RAPPORT SUR L'UTILISATION DE MODELES DE RNN SUR LA PREDICTION DE LA HAUTEUR DE NEIGE DANS LES MONTAGNES FRANCAISES.

Lien projet Github (scripts, données, graphiques) :

Quand je randonnais, skiais ou faisais du vélo dans les Alpes françaises, je me suis souvent demandé comment marchait le processus de chute de neige sur le sol et sa fonte. C'est assez compliqué et cela dépend de beaucoup de paramètres. L'ajout de neige dépend majoritairement des chutes de neige, évidemment. Pour la fonte, c'est plus complexe. C'est dû à :

- La température de l'air
- Le rayonnement solaire et sa durée dans la journée
- Le versant de la montagne. Sur un adret (versant sud) la neige fond plus vite que sur un ubac (versant nord). On peut particulièrement voir la différence entre les deux versants en avril, mai et juin.
- La pluie, surtout si l'eau a une certaine température
- le vent peut également arracher les couches de neige supérieures
- La masse de la neige : à cause de ses propriétés calorifiques, une plus grosse masse de neige mettra plus de temps à fondre
- D'autres facteurs qui sont plus difficiles à cerner et à mesurer. Vont intervenir les ombres d'arbres ou de montagnes, les temps d'exposition au soleil, les microclimats, les passages d'animaux ou d'hommes.

Exemple extrême de ce qu'on peut voir dans les montagnes de l'Hémisphère nord d'avril à juin : sur le versant sud (versant de gauche), la neige a complètement fondu quand sur le versant nord (versant de droite) il en reste une quantité considérable

Au lieu d'essayer d'utiliser de nombreuses formules de thermodynamique (que j'ai oublié depuis des années, ou que je ne connais pas, haha), pourquoi ne pas utiliser des méthodes de Deep Learning ? Si je trouve assez de données temporelles en opendata avec les principaux paramètres évoqués et la hauteur de neige dans les montagnes françaises, les méthodes de RNN (Recurrent Neural Networks) pourront s'avérer intéressantes.

Par chance, pour la France, Météo-France a mis à disposition, sur son portail opendata, des données intéressantes. Le lien est à la fin du rapport.

Les données sont au format .csv, avec un fichier pour chaque mois. On peut déjà remarquer que :

- Les mesures remontent à janvier 1996
- Il y a une mise à jour quotidienne

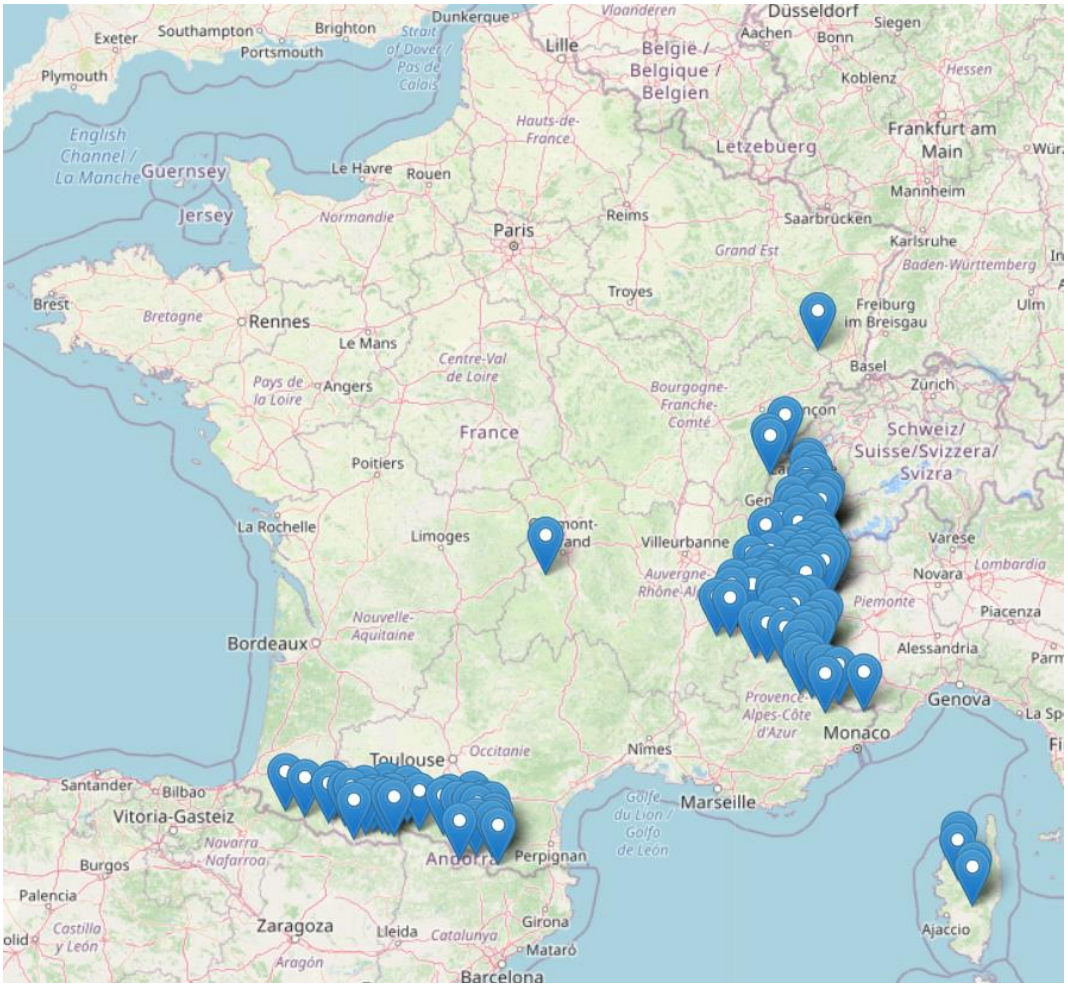
- Les points de mesure sont majoritairement localisés dans les stations de sports d'hiver ou dans des villages. Ou dans leur proche voisinage. Quelques points de mesure sont situés plus haut en montagne, près des remontées mécaniques.
- Il y a majoritairement un point de mesure par station. Parfois deux ou trois. Chamonix en a par exemple quatre.
- Les paramètres disponibles sont nombreux et expliqués ici : https://donneespubliques.meteofrance.fr/client/document/doc_parametres_nivo_197.pdf
  Bonne nouvelle : parmi ces mesures, celles dont j'ai parlé en introduction semblent bien apparaitre : hauteur de neige, hauteur de neige fraiche, vent, précipitations, températures, nuages... pour les données physiques. Et la latitude, longitude, altitude… pour les données géographiques.

On a un total de plus de **647 000** mesures dans les montagnes françaises de janvier 1996 à mai 2021 inclus.

