, la rugositat de la superfície, la quantitat de neu acumulada, etc. Evidentment, caldria fer un estudi molt específic de cadascuna, però queda clar que les dades importants per al model hi són.

Per últim, caldria creuar les dades dels butlletins amb les de les variables fent servir les coordenades de les regions on s'ha realitzat el butlletí. Amb aquesta tasca

s'aconseguiria un conjunt de dades molt similar al que es proporciona a l'article de referència. I, per tant, es podria fer l'entrenament del model i fer-lo servir per a predir el nivell de perill d'allaus amb unes condicions concretes.

6 Conclusions

Per concloure el meu treball de final de grau puc dir que he assolit els objectius principals en general, exceptuant el mencionat a la secció de treball futur, i que han estat obtinguts utilitzant tècniques, en gran part, adquirides durant el grau de Matemàtica Computacional i Analítica de Dades. També he de destacar que és un àmbit en el qual es pot realitzar molta més recerca i treball futur per a aconseguir resultats molt més bons, tant en precisió dels models com en explicabilitat, i la possibilitat d'aplicar aquesta metodologia a zones locals.

Fent un repàs dels objectius plantejats en un inici, el primer és el de ser capaç de reproduir els resultats de l'article de referència. He estat capaç d'entendre tots els passos de la metodologia que s'ha fet servir, he descarregat i entès el conjunt de dades i, per acabar, he creat un model d'aprenentatge computacional que donava mètriques properes al 75% d'encert (similar al que es presenta a l'article).

El segon punt, per donar més explicabilitat al model fet servir, també l'he aconseguit, fent servir la implementació nativa dels models basats en arbres de decisió i amb la implementació de Shapley Values a la llibreria SHAP. Aquesta darrera tècnica no és explicada durant el meu grau tot i que sí que he cursat l'assignatura de Teoria de Jocs com a optativa de quart curs. Això implica entendre el concepte teòric i el procés de la metodologia computacional.

El darrer punt dels objectius és el d'obtenir una implementació mixta entre la mencionada al primer punt i fent servir dades externes. Tot i que aquest punt no l'he aconseguit implementar sí que he pogut fer una anàlisi de quina seria la metodologia

a seguir, començant per descarregar les dades seleccionades de Copernicus i seguint per l'extracció de característiques i finalment executant el model més adient. És important entendre que després d'obtenir el conjunt de dades s'hauria de combinar amb el nivell de perill.

Finalment puc donar per acabat el meu treball havent completat tota la metodologia per a veure com les tècniques d'aprenentatge automàtic poden donar suport en la gestió del risc d'allaus de neu, concretament en la predicció del nivell de perill.

Annex

Annex I - Variables del conjunt de dades

Prenent com a referència l'arxiu 'readme' de les dades, les variables o característiques de les quals disposo són les següents:

Columnes descriptives:

- 'datum': data de les dades
- 'station_code': codi de l'estació meteorològica de la xarxa IMIS
- 'sector_id': número de codi de la regió de la localització de l'estació meteorològica
- 'warnreg': número de codi del mapatge de les àrees de risc
- 'forecast_initial_date': data inicial i hora de la finestra de la predicció
- 'forecast_end_date': data final i hora de la finestra de la predicció
- 'elevation_th': elevació crítica (msnm) predita en el butlletí per a cada regió
- 'set': set d'entrenament o de test

Característiques (contínues) pel model:

- 'elevation_station': elevació de l'estació meteorològica (msnm)
- 'Qs': calor sensible [W/m^2]
- 'Ql': calor latent [W/m^2]
- 'TSG': temperatura del sòl [°C]
- 'Qg0': calor del sòl a la interfície [W/m^2]
- 'Qr': energia de la pluja [W/m^2]
- 'OLWR': radiació infraroja sortint [W/m^2]

- 'ILWR': radiació infraroja entrant [W/m^2]
- 'LWR_net': radiació infraroja neta [W/m^2]
- 'OSWR': radiació de reflexió d'ones curtes [W/m^2]
- 'ISWR': radiació d'ones curtes entrant [W/m^2]
- 'Qw': radiació d'ones curtes neta [W/m^2]
- 'pAlbedo': albedo parametritzat [-]
- 'ISWR_h': radiació d'ones curtes entrant en horitzontal [W/m^2]
- 'ISWR_diff': radiació d'ones curtes entrant directa [W/m^2]
- 'ISWR_dir': radiació d'ones curtes entrant difusa [W/m^2]
- 'TA': temperatura de l'aire [°C]
- 'TSS_mod': temperatura de la superfície (modelada) [°C]
- 'TSS_meas': temperatura de la superfície (mesurada) [°C]
- 'T_bottom': temperatura de la base [°C]
- 'RH': humitat relativa [-]
- 'VW': velocitat del vent [m/s]
- 'VW_drift': deriva de la velocitat del vent [m/s]
- 'DW': direcció del vent [°]
- 'MS_Snow': taxa de precipitació sòlida [kg/m^2/h]
- 'HS_mod': altura de neu (modelada) [cm]
- 'HS_meas': altura de neu (mesurada) [cm]
- 'hoar_size': mida dels cristalls de gel [cm]
- 'wind_trans24': deriva del vent de 24h [cm]

