



**UFR Sciences  
Département de géographie**

**Laboratoire des Sciences du  
Climat et de l'Environnement**

## **MASTER "GESTION DE L'ENVIRONNEMENT"**

### **M1**

Parcours "Pollution Atmosphérique, changement climatique, Impacts sanitaires, énergies  
Renouvelables (AIR)"

# **Combinaison de simulations climatiques par "machine learning"**

**Mats GARVIK**

**Direction de recherche : Dr. Mathieu VRAC, Dr. Soulivanh THAO, Pr. Grégoire MARIÉTHOZ**

**21 juin 2019**

# Table des matières

0.1 Remerciements . . . . .	2
0.2 Présentation du laboratoire . . . . .	2
<b>1 Combinaison de simulations climatiques par "machine learning"</b>	<b>3</b>
1.1 Introduction . . . . .	3
1.2 Objectifs . . . . .	4
<b>2 Données et méthodes de l'algorithme du graph cut</b>	<b>4</b>
2.1 Données . . . . .	4
2.1.1 Modèles à combiner . . . . .	4
2.1.2 Données de référence . . . . .	6
2.2 Méthodologie . . . . .	6
2.3 Choix d'attribution des nœuds . . . . .	10
2.3.1 No_st . . . . .	10
2.3.2 St_smallest . . . . .	10
<b>3 Résultats</b>	<b>12</b>
3.1 Températures . . . . .	12
3.1.1 Attribution des nœuds . . . . .	12
3.1.2 Nombre de points attribués . . . . .	15
3.1.3 Échelles temporelles . . . . .	16
3.1.4 Comparaison et projection . . . . .	18
3.2 Précipitations . . . . .	21
3.2.1 Attribution des nœuds . . . . .	21
3.2.2 Nombre de points attribués . . . . .	21
3.2.3 Échelles temporelles . . . . .	22
3.2.4 Comparaison et projection . . . . .	23
<b>4 Conclusions et perspectives</b>	<b>25</b>
4.1 Apport personnel . . . . .	25
<b>5 Bibliographie</b>	<b>26</b>
<b>Annexes</b>	<b>27</b>
<b>A Code</b>	<b>27</b>
<b>B Cartes</b>	<b>33</b>
<b>C Plots</b>	<b>41</b>

## 0.1 Remerciements

En premier lieu, je tiens à remercier Mathieu VRAC et Soulivanh THAO, mes référents de stage au LSCE, de m'avoir offert l'opportunité de faire ce stage, bien que mon profil ne correspondait pas forcément aux critères initiaux, et d'avoir répondu à mes nombreuses questions. Je remercie également Grégoire MARIÉTHOZ, mon troisième référent de stage, qui m'a indiqué des pistes qui se révèlèrent clés lors de ce stage.

De plus, je souhaite remercier l'intégralité de l'équipe ESTIMR pour leur accueil très chaleureux durant cette première partie de stage.

Enfin, je remercie évidemment mes proches, pour leur soutien et leurs encouragements durant le stage.

## 0.2 Présentation du laboratoire

Le LSCE est une unité mixte de recherche (UMR 8212) entre le CEA (Commissariat à l'énergie atomique et aux énergies alternatives), le CNRS (Centre national de la recherche scientifique) et l'Université de Versailles Saint-Quentin (UVSQ), localisé sur deux sites (Campus du CEA-Orme des Merisiers et du CNRS Gif-sur-Yvette). Il fait partie de l'Institut Pierre Simon Laplace (IPSL). Le LSCE regroupe environ 300 chercheurs, ingénieurs et agents administratifs dont 150 personnels permanents issus des 3 tutelles et plusieurs dizaines d'étudiants thésitifs.