I) STATIONS DE MESURE : LOCALISATION, ALTITUDES, COMPLETUDES

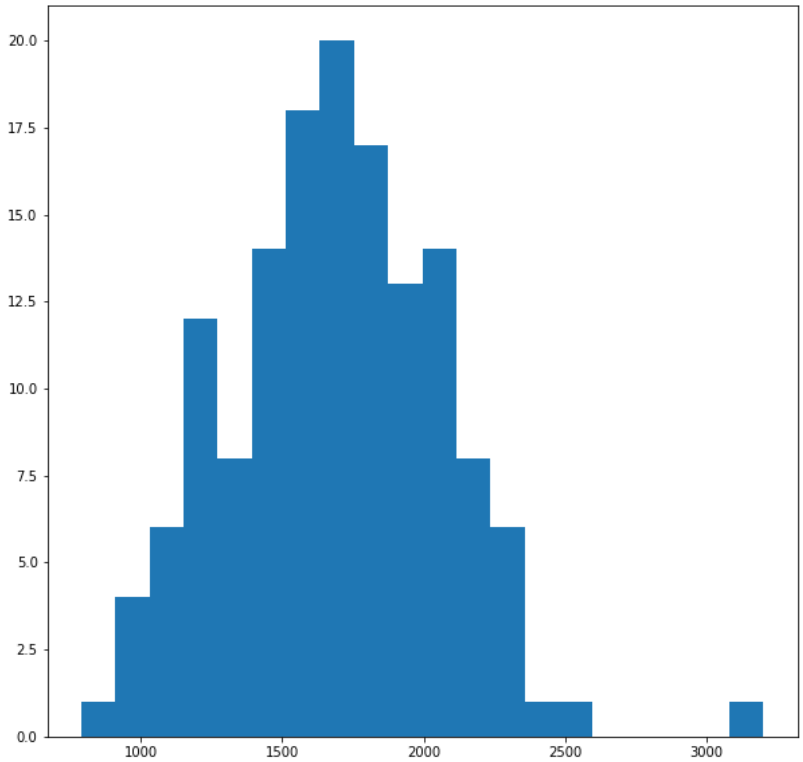
A) GEOGRAPHIE ET ALTITUDE

Ci-dessous, la projection sur une carte OpenStreetMap des stations de mesure. Elles sont bien situées dans les zones montagneuses françaises.



Ci-dessus, la projection sur une carte OpenStreetMap des stations de mesure. Elles sont bien situées dans les zones montagneuses françaises.

Si on regarde la distribution de l'altitude des points de mesure. L'altitude moyenne est de 1693 mètres :



A) NOMBRE DE MESURES

Ci-dessous les 40 stations avec le plus de mesures. Attention, cela ne veut pas dire que les mesures sont ben remplies ! parmi elles, et en fonction des stations, de nombreuses mesures seront incomplètes

	Nom
Iraty	11008
CEILLAC_NIVO	10392
VACHERESSE AUXI	10079
St Hilaire	10046
St Christophe	9985
Maljasset	9264
St Etienne en Devoluy	9216
PELVOUX ST ANTOINE	9093
Les Portes en Valgaudemar	8969
Le Grand Bornand	8959
Courchevel	7911
MEGEVE AUXI	7423
St Paul d Oueil	6718
SENTEIN EYLIE HAUT	6510
La Plagne	6433
ISOLA	6377
Bessans	6371

Mont d Olmes	6016
Val Casterino	5966
VILLARD-BOIS-BARBU	5948
Estenc	5815
Tignes	5725
Val d Isere Joseray	5676
Les Menuires	4929
Les 2 Alpes (Toura NE)	4810
La Rosiere	4765
Flaine	4688
CAUTERETS_LYS	4611
LES CARROZ VILLAGE	4595
Lognan	4557
Auron	4528
Les Arcs	4516
Piau	4508
L Alpe d Huez (SATA)	4492
Avoriaz	4444
Meribel Mottaret	4440
L Hospitalet	4389
CHAMONIX-OBS	4379
LES CARROZ KEDEUSAZ	4307
Pralognan	4161

II) PARAMETRES

D'après la table des données disponibles sur le site de Meteo-France, nous avons ci-dessous les noms, les noms dans les données, le type et l'unité. J'ai calculé la complétude et mis un commentaire à partir de mes observations.

Description (document)	Nom dans les données	type	Unité	Complétude 1996-2021	Commentaire
Indicatif OMM station	Numer_sta	Char		100%	Code de la station, pour faire la jointure avec les données géographiques des stations
Date (UTC)	date	Char	Date-heure	100%	
Altitude de la station	Haut_sta	réel	m	100% (jointure)	Utile pour l'étude
Direction du vent moyen 10 mn	dd	Int	degrés	97%	
Vitesse du vent moyen 10 mn	ff	Float	m/s	97%	Utile pour l'étude
Température	t	Float	Kelvin	96%	Utile pour l'étude
Point de rosée	td	Float	Kelvin	74%	Un temps envisagé pour l'étude, finalement abandonné
Humidité	u	nt	%	74%	Un temps envisagé pour l'étude, finalement

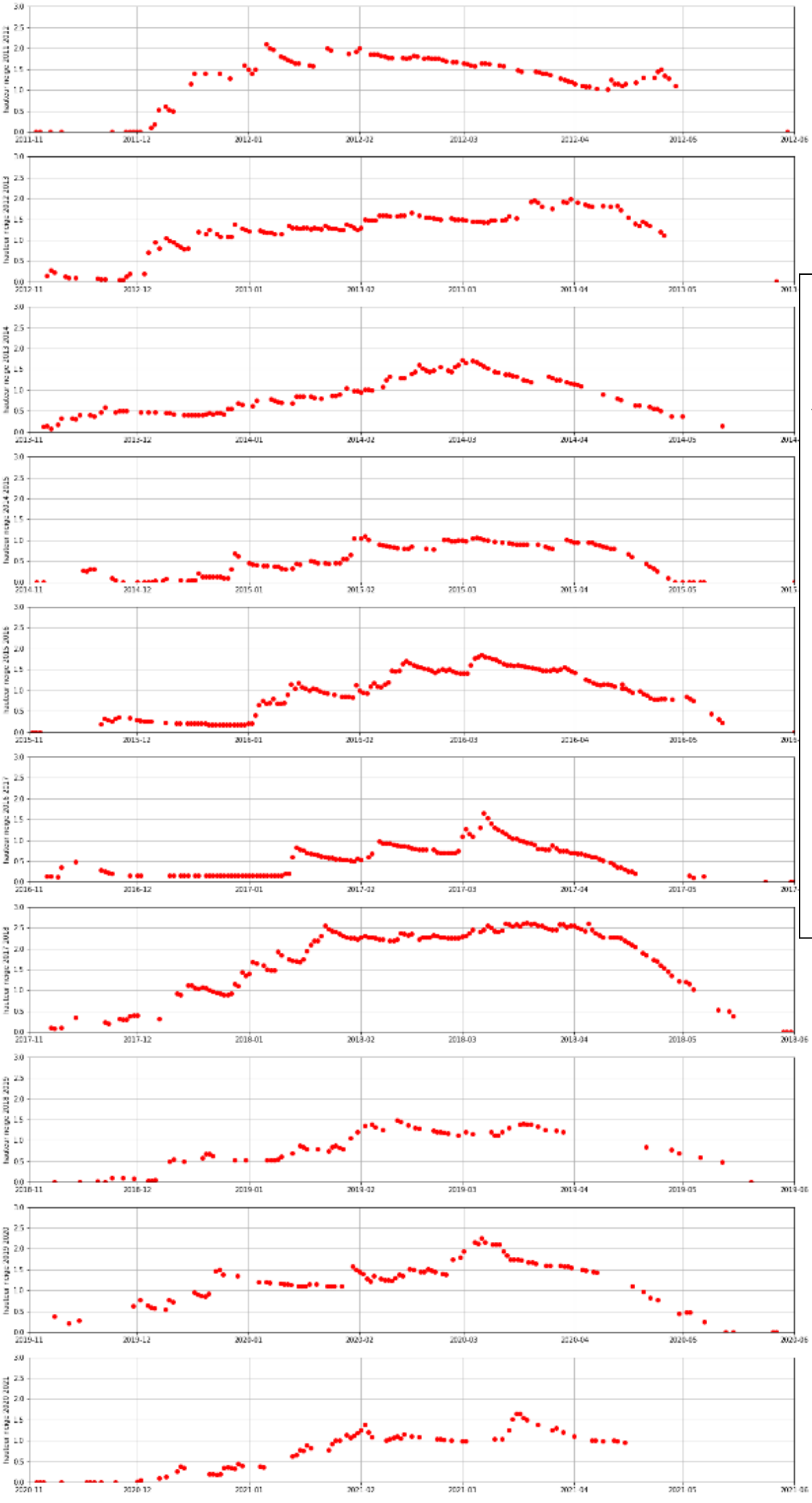
					abandonné
Temps présent	ww	int	code	98%	
Temps passé 1	w1	int	Code	98%	
Temps passé 2	w2	int	code	98%	
Nebulosité totale	n	réel	%	98%	Utile pour l'étude
Nébulosité des nuages de l'étage inférieur	nbas	int	octa	96%	Utile pour l'étude. J'ai vérifié sur quelques webcams et cela semble correspondre à la couverture nuageuse et ainsi en déduction la proportion des rayons du soleil atteignant le sol (0 : ciel bleu, 10 : forte couverture nuageuse)
Hauteur de la base des nuages de l'étage inférieur	hbas	int	octa	63%	
Type des nuages de l'étage inférieur	cl	int	Code	100%	
Type des nuages de l'étage moyen	cm	int	code	100%	
Type des nuages de l'étage supérieur	ch	int	Code	100%	
Précipitations dans les N dernières heures	rrN	réel	mm	61%	Utile pour l'étude
Température minimale sur N heures	tnN	réel	Kelvin	53% pour N=24, 7% pour N=12	Utile pour l'étude. N = 24 heures dans les données
Température maximale sur N heures	txN	réel	K	53% pour N=24, 7% pour N=12	Utile pour l'étude. N = 24 heures dans les données
Hauteur totale de neige	Ht_neige	réel	m	97%	Utile pour l'étude
Hauteur de la neige fraiche	ssfrai	réel	m	64%	Utile pour l'étude
Période de mesure de la neige fraiche	perssfrai	réel	m	64%	Paramètre utile pour filtrer sur certaines données (voir plus tard)
Phénomène spécial	phenspeN	réel	Code	99% pour N=1, 45% pour N=2	
Etendue couche nuageuse 1	Nnuage1	int	code	0%	
Température de surface de la neige	T_neige	réel	Kelvin	63%	
Etat de la neige	Etat_neige	int	code	35%	
Profondeur d'enfoncement de la sonde	Prof_sonde	int	m	64%	
Nuages dans la vallée	Nuage_val	int	Code	63%	
Chasse-neige en altitude	Chasse_neige	int	code	65%	
Description de l'avalanche observée	Aval_descr	int	code	31%	
Genre d'avalanche	Aval_genre	int	Code	31%	
Altitude de départ de l'avalanche	Aval_depart	int	Code	31%	
Exposition de l'avalanche	Aval_expo	int	code	31%	
Estimation du risque d'avalanche	Aval_risque	int	code	65%	
Direction du vent en altitude	Dd_alti	int	degré	25%	
Force du vent en altitude	Ff_alti	réel	m/s	25%	
Hauteur de neige en altitude	Ht_neige_alti	réel	m	15%	
Hauteur de neige fraiche	Neige_fraiche	réel	m	17%	Paramètre inconnu, malgré son nom
Teneur en eau liquide volumique de la neige	Teneur_eau	int	%	0%	
Type de grains de surface prédominants	Grain_predom	int	Code	47%	