- 'wind_trans24_7d': deriva del vent de 7 dies [cm]
- 'wind_trans24_3d': deriva del vent de 3 dies [cm]
- 'HN24': alçada de la neu nova, últimes 24h [cm]
- 'HN72_24': alçada de la neu nova, últims 3d [cm]
- 'HN24_7d': alçada de la neu nova, últims 7d [cm]
- 'SWE': aigua acumulada en el mantell nival [kg/m^2]
- 'MS_water': quantitat total d'aigua [kg/m^2]
- 'MS_Wind': pèrdua de massa per erosió del vent [kg/m^2]
- 'MS_Rain': rati de pluja [kg/s^2/h]
- 'MS_SN_Runoff': lisímetre virtual [kg/s^2/h]
- 'MS_Sublimation': massa de sublimació [kg/m^2]
- 'MS_Evap': massa evaporada [kg/m^2]
- 'TS0': temperatura de la neu a 0.25 m [°C]
- 'TS1': temperatura de la neu a 0.5 m [°C]
- 'TS2': temperatura de la neu a 1 m [°C]
- 'Sclass2': classe d'estabilitat [-]
- 'zSd_mean': profunditat de l'índex de la taxa de deformació [cm]
- 'Sd': índex de la taxa de deformació [-]
- 'zSn': profunditat de l'índex d'estabilitat natural [cm]
- 'Sn': índex d'estabilitat natural [-]
- 'zSs': profunditat de l'índex d'estabilitat Sk38 [cm]
- 'Ss': índex d'estabilitat Sk38 [-]

- 'zS4': profunditat de l'índex d'estabilitat estructural [cm]
- 'S4': índex d'estabilitat estructural [-]
- 'zS5': profunditat de l'índex d'estabilitat 5 [cm]
- 'S5': índex d'estabilitat 5 [-]
- 'pwl_100': capa(s) feble(s) persistent(s) als 100 cm de profunditat des de la superfície [-]
- 'pwl_100_15': capa(s) feble(s) persistent(s) a profunditats entre 15 cm i 100 cm [-]
- 'base_pwl': capa feble persistent a la base [-]
- 'ssi_pwl': índex d'estabilitat estructural a la capa feble [-]
- 'sk38_pwl': índex d'estabilitat Sk38 a la capa feble [-]
- 'sn38_pwl': índex d'estabilitat natural a la capa feble [-]
- 'ccl_pwl': longitud de tall crítica a la capa feble [m]
- 'ssi_pwl_100': índex d'estabilitat estructural a la capa feble superficial [-]
- 'sk38_pwl_100': índex d'estabilitat Sk38 a la capa feble superficial [-]
- 'sn38_pwl_100': índex d'estabilitat natural a la capa feble superficial [-]
- 'ccl_pwl_100': longitud de tall crítica a la capa feble superficial [m]
- 'Pen_depth': profunditat de penetració d'un esquiador [cm]
- 'min_ccl_pen': longitud mínima de tall crítica a una capa més profunda de la profunditat de penetració [m]

Objectiu del model:

- 'dangerLevel': nivell de perill assignat seguint els nivells del European Avalanche Danger Scale (1-Baix, 2-Moderat, 3-Considerable, 4- Alt i 5-Molt Alt)

Bibliografia

Per a l'elaboració del meu treball he fet servir els següents documents i pàgines:

Article

Pérez-Guillén, C., Techel, F., Hendrick, M., Volpi, M., van Herwijnen, A., Olevski, T., Obozinski, G., Pérez-Cruz, F., and Schweizer, J.: Data-driven automated predictions of the avalanche danger level for dry-snow conditions in Switzerland, Nat. Hazards Earth Syst. Sci., 22, 2031–2056, https://doi.org/10.5194/nhess-22-2031-2022, 2022.

Codi i dades: repositori

Aprenentatge computacional i conceptes matemàtics

AdaBoost: https://medium.com adaboost-classifier

SVM: https://www.ibm.com/docs/es/spss-modeler/saas?topic=models-how-svm-works

Random Forest: https://www.ibm.com/topics/random-forest

Scikit-Learn: https://scikit-learn.org/stable/

Shapley Values: https://towardsdatascience.com Shapley Values i https://shap.readthedocs.io/en/latest/index.html

Mètode de Monte Carlo: https://www.ibm.com/topics/monte-carlo-simulation

Nivologia i servei d'alerta d'allaus

EAWS (European Avalanche Warning Services): https://www.avalanches.org/

Risc d'allaus: https://www.slf.ch/en/index.html

Snowpack: https://www.slf.ch/en/snow/snowpack/snow-cover-modelling.html

Xarxa IMIS: https://www.sensalpin.ch/en/measuring-networks/imis/

ICGC: https://www.icgc.cat/ca/

AEMET: https://www.aemet.es/ca/

Copernicus

Web del programa Copernicus: https://www.copernicus.eu/en

ECMWF reanàlisi: https://www.ecmwf.int/en/research/climate-reanalysis

Dades: cds climate copernicus.eu ERA5