Depuis janvier 2015, le LSCE s'organise en trois thèmes scientifiques, dont Climat et Cycles - Modélisation de leurs variabilités et de leurs interactions, thème dont fait partie l'équipe ESTIMR (Extrêmes : Statistiques, Impacts et Régionalisation). L'équipe ESTIMR a pour objectifs principaux : la compréhension et la modélisation de la variabilité climatique et environnementale à différentes échelles spatiales – depuis les très grandes structures liées à la dynamique atmosphérique jusqu'à des phénomènes très locaux – et à différentes échelles temporelles – pour l'étude de climats passés, des processus présents et des évolutions futures. Dans ce cadre très vaste, l'une des forces de l'équipe ESTIMR repose sur l'utilisation et le développement de modèles statistiques de l'état-de-l'art adaptés aux problématiques climatiques, grâce à une interaction multidisciplinaire soutenue entre climatologie, modélisation, physique et statistiques. L'équipe ESTIMR se trouve donc à l'interface entre la recherche en climatologie et certaines questions sociétales (liées aux impacts du changement climatique) en développant des outils statistiques servant de passerelles entre les deux, et utilisables par la communauté scientifique et la société.

# 1 Combinaison de simulations climatiques par "machine learning"

---

**Abstract.** Bien que les modèles climatiques soient de plus en plus précis, il est nécessaire de les ajuster en les comparant à des réanalyses et en corrigeant leurs biais. Il existe différentes techniques, notamment la combinaison de simulations climatiques. Cette dernière est généralement faite par une approche ensembliste par moyenne. Cependant, une autre alternative est envisagée, correspondant à un choix de modèle. Ce choix est effectué grâce au graph cut, une méthode de machine learning utilisée en géostatistiques ainsi qu'en synthèse d'image, mais jamais appliquée à des simulations climatiques. L'objectif de cette méthode est de choisir entre deux modèles, pour chaque point de grille, le meilleur des deux, tout en gardant une continuité spatiale grâce à l'algorithme de Ford et Fulkerson (Ford and Fulkerson, 1956). Cette méthode permet de combiner le meilleur des deux modèles, et ainsi d'optimiser le résultat. Le graph cut présente un biais moyen plus faible que les modèles, qu'ils soient pris séparément ou combinés par moyenne, pour la France et l'Europe, que ce soit sur des données annuelles ou saisonnières.

---

## 1.1 Introduction

Les modèles climatiques et leurs projections sont des outils de plus en plus importants, dans la mesure où le changement climatique est renforcé par les activités anthropiques (Solomon et al., 2007). Cependant, bien que ces modèles soient améliorés régulièrement, il n'existe pas de modèle qui correspond parfaitement à la réalité, à cause par exemple d'une résolution trop faible (bien qu'il existe des alternatives pour un domaine régional ou local, notamment avec les méthodes de *downscaling*), à cause d'approximations dans les paramétrisation, par une mauvaise estimation du forçage (Sanderson et al., 2008) ou d'autres facteurs. Ces modèles masquent donc les différences à grande échelle. De ce fait, ces modèles présentent des biais : des erreurs propres aux modèles que l'on essaie de quantifier en comparant leurs simulations à des observations passées. Il est donc important de corriger ces biais, afin de rendre ces simulations plus proches de la réalité.

Il existe différentes manières de corriger les biais, avec par exemple des corrections statistiques (e.g. Vrac, 2018). Il existe aussi une autre méthodologie, qui consiste à combiner les modèles, en espérant que leurs biais se compensent, et de ce fait, créer un nouveau modèle qui soit plus proche de la réalité. Cette approche permet de faire ressortir le signal, quitte à cacher les disparités des modèles. Le but est d'obtenir de meilleurs résultats qu'avec les modèles pris séparément, même si cela varie en fonction des variables climatiques étudiées. De ce fait, un poids est attribué aux modèles afin d'en faire la moyenne. En général, un poids égal est attribué à chaque modèle, permettant d'obtenir de meilleurs résultats (Auger et al., 2018). Ainsi, pour obtenir un modèle  $CM_{mean}$

issu des moyennes des différents modèles  $CM$ , un poids  $p$  va leur être attribué :

$$CM_{mean} = p_1 * CM_1 + p_2 * CM_2 + \dots + p_n * CM_n \quad (1)$$

Dans ce stage, une autre approche est envisagée, celle du graph cut : une technique de "machine learning" utilisée principalement en géostatistiques (Li et al., 2016) ainsi qu'en synthèse d'image (Kwatra et al., 2003), mais jamais dans un contexte de simulation climatique. L'idée de Kwatra et al. était de considérer qu'on peut mettre sous forme de graphe une image, en considérant que les pixels représentent les nœuds et que la continuité entre deux pixels représenterait l'arête du graphe ainsi que son coût.