Type des grains de surface les plus nombreux	Grain_nombre	int	code	47%	
Diamètre moyen des grains	Grain_diametr	int	Mm	13%	
Indicateur d'homogénéité de la couche	homogeneite	Int	Code	32%	
Masse volumique de la neige	M_vol_neige	réel	Kg/m3	16%	

B) PARAMETRE « HAUTEUR DE NEIGE »

Probablement le plus important car c'est ce que je vais chercher à prédire. Ci-dessous un exemple de station pour laquelle ce paamètre est à peu près complet (la Plagne(73)), avec chaque saison les unes par-dessus les autres.

J'ai déposé, pour chaque station, la hauteur de neige disponible (après filtre des valeurs aberrantes et en ne gardant que les valeurs <10h du matin) pour chaque intervalle [1er novembre – 1er juin] de 1996 à 2021 dans figures/hauteur_neige.



Exemple de mesure de hauteur de neige du 1er novembre au 1er juin de chaque année de l'hiver 2011-2012 (en haut) à 2020-2021 (en bas), de 0 à 3m de neige, pour une station dont les données sont parmi les plus complètes : la Plagne(73).

La saison 2017-2018, connue pour la couverture neigeuse exceptionnelle, est bien visible (4ème graphique en partant de la fin)

A) PARAMETRES GARDES

J'ai décidé de garder, parmi la large palette de paramètres proposés, ceux qui semblent avoir, par bon sens, un impact sur la hauteur de neige.

- Hauteur de neige totale : ce sera la valeur à prédire
- Hauteur de neige fraiche : va influer directement sur la hauteur de neige totale. C'est principalement elle qui va faire grimper la hauteur de neige totale, par une simple addition. Valeur très importante
- Température minimale sur les dernières 24h
- Température maximale sur les dernières 24h
- Altitude : n'est peut-être pas très utile car les températures devraient suffire.
- Vitesse Moyenne du vent sur les dernières 10 min : comme j'avais dit plus haut, il a peut-être une faible influence sur la diminution de la hauteur de neige
- Précipitations : ce sont probablement les pluies, et non la neige (il ne correspond pas au paramètre d'hauteur de neige fraiche,
- Nébulosité : la quantité de nuages impacte l'énergie reçue par le sol

A) PARAMETRES AJOUTES

Quelques valeurs qui n'étaient pas présentes, mais que j'ai jugé bon d'ajouter, étant donné leur importance et leur absence dans les données de Meteo-France :