Ainsi, il est possible d'appliquer la même approche, en considérant qu'une matrice de variable climatique peut être mise sous la forme d'une image, et donc d'un graphe. Donc, plutôt que de calculer la moyenne des biais de deux modèles de manière uniforme pour chaque point de grille, il va s'agir de choisir, pour chaque point de grille, l'un des modèles, idéalement celui qui a le biais le plus faible. Ainsi, au lieu d'avoir la moyenne de différents biais dans un point, il n'y aura le biais que d'un modèle. Pour ce faire, l'idée est d'identifier les zones de faible biais comme zone de référence pour le graph cut, et de se baser sur un algorithme qui sera décrit en section 2.2. L'idée sera de voir le résultat de la prise en compte de la continuité spatiale, mais aussi de la minimisation des biais.

## 1.2 Objectifs

Afin de répondre aux attentes du stage, plusieurs objectifs ont été définis :

- Mettre en place et appliquer la méthodologie de graph cut sur deux simulations climatiques provenant de l'exercice international "Coupled Model Intercomparison Project" (CMIP). Cette méthodologie sera appliquée sur une période passée (1979-2008) et projetée sur une période future (2041-2070 et 2071-2100).
- Analyser et comparer les différents résultats et approches de la méthodologie entre eux, afin de déterminer une approche optimale.
- Comparer les résultats du graph cut à ceux des approches ensemblistes par moyenne.

# 2 Données et méthodes de l'algorithme du graph cut

## 2.1 Données

### 2.1.1 Modèles à combiner

Afin d'effectuer les tests du graph cut, il était nécessaire de choisir des sorties de modèles climatiques. Pour ce stage, deux modèles ont été utilisés : celui du Centre National de Recherche Météorologique (CNRM) ainsi que celui de l'Institut Pierre Simon Laplace (IPSL).

Le modèle du CNRM correspond au modèle ARPEGE-Climat 5.2.1, issu du modèle global CNRM-CM5, mis en place pour le CMIP5 ("Coupled model intercomparison project") en 2011. Ce modèle possède une résolution allant de 50km en son centre à 100km pour un total de 115680 points de grille sur le globe.

Le modèle de l'IPSL correspond au modèle LMZD4, issu du modèle global IPSL-CM5, mis lui aussi en place pour le CMIP5 en 2011. Ce modèle possède une résolution allant jusqu'à 80km pour un total de 115680 points de grille sur le globe.

Les variables utilisées dans ce stage sont la température et la précipitation, mises sous forme de grille correspondant à celle de la donnée de référence ERA-Interim. Initialement, chaque point contient, pour une variable, des données journalières couvrant une période de 30 ans (1979-2008). Après transformation des 30 années de données en une valeur (de moyenne ou d'écart-type), la moyenne ou l'écart-type de l'ensemble des données de la série temporelle en chaque point de grille, a été calculée (par an ou par saison), donnant ainsi une valeur par point (figure 1).

La méthodologie du graph cut est appliquée sur la période décrite précédemment et les meilleurs résultats seront projetés sur des projections couvrant elles aussi des périodes de 30 ans (2041-2070 et 2071-2100).

Les domaines des tests correspondent approximativement à la France (entre 5.25°W, 40.5°N et 9°E, 54.75°N) et à l'Europe (entre 20.25°W, 34.5°N, et 60°E, 75°N), et représentant respectivement 400 et 5940 points de grille.

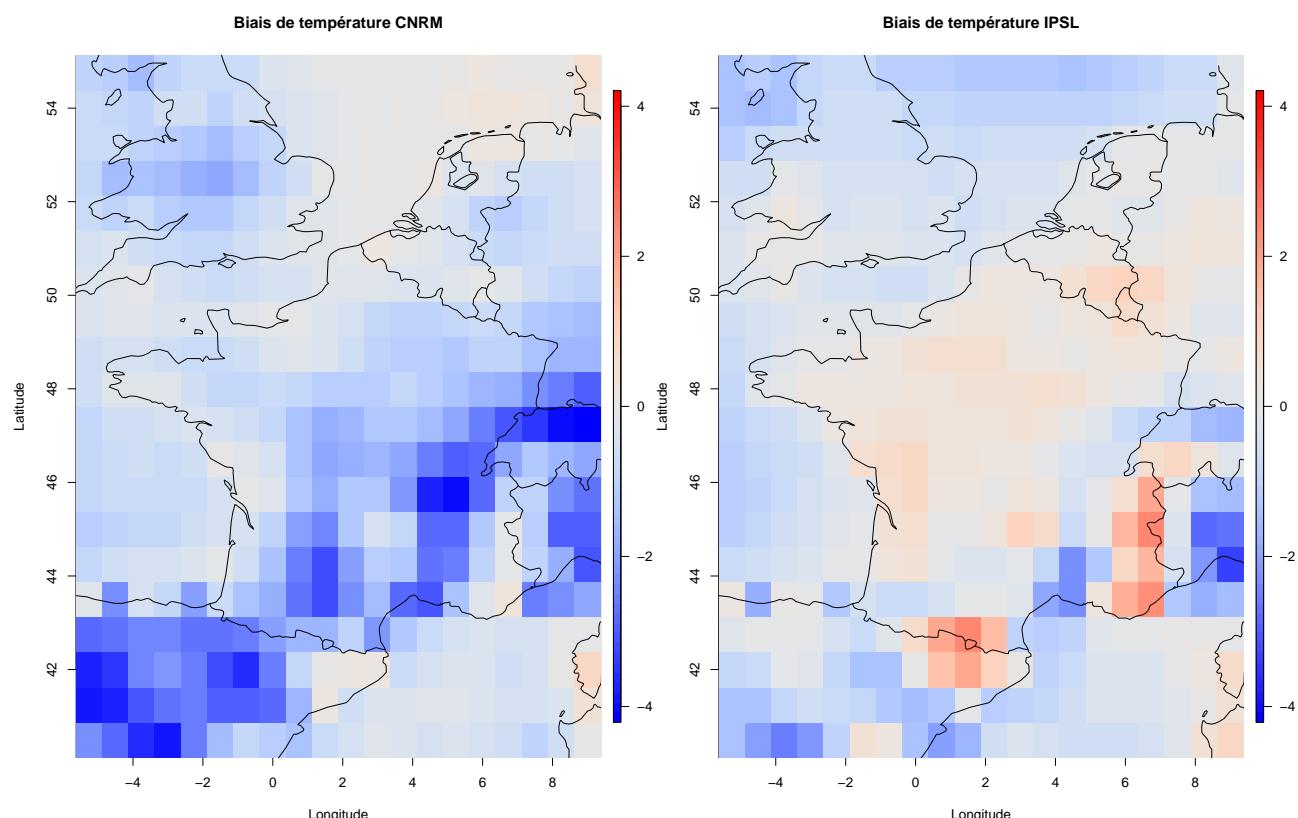


FIGURE 1 – Cartes des biais de température (°C) des modèles CNRM et IPSL en France

### 2.1.2 Données de référence

Afin d'effectuer des tests sur les modèles, il faut comparer leur simulations à des données de références, afin de calculer leurs biais, et ainsi de les optimiser grâce au graph cut. Les données choisies sont celles de la réanalyse ERA-Interim (Berrisford et al. 2011) du Centre européen pour les prévisions météorologiques à moyen terme ("European Centre for Medium-Range Weather Forecasts", ECMWF, en anglais) pour la période citée précédemment. Ce modèle possède une résolution globale d'approximativement 79km, pour un total de 115680 points de grille sur le globe également (figure 2).