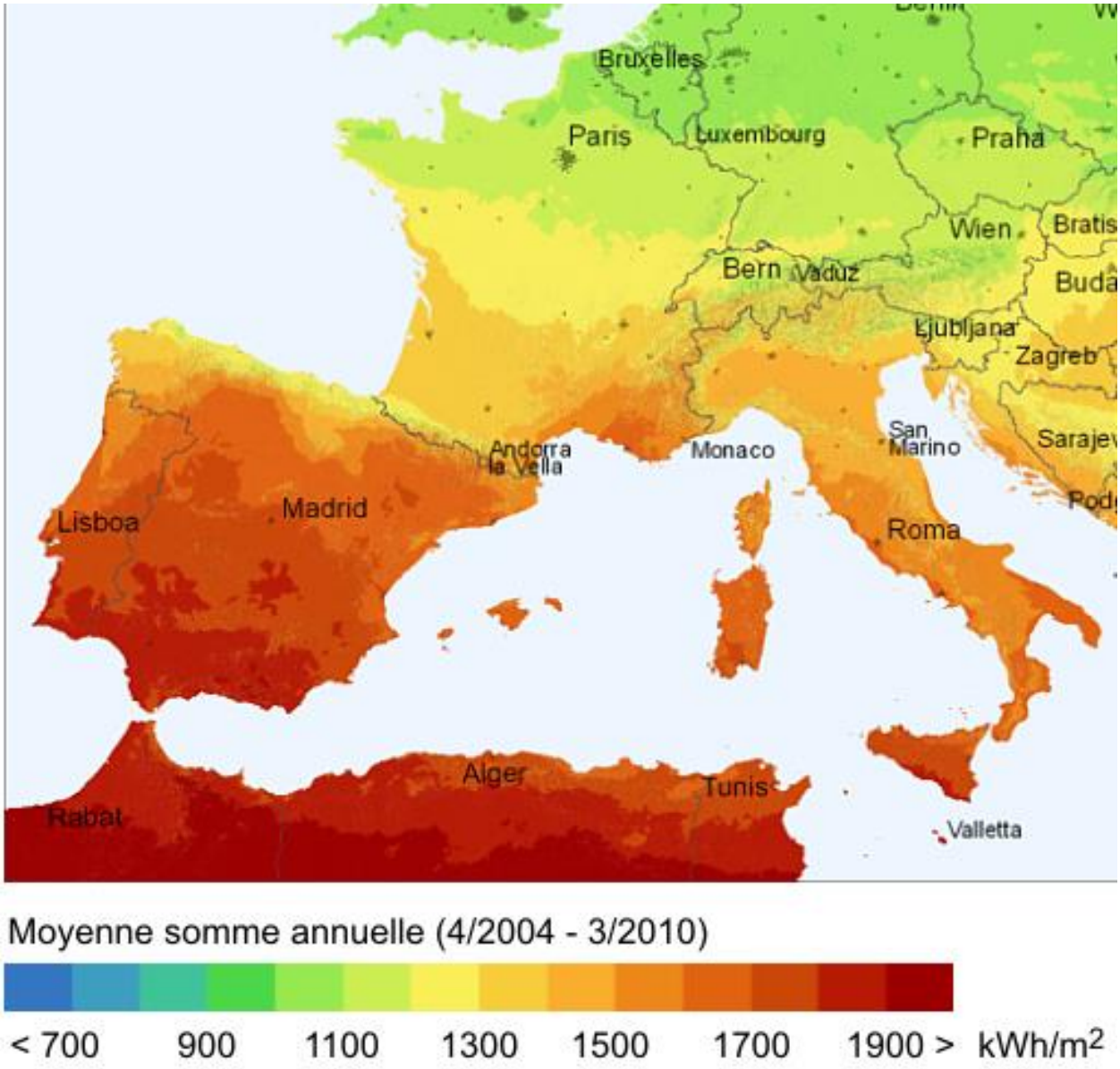
- Rayonnement solaire : un paramètre très influent sur la fonte ou le maintien de la neige. La quantité de rayonnement solaire reçue peut aller du simple au quadruple entre l'hiver et l'été dans les montagnes françaises : entre une journée d'hiver ou le soleil se lève tard, se couche tôt à cause des montagnes qui le cachent, et monte à moins de 30 degrés dans le ciel, le tout sur une journée courte, et d'autre part une journée d'été longue, ou il se lève tôt et se couche tard, est moins caché par des montagnes, et monte vite à plus de 70 degrés dans le ciel, on comprend la différence considérable d'énergie solaire reçue entre une journée de beau temps de fin décembre, et de fin juin. La neige a un albédo de 0,5 à 0,7 et réfléchit environ 70% des rayonnements solaires. Bien que la neige plus « sale », vieille, couverte de poussières, de plantes mortes des mois d'avril, mai et juin soit plus foncée et absorbe plus d'énergie solaire, je vais faire l'hypothèse que son albédo et le même tout au long de l'année. La France reçoit selon les régions 1000 à 1600 kWh/m² à l'horizontale (voir carte), compte tenu des couvertures nuageuses. Il sera difficile de calculer précisément chaque jour de l'année pour chaque station l'énergie reçue. Je n'ai pas tenu compte de la couverture nuageuse également, puisque le paramètre « nébulosité » est déjà mesuré séparément. Aussi je me contenterai uniquement du rayonnement reçu sans nuages, quand le soleil est au plus haut dans le ciel. Et ce sur une surface plate. en utilisant la latitude de la station et la date, j'ai cherché à calculer une valeur proportionnelle à l'énergie reçue quand le soleil est au plus haut dans le ciel.. La valeur est de 1 quand le soleil est au zenith, à 0 quand il est à l'horizon. Il n'y a pas d'intéret à s'approcher des valeurs réelles (évoquées ci-dessus avec la carte) étant donné que les modèles de réseaux de neurones vont, dans tous les cas, remettre le maximum de rayonnement solaire de nos données à 1 et le minimum à zéro. J'ai utilisé les formules suivantes : min_ray est le rayonnement reçu le jour ou le soleil est au plus bas dans le ciel à son maximum (~21 décembre) et max_ray quand il est au plus haut (~21 juin), pour une latitude donnée. 23,5 est l'inclinaison de la Terre, d'où la différence en degrés avec la latitude (qui est autour de 45° en France métropolitaine)

$$min\_ray = \cos\left(2 * \pi * \frac{latitude - 23.5}{360}\right)$$

$$max\_ray = \cos\left(2 * \pi * \frac{latitude + 23.5}{360}\right)$$

Ensuite, je calcule, en utilisant ces minimum et maximum, la quantité reçue en fonction du jour dans l'année. 172 est utilisé pour recaler le maximum (21 juin) sur le premier jour : 172 est le nombre de jours séparant le 1er janvier du 21 juin :

$$Ray\ sol = \frac{max\_ray + min\_ray}{2} + \cos\left(2 * \pi * \frac{day - 172}{365{,}25}\right) * \frac{\text{max\_ray} - \text{min\_ray}}{2}$$



Source : Solargis. http://solargis.info : Somme moyenne annuelle d'énergie reçue en k>h/m²

- Versant de la station : j'ai pour cela simplement regardé le versant sur les cartes Geoportail IGN pour donner l'orientation de la pente, à l'aide des lignes de niveau. J'ai donné un chiffre de 0 à 1, 1 étant un versant plein sud, 0 un versant plein nord, 0,5 en plein est ou plein ouest. Ce sont bien sûr des valeurs approximatives et obtenues à l'œil, et c'est tout à fait critiquable. Dans la version actuelle, je n'ai pas tenu compte du versant

Avec l'ajout des deux paramètres ci-dessus, on voit qu'il y a des tentatives de mieux matérialiser la quantité d'énergie solaire reçue par le sol. J'ai ainsi la nébulosité, j'ai ajouté le rayonnement en fonction de la position du soleil dans l'année, et la pente. Il vaudrait mieux faire une combinaison des trois (multiplication probablement) pour avoir une approximation de l'énergie solaire reçue.

III) APPROCHE DU MODELE ET TRAITEMENTS

Avant d'aller vers le modèle, il faut évidemment regarder la complétude et la qualité des données. Bien sûr, il y a de nombreuses données manquantes. Je ne sais pas comment fonctionnent les stations de mesures : elles sont très probablement composées de thermomètres, anémomètres et de nombreux autres instruments de mesure et de calculateurs pouvant fournir l'étendue de paramètres vus ci-dessus. Les données manquantes viennent peut-être d'erreurs de calcul, de capteurs en panne, de non validation dans les calculs, de débranchement de la station (hors hiver notamment), ou ont d'autres causes. J'ai ainsi remarqué :

- La complétude par paramètre est donnée en dernière colonne dans le tableau des paramètres dans la partie précédente
- Les paramètres sélectionnés pour aller dans le modèle sont listés et expliqués dans la partie précédente
- Les mesures sont majoritairement concentrées sur la période hivernale. En général de décembre à avril. Certaines stations émettent en dehors de ces périodes
- Probablement à cause du confinement en France lié à la crise sanitaire en 2020 et 2021, la saison 2020-2021 est peu complète comparée à d'autres années. De même, à partir de mi-mars 2020, date du début du premier confinement en France, la majorité des stations ont arrêté d'émettre. Cela va compliqué de faire la modélisation de ces années-la.
- Le paramètre « perssfrai » (période mesure neige fraiche) doit être traité (voir partie dédiée ci-dessous)
- En fonction de l'heure de la journée ou les mesures sont faites, il faudra traiter cela (voir partie dédiée ci-dessous)
- Il restera encore quelques cas spécifiques à traiter (valeurs aberrantes, doublons..) (voir partie dédiée ci-dessous)

B) PREMIERS TRAITEMENTS

Comme dans la plupart des cas de data science, il y a des premiers traitements à faire sur les données : suppression des doublons, des dates et des stations manquantes, conversion au bon format des dates et des paramètres, jointure données de mesure / données station (altitude, latitude, longitude, nom).