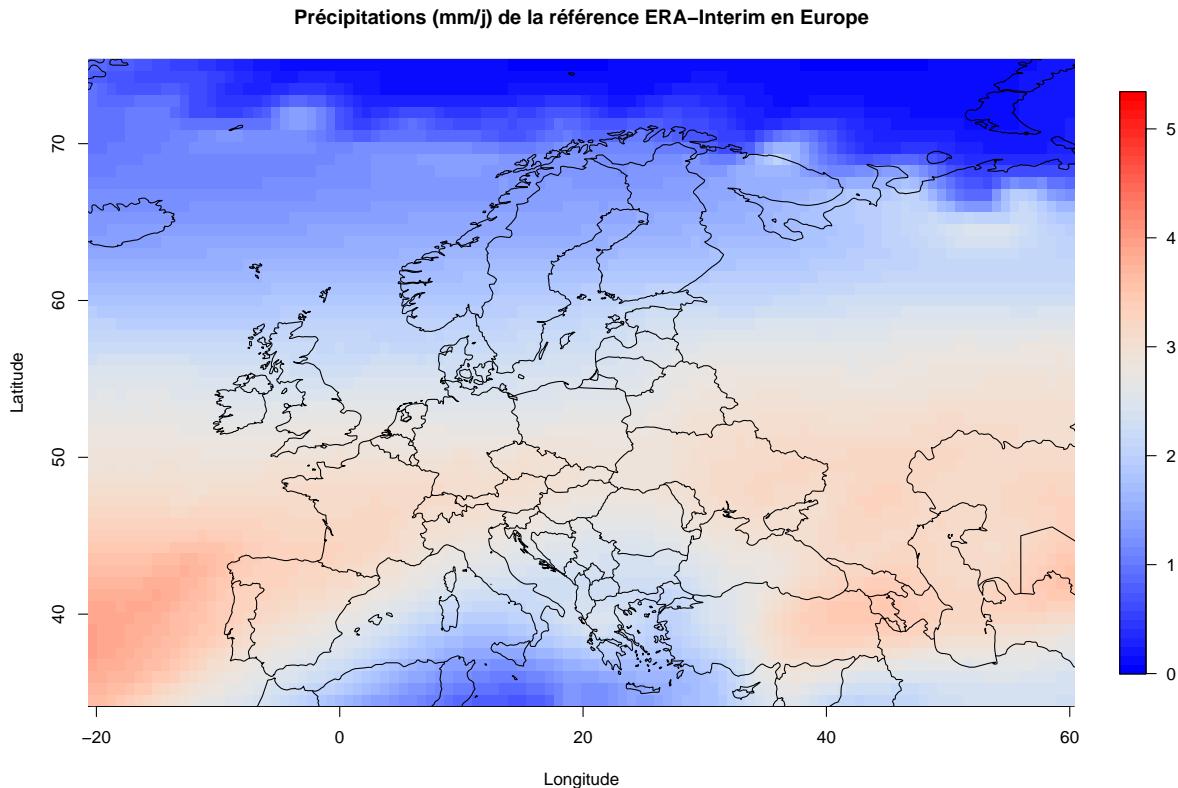


FIGURE 2 – Carte des précipitations moyennes annuelles (mm/j) de la référence ERA-Interim en Europe

## 2.2 Méthodologie

Le graph cut est une méthode de "machine learning" (Boykov et al., 2001), initialement créée pour répondre au théorème de *maxflow-mincut* de Ford de Fulkerson (Ford and Fulkerson, 1956). Ce théorème, dont l'application est présentée en figure 3, consiste à découper en deux de manière optimale un graphe pondéré (l'analogie du *flow* et d'une direction du graphe permet de comprendre le principe, mais il n'y aura ni de *flow* ni de direction dans le travail présenté) est découpé. Le but du graph cut est de conserver le *flow* maximum du graphe, et donc de trouver la coupure dont le coût est minimal. Ainsi, à la suite de cette coupure, les points sont reliés à l'un des deux terminaux du graphe (ceux en bleu au terminal *s* ou *source* et ceux en noir au terminal *t* ou *target*).

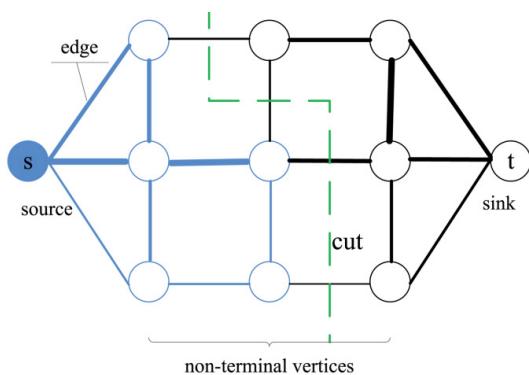


FIGURE 3 – Exemple d'un graphe pondéré et d'un cut (source : Li et al., 2016)

Cette méthode a ensuite été reprise dans divers domaines, comme la synthèse d'image (Kwatra et al., 2003). Le fait de minimiser la coupure entre deux parties d'un graphe a été détourné aux images, permettant ainsi de "coller" des images tout en minimisant les discontinuités entre ces dernières (figure 4).

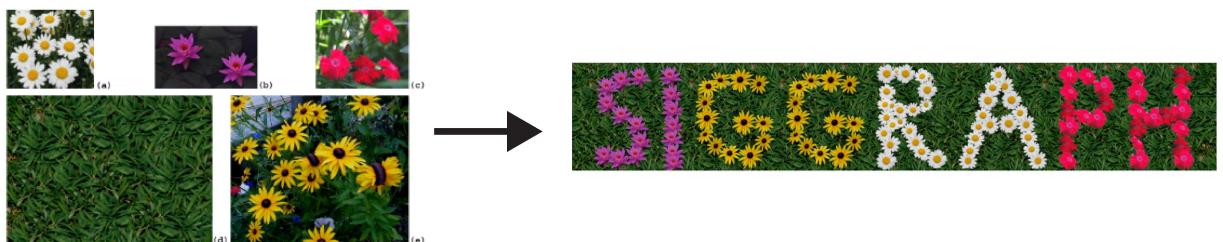


FIGURE 4 – Exemple d'une synthèse d'images par graph cut (source : Kwatra et al., 2003)

En effet, il est possible de transformer une image en un graphe en considérant qu'un pixel correspond à un nœud, et une arête à la jointure de deux pixels (figure 5).

L'idée de ce stage est également de détourner le graph cut, mais cette fois à des simulations climatiques. Il est possible de représenter ces simulations sous forme d'image, et ainsi, par le même procédé, d'appliquer le graph cut. Ainsi, les points de grille correspondront aux nœuds du graphe, et l'homogénéité de la carte sera représentée par le coût d'une arête. Afin d'obtenir ce coût, le voisinage des points de grille a été défini selon le voisinage de Von Neumann (ou 4-neighborhood) avec une distance de Manhattan égale à 1, comme présenté en figure 6).

Le but est de combiner deux simulations de modèles, afin de tirer le meilleur de chacun, et ainsi de minimiser leur biais, tout en gardant une continuité spatiale, grâce à l'algorithme de Ford et Fulkerson. De ce fait, le *flow* correspondra à la continuité spatiale, et sera donc minimisé tout