C) TRAITEMENT DES DONNEES : « PERIODE MESURE NEIGE FRAICHE »

Bien que Météo-France ne donne aucune information sur ce paramètre, on remarque qu'il n'y a quasiment que quatre valeurs: 10, 50, 99 et NA ('mq' dans les données) et il apparait qu'il est un indicateur de la complétude des données de la mesure (voir tableau ci-dessous). On dirait que ce paramètre est une sorte de code indiquant les disponibilités des diverses données. On peut voir la complétude paramètre ci-dessous :

	-10	-50	99	mq (manquant)
Latitude	0,854698	0,935442	0,907458	0,844301
Longitude	0,854698	0,935442	0,907458	0,844301
Altitude	0,854698	0,935442	0,907458	0,844301
Nom	0,854698	0,935442	0,907458	0,844301
numer_sta	1	1	1	1
date	1	1	1	1
datetime	1	1	1	1
ff	0,981651	0,996053	0,971679	0,959772
t	0,978261	0,986144	0,956363	0,955142
td	0,739034	0,89548	0,822003	0,612763
u	0,73982	0,89645	0,823391	0,613456
n	0,983678	0,996927	0,967469	0,973588
nbas	0,969814	0,994134	0,962195	0,930042
rr24	0,606123	0	0,933129	0,581308
tn24	0,514958	0	0,866354	0,471266
tx24	0,513779	0	0,864556	0,469932
ht_neige	0,994536	0,993627	0,992785	0,927531
ssfrai	1	1	1	0
perssfrai	1	1	1	0
hour	1	1	1	1
year	1	1	1	1

Et qui étonnamment se répartit aussi très inégalement selon les années :

year	-99	-50	-10	-50	-60	-99	mq
1996	0	0	0	0	0	0	22747
1997	0	0	0	0	0	0	23764
1998	0	0	0	0	0	0	26600
1999	0	0	0	0	0	0	28917
2000	0	0	0	0	0	0	29432
2001	0	0	0	0	0	0	28678
2002	0	0	0	0	0	0	29515
2003	0	0	4098	0	0	0	23644
2004	0	0	29212	0	0	0	995
2005	0	0	29476	0	1	0	1247
2006	0	0	30608	0	2	0	1106
2007	0	0	30686	0	9	0	827
2008	0	0	5684	8558	1	15210	1480

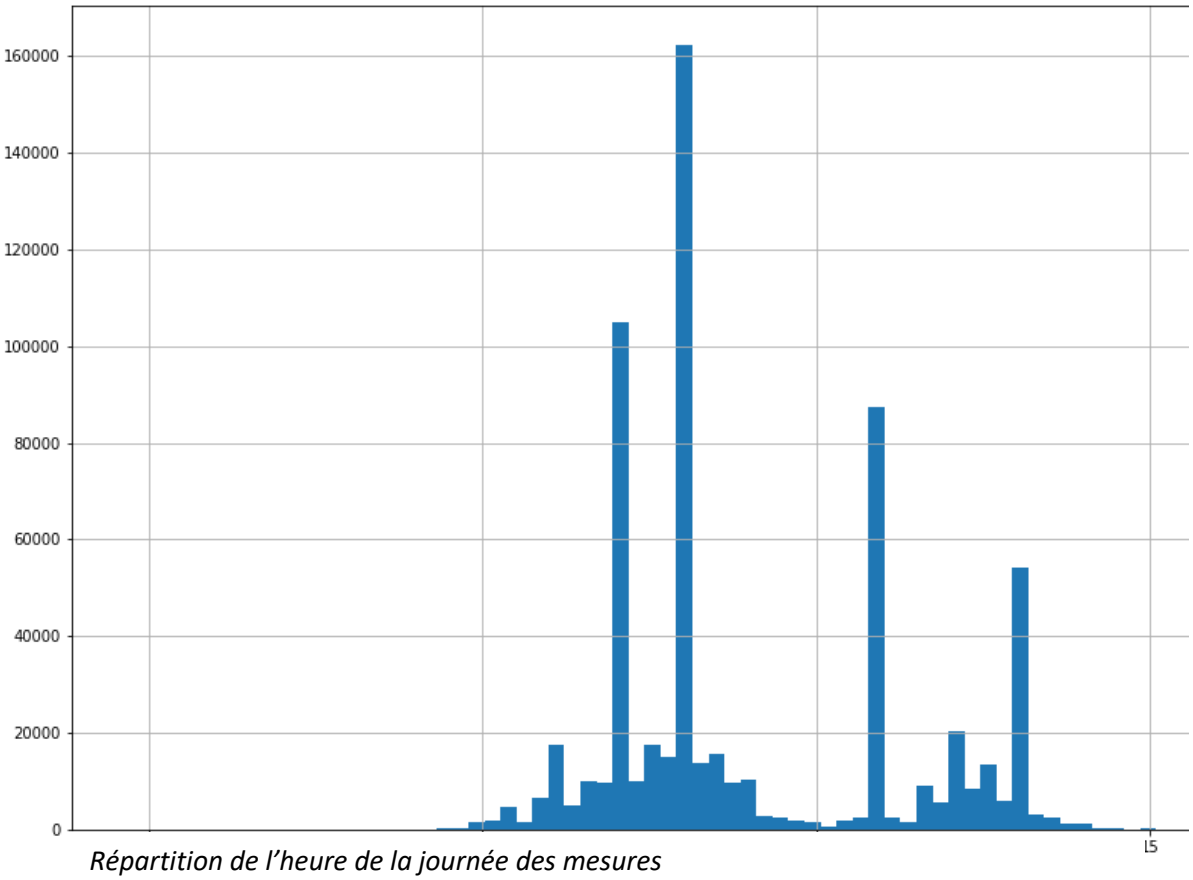
2009	0	0	0	10287	0	18920	1184
2010	0	0	0	10461	0	18834	1084
2011	0	0	0	9338	0	17403	940
2012	0	0	0	8923	0	17323	794
2013	0	0	0	8373	0	16805	1298
2014	0	0	0	7302	0	15009	1626
2015	0	0	0	6699	0	14669	390
2016	0	0	0	6625	0	15478	422
2017	0	0	0	5046	0	14260	345
2018	0	0	0	4630	0	12144	265
2019	0	0	0	3547	0	10080	244
2020	49	10	0	2382	0	7821	196
2021	0	0	0	567	0	3424	110
TOTAL	**49**	**10**	**129764**	**92738**	**13**	**197380**	**227850**

Compte tenu de ce qu'on a la, je décide de supprimer toutes les mesures dont le paramètre « période mesure neige fraiche » est de « -50 » car il y manque la totalité des températures et des précipitations, très importantes pour le modèle.

D) TRAITEMENT DES DONNEES : HEURE DE LA JOURNEE

Si l'on trace la répartition des heures de la journée (voir histogramme ci-dessous), on remarque qu'il y a deux parties : une le matin avant 10h, et une après 10h. Une grande majorité de ces mesures sont effectuées à 8h00, à 9h00 et à 13h00. Compte tenu du fait que :

- les paramètres météorologiques varient fortement en montagne entre le matin et l'après-midi. Il vaut mieux mettre dans une série temporelle les mesures effectuées à peu près au même



Répartition de l'heure de la journée des mesures

moment de la journée. Cela permettra aussi, pour les paramètres calculant des valeurs sur les dernières 24h (températures,

- plus des deux tiers des couples [station, jour] ont au moins 2 mesures (quasiment toujours une le matin et une l'après-midi). Or une seule mesure par jour est suffisante
- Je décide compte tenu de cela de ne garder que les données avant 10 heures du matin. Il devrait y avoir relativement peu de pertes de données.
- le « période mesure neige fraiche » = « -50 » qu'on a voulu supprimer (voir point précédent) est plutôt situées sur les horaires après 10h

E) DERNIERS TRAITEMENTS

Après avoir appliqué les deux traitements abordés ci-dessus, il reste quelques dernières étapes, qui concernent peu de données mais qui doivent être faites :

- Il arrive qu'il reste des données en double pour un couple (station, jour), malgré le filtre de 10h du matin. Dans ce cas, on prend celle des deux qui est la plus complète, ou alors la première.
- Régulièrement, pour une raison inconnue, dans un couple (station, jour) avec deux mesures, l'une présente une hauteur de neige aberrante. Evidemment, on supprimer celle-ci dans ce cas
- Pour la hauteur de neige, il arrive qu'il y ait une erreur de mesure ou de calcul : la valeur de la hauteur de neige peut s'avérer complètement en dehors de la tendance. Dans ce cas, je supprime cette valeur aberrante et j'interpole à partir des jours précédents et suivants

Ces traitements concernent peu de données (quelques centaines à peine) mais se sont avérés assez éprouvants à faire en terme de code Python.

F) BATCHS A GARDER

Comme j'ai dit précédemment, les mesures sont surtout concentrées entre décembre et avril (voir histogramme ci-dessous). Elles commencent et se terminent à des jours différents selon les stations et les années. A l'origine de ce projet, mon idée était surtout de modéliser les périodes d'avril à juillet, avec la complexe fonte des neiges expliquée en introduction. Malheureusement, les données sont donc manquantes pour une bonne partie de cette période. Il va falloir logiquement se plier à ce que nous imposent la disponibilité des données : la période de décembre à avril chaque année. Et il faudra en plus filtrer sur une partie seulement des 144 stations * 24 saisons = 3456 stations/saisons possibles.

Compte tenu de cela, on va découper en batchs, pour le moment de périodes égales, et pour les saisons suffisamment complètes.

En observant la complétude des données j'ai finalement choisi de prendre (parfois de manière un peu hasardeuse) :

- des batchs qui s'étaleraient du 15 décembre au 15 avril inclus, soit des batchs de **122 jours**. Pour les années bissextiles, cela s'étale du 15 décembre au 14 avril.
- garder ceux dont il y a plus de 80% des jours qui comportent une mesure, soit 0,8*122 = 98 jours au moins
- garder ceux dont tous les paramètres sont complets à au moins 70% (y compris si je compte les mesures manquantes

J'obtiens alors, après avoir fait ces filtres, 765 batches sur lesquels je peux faire mon apprentissage

Ci-dessous, deux tableaux donnant les années (l'année est celle du mois de décembre, donc du début du batch) les plus représentées, et les stations les plus représentées dans les 765 batchs :

Les années les plus représentées sont étonnamment situées dans les années 2000. Etonnamment, les années récentes ne sont pas les plus remplies, du moins pas suffisamment pour en faire des batchs. 2019-2020 n'a aucun batch, suite à l'interruption des envois de mesures de la plupart des stations suite au confinement en France de mi-mars 2020. 2020-2021 n'est présent qu'une fois (station de Ceillac)

Pour les stations les plus représentées, je n'ai pas d'explication pour la complétude des données de telle ou de telle station avec sa conséquence sur les batchs.

La Rosiere	16
MEGEVE AUXI	15
Meribel Mottaret	15
Auron	15
Valmorel	15
CEILLAC_NIVO	15
Flaine	15
ISOLA	15
Tignes	14
Courchevel	14
Val d Isere Joseray	14
PUY ST VINCENT 160(	14
Les Menuires	14
St Paul d Oueil	13
St Christophe	13
Super_Devoluy	13
Ax Bonascre	13
Serre Chevalier	13
.es 2 Alpes (Toura NE	13
Iraty	13
Portes en Valgauder	13
Luz Ardiden	12
L Alpe d Huez (SATA)	12

2008	68
2009	67
2006	66
2003	62
2005	60
2004	59
2010	56
2012	52
2011	51
2007	51
2015	37
2017	34
2016	30
2013	27
2014	24
2018	20
2020	1

G) REMPLISSAGE DES TROUS

Après avoir filtré sur les batchs, il reste malgré tout un certain nombre de trous dans les données. Pour l'apprentissage, tous les bachs doivent être remplis. CI-dessous le taux de complétude de chacun des paramètres sur les données 765 batchs (=122*765 = 93330 lignes) :

date	1
datetime	0,907639559
_vent_moy_10min_m/s	0,904425158
temperature	0,904018001
point_de_rosee	0,811078967
humidite	0,81163613
nebulosite	0,905185899
nebulosite_etage_inf	0,903900139
precipitations_24h	0,887142398
temperature_min_24h	0,891867567
temperature_max_24h	0,889885353
hauteur_neige	0,902517947
hauteur_neige_fraiche	0,903728705

Je les ai rempli ainsi :

- Hauteur de neige : probablement le paramètre le plus critique. Comme on a vu, il est plutôt bien rempli. Le plus simple et le plus réaliste ici est de faire une interpolation.
- Hauteur de neige fraiche : un paramètre critique aussi, car il joue directement sur la hausse de la hauteur de neige. Les jours avec chute de neige (donc avec la valeur>0) étant peu nombreux, et la valeur difficile à déterminer en fonction du jour précédent ou suivant, j'ai pris le risque de remplir les trous par zéro
- Température minimum 24h : interpolation linéaire, le plus simple et réaliste
- Température maximum 24h : interpolation linéaire, le plus simple et réaliste
- Altitude de la station : toujours rempli
- Vitesse Moyenne du vent : mise à zéro
- Nébulosité : correspondant à la couche nuageuse, c'est une valeur difficile à déterminer. J'ai pris le risque d'interpoler
- Précipitations : même raisonnement que pour la hauteur de neige fraiche : on remplace par zéro.

Un problème qui se pose malheureusement sur plusieurs batchs, c'est lorsque les données incomplètes se concentrent tout au début (décembre) ou tout à la fin (avril), avec respectivement la première mesure commençant plusieurs jours après le début du batch, et la dernière mesure étant bien avant la fin du batch : l'interpolation n'est pas possible, je fais une extrapolation en mettant la même valeur que la première/dernière, pour tous les paramètres qui sont interpolés en temps normal. C'est surement source d'imprécisions dans les données

Nombre de mesures par mois de 1996 (en bas) à 2020 (en haut), par ordre chronologique. Comme on peut voir, les mois les plus riches en mesures sont Janvier, février, mars. Décembre et avril sont remplis à moitié. Janvier 2008 est le mois le plus riche sur l'historique, avec plus de 6000 mesures

IV) MODELISATION

A) MODELISATION

Nous allons utiliser des algorithmes basés sur les réseaux de neurones adaptés aux séries temporelles : les RNN (recurrent neural network). Sur Python, Keras propose plusieurs de ces algorithmes prêts à l'emploi, dont les plus connus :

- RNN (Recurrent neural network) : on devrait s'attendre à avoir des résultats mauvais, étant donné qu'il y a le problème du vanishing gradient
- LSTM (Long-Short Term Memory Cell) : on devrait avoir de meilleurs résultats
- GRU (Gated Recurrent Unit), une variante plus simple du LSTM, ou l'on devrait s'attendre à avoir de meilleurs résultats

Je ne vais pas m'attarder sur le fonctionnement de ces algorithmes, ce ne sont pas les articles qui manquent sur internet sur le sujet. Quelques liens sont mis dans les références et articles tout à la fin.