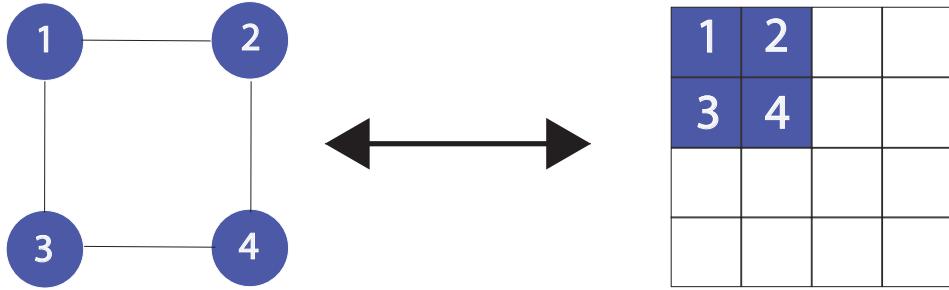


FIGURE 5 – Exemple de transformation d'un graphe en une grille de points

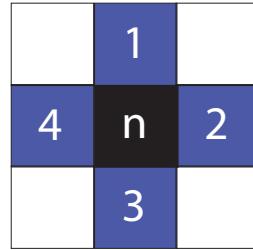


FIGURE 6 – Exemple des voisins adjacents pour un pixel  $n$

en essayant de minimiser les biais, grâce à l'attribution des points aux terminaux du graphe. Ainsi, afin de recréer le graphe à partir des simulations climatiques, il faudra rattacher arbitrairement les points de grille à des points fictifs, représentant les terminaux du graphe (nommés *source* et *target*). Ces terminaux correspondront aux modèles à combiner. Les points qui n'ont pas été rattachés seront donc attribués à l'un des modèles par l'algorithme du graph cut. Enfin, afin d'obtenir un résultat le plus homogène possible, le poids des arêtes sera représenté par la continuité entre deux points de grille. Ainsi, la valeur d'une arête correspondra à la somme de la valeur absolue des biais des deux modèles des deux points de grille adjacents :

$$C_E(u, v, A, B) = \delta(u, A, B) + \delta(v, A, B) \text{ avec } u, v = \text{points de grille et } A, B = \text{simulations} \quad (2)$$

C'est la somme de ces coûts que l'algorithme va minimiser en appliquant la coupure.

Afin d'appliquer le graph cut à des simulations climatiques, une méthodologie doit donc être mise en place, le but étant de passer une matrice de données climatiques sous forme de points de grille à une matrice représentable en graphe, et appliquer l'algorithme du graph cut via le package "opTrees" (Fontenla, 2014). Pour ce faire, plusieurs fonctions et ajustements ont dû être créés pour effectuer cette transformation.

Afin d'avoir des comparaisons pour des tests, 10 échantillons des biais les plus faibles des modèles ont été sélectionnés. Afin d'obtenir une symétrie dans les valeurs, ces données (mises en valeur absolue) ont été classées grâce à une fonction de répartition empirique ("Empirical function repartition", ECDF, en anglais) et les échantillons allant de 5% (correspondant à plus ou moins 2.5 % des biais les plus faibles) à 50%, avec un pas de 5%, ont été sélectionnés.

Cependant, certains points de grille correspondaient à un biais faible pour les deux modèles, hors un point ne peut être relié aux deux terminaux. C'est pourquoi deux approches ont été mises en place et seront décrites en section 2.3.

Maintenant, il faut transformer les matrices des modèles en matrices de graphes, possédant deux colonnes pour les nœuds voisins, et une colonne pour le poids des arêtes. Comme dit précédemment, le poids des arêtes correspond ici à la somme des biais des deux modèles pour les deux points voisins. Une fonction de voisinage a donc été créée, de même pour le coût des arêtes (au lieu de prendre en compte les indices, les valeurs des modèles ont été prises en compte et ont été additionnées comme présenté en figure 7).