Les apprentissages se font donc dans ces conditions :

- 765 batches, de longueur 122 chacun, avec 1Y (hauteur de neige), 11 valeurs X
- 3 algorithmes de RNN utilisés : LSTM(20) + un MLP , Simple RNN (20) + un MLP et GRU(20) + un MLP
- Pour les prédictions, la hauteur de neige prédite pour chaque modèle est recalée en mettant au premier jour la valeur prédite égale à la valeur réelle ce jour-là (voir dans les graphiques, au début des courbes tout à gauche)
- 1000 épochs
- Train : sélection aléatoire de 90% des 765 batchs. Test : les 10% restants
- Optimiser : Adam
- Loss : Mean Squared Error

B) RESULTATS

J'ai refait passer Y (=hauteur de neige= de [0 ;1] (pour l'apprentissage) à [0 ; 3,75]. J'ai calculé, pour chaque batch, la différence moyenne entre la valeur Y prédite et la valeur Y réelle :

$$difference\ moyenne = \frac{1}{122}\sum_{j=1}^{122} abs(Ypred_j - Yreel_j)$$

J'ai également calculé l'écart-type moyen, permettant de voir si la différence Y prédit / Y réel est stable ou pas : on verra plus bas qu'il y a de nombreux cas avec une différence stable, autrement dit avec des courbes qui sont « en parallèle ». Les résultats sont :

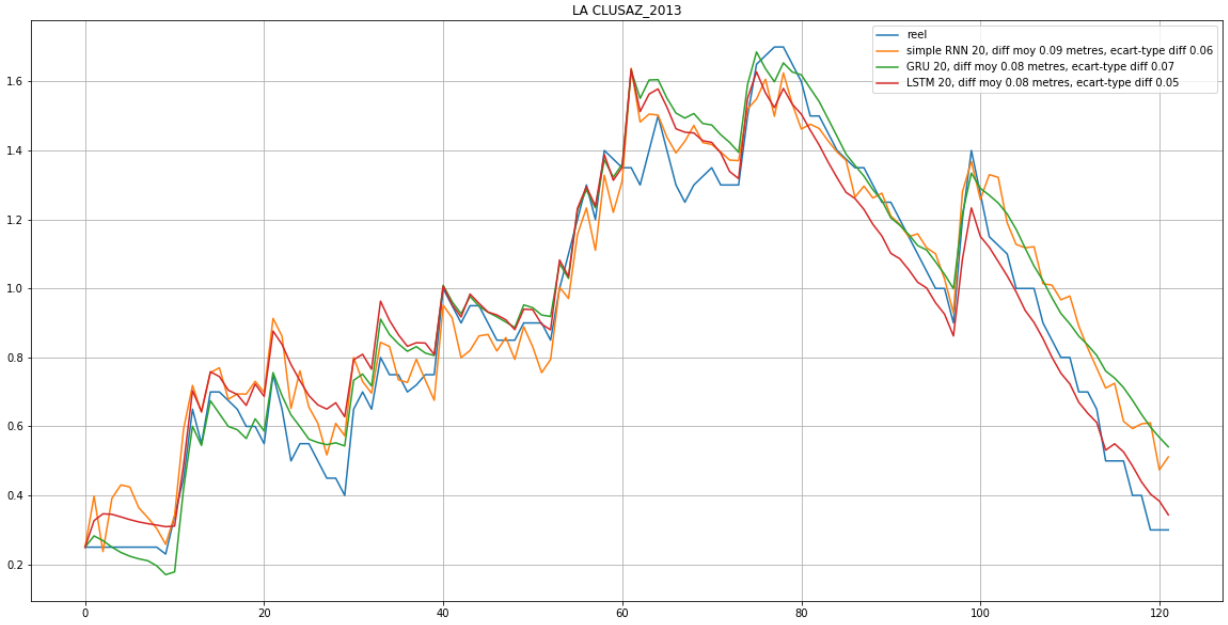
Nom modèle	Loss	Différence moyenne Y prédit / Y réel (m) sur Test baches	
Simple RNN 20 units	0,0066	0,23m	
LSTM 20 units	0,0051	0,18m	
GRU 20 units	0,0050	0,20m	

Comme on pouvait s'y attendre, simple RNN est le modèle le moins précis à cause des problèmes de vanishing gradient. GRU s'avère être le plus précis ici.

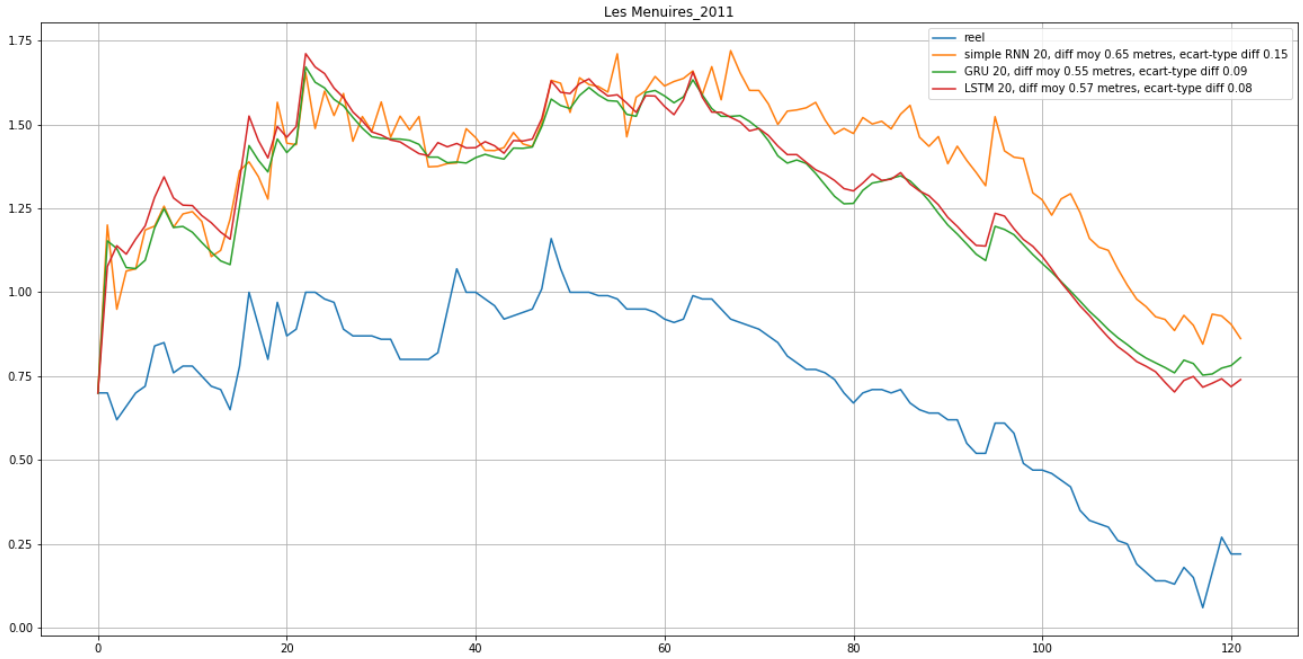
Les graphiques de résultats avec les 4 courbes (réelles, prédiction LSTM, prédiction GRU, prédition RNN) sont dans le dossier *figures/predict_compare_gru_rnn_lstm_with_recalibration_zero*

J'ai tracé les exemples les plus fréquents d'erreurs ou de précisions ci-dessous

- Un exemple de bonne prédiction (La Clusaz 2013-2014) : la hauteur de neige réelle (en bleu) est très proche de la hauteur prédite dans les trois modèles : faible écart-type et faible différence moyenne (moins de 0,07m pour les trois). Ce qui est étonnamment précis

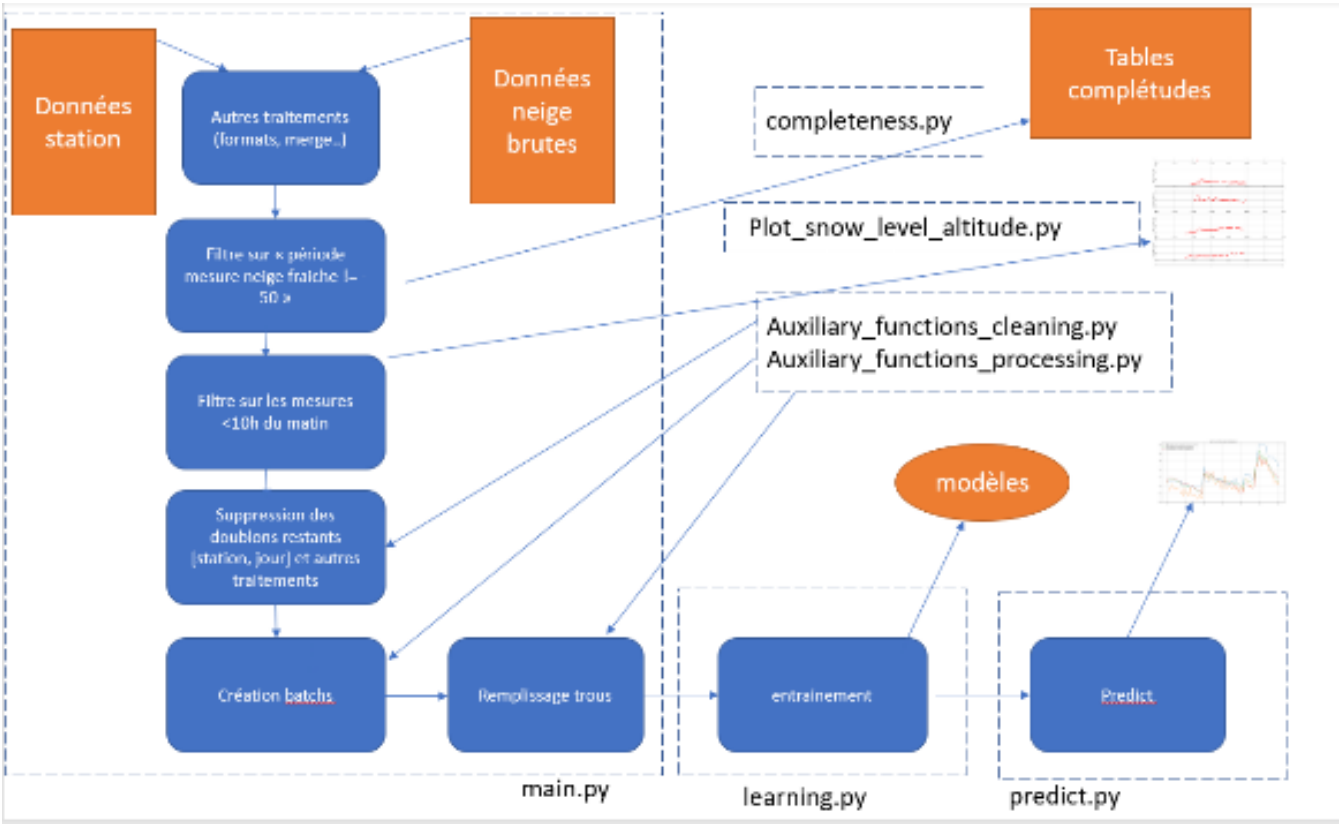


- Un exemple de prédiction ou la forme de la courbe est la même, mais il y a un décalage qui se crée entre les deux (Les Ménuires 2011-2012) : entre la hauteur de neige réelle (en bleu) et les 3 prédictions, il y a quasiment en permanence un décalage de 55 à 65cm (voir différences moyennes en haut à droite du graphique) qui se crée dès les premiers jours. Ce décalage dès le début arrive sur plusieurs batchs. Cependant, on constate que la différence est à peu près égale sur la suite (voir l'écart-type des différences en haut à droite du graphique, qui est faible). Donc si l'on recale sur le 5ème jour par exemple, les graphiques correspondraient à peu près.



V) STRUCTURE DU CODE

Le code, fait intégralement en Python, se trouve dans scripts/ . Il fonctionne schématiquement ainsi :



Il suffira de lancer « main.py », suivi de « learning.py » suivi de « predict.py » pour faire la trame principale d'import, cleaning, processing, apprentissage et prédiction.

VI) PISTES D'AMELIORATION

Evidemment, de nombreuses améliorations peuvent être apportées quant à cette approche et à ce modèle :

- Ajouter d'autres sources de données : les données du col de porte, très complètes et avec un grand historique mais aussi d'autres en France si on en trouve, si elles sont disponibles. On peut également creuser du côté des stations étrangères : Allemagne, Italie, Suisse, si elles sont disponibles. Malgré quelques recherches, je n'en ai pas trouvé.
- Remplir les trous dans les séries temporelles par des méthodes plus mathématiques et techniques que simplement des interpolation linéaires ou la mise à zéro, qui relevaient ici des connaissances physiques ou à des impressions liées à ces paramètres
- Eliminer les valeurs fausses de hauteur de neige par des méthodes plus sures et mathématiques qu'en comparant simplement avec les jours précédents et suivants et en éliminant si la valeur est au-delà d'un certain seuil
- Pour créer les batchs, Nous avions de toute façon 765 batchs ici, ce qui est largement suffisant compte tenu de leur taille et des X. On peut filtrer sur les saisons avec une complétude de plus 90% des mesures par exemple, ce qui donnera moins de batchs, tout en ayant toujours suffisamment pour un bon apprentissage en RNN
- Calculer la taille des batchs optimaux et non pas prendre « au feeling » du 15 décembre au 15 avril de chaque année.
- Garder plus de paramètres (y compris les plus inattendus) et voir le résultat sur les modèles
- Calculer un paramètre de rayonnement reçu par le sol en combinant la pente, la hauteur du soleil, ainsi que la nébulosité. Au lieu de garder les trois séparément.
- Faire les modèles de manière un peu différente : pour faire monter la hauteur de neige, se contenter d'une addition de la « hauteur neige fraiche » à la hauteur de neige j-1 uniquement. Et empêcher en dehors de cela la courbe de « monter ». Et ensuite ne laisser le modèle de Deep Learning tourner que pour la diminution de la hauteur de neige.
- Empêcher le modèle de prédire une valeur de hauteur de neige plus basse que zéro : cela arrive quelques fois, notamment pour des stations / années ou le niveau de neige est faible.

VII) REFERENCES :

Un grand merci à Clémence, qui a aidé dans ce projet, plus particulièrement dans les étapes difficiles du code et dans la validation.

SOURCES :

- Site opendata Météo-France (données, table des paramètres, liste des stations) :https://donneespubliques.meteofrance.fr/?fond=produit&id_produit=94&id_rubrique=32

- Table des paramètres : https://donneespubliques.meteofrance.fr/client/document/doc_parametres_nivo_197.pdf

- Cartes Geoportail : https://www.geoportail.gouv.fr/carte

- Article dont je me suis inspiré pour tracer la carte : https://makina-corpus.com/blog/metier/2019/python-carto

- GRU, LSTM, RNN :
- https://colah.github.io/posts/2015-08-Understanding-LSTMs/
- https://penseeartificielle.fr/comprendre-lstm-gru-fonctionnement-schema/