1	2		
3	4		

Noeud 1	Noeud 2	Poids
1	2	(p <sub>1</sub> +p <sub>2</sub> )
1	3	(p <sub>1</sub> +p <sub>3</sub> )
2	4	(p <sub>2</sub> +p <sub>4</sub> )

FIGURE 7 – Exemple de transformation d'une grille de points en matrice de voisinage et de poids

Maintenant, la matrice correspond aux critères nécessaires pour la fonction du graph cut (table 1) et cette fonction peut-être lancée. Cette matrice possédera donc un nombre différent de nœuds en fonction de l'échantillon choisi en amont, mais aussi en fonction des approches définies. Après le calcul, l'algorithme attribuera donc chaque point de grille à une simulation.

Nœud 1	Nœud 2	Coût
Nœud source	1	10000
Nœud source	5	10000
...	...	...
880	988	0.01793
880	881	0.28313
881	989	0.08742
...	...	...
Nœud target	5120	10000
Nœud target	5189	10000

TABLE 1 – Exemple d'une matrice utilisable par l'algorithme du graph cut.

## 2.3 Choix d'attribution des nœuds

Pour permettre à l'algorithme de prendre en compte la minimisation des biais, les biais les plus faibles des modèles ont été reliés aux terminaux. Le modèle IPSL, a été défini comme *source* et le modèle CNRM comme *target*. Ainsi, un point de grille possédant un biais faible sera défini comme un noeud relié à un terminal. Cependant, pour certains nœuds, les deux modèles peuvent présenter un biais faible, le problème étant qu'un nœud ne peut pas être rattaché aux deux terminaux. Deux approches ont donc été mises en place. L'une consiste à ne pas attribuer un point ayant un biais faible pour les deux modèles (*no\_st*). L'autre consiste à attribuer un point ayant un biais faible pour les deux modèles à celui ayant le biais le plus faible (*st\_smallest*). Afin que l'algorithme ne fasse pas de coupure entre ces points et les terminaux, puisque le but est de minimiser le coût de la coupure tout en minimisant les biais, un coût beaucoup plus important (10000) que les coûts réels leur ont été attribué (0.40 en moyenne).

### 2.3.1 No\_st

Comme dit précédemment, cette attribution n'est applicable qu'à un point de grille possédant un biais faible pour les deux modèles. Cette approche consiste à n'attribuer à aucun terminal ce point de grille. L'idée derrière cette méthode d'attribution consiste à privilégier les points de grille possédant un biais faible pour un modèle, et un biais moyen voire fort pour l'autre, permettant ainsi d'effacer les fortes différences entre les modèles. De plus, cette méthode permet d'avoir une modification des points de grille attribués, car certains points peuvent disparaître en augmentant le nombre de points attribués.

En effet, un point peut avoir un biais faible pour les deux modèles, tandis qu'il était faible pour un seul modèle en prenant un échantillon plus petit (table 2). Par exemple, en prenant 25% des biais les plus faible, un point de grille  $n$  peut être attribué au modèle IPSL, mais pas au modèle CNRM. Cependant, si l'on passe à 30%, ce point peut lui aussi devenir un point à attribuer au modèle CNRM, et donc ne plus être attribué. Enfin, en ayant un plus faible nombre de points reliés aux terminaux, l'algorithme pourra effectuer des coupures entre des points de grille ayant des biais faibles pour les deux modèles. Ces derniers seront donc reliés par des arêtes ayant un faible coût, et donc le coût total sera faible lui aussi.

### 2.3.2 St\_smallest

Comme pour l'autre méthode d'attribution, celle-ci n'est applicable qu'à un point de grille possédant un biais faible pour les deux modèles. Cependant, plutôt que de ne faire aucune attribution, le point de grille sera attribué au modèle possédant le plus faible biais. L'idée ici est de conserver tous les points de l'échantillon ayant un biais faible, et de sélectionner celui ayant le biais le plus faible. Ainsi, un plus grand nombre de points seront attribués aux terminaux, augmentant proportionnellement avec la taille des échantillons (table 3). Cependant, étant donné qu'une plus grande part des biais faibles seront attribués, la coupure se fera entre des biais moyens voire fort,

Pourcentage de points pris (+/-)	Nombre de points reliés à la source (IPSL)	Nombre de points reliés à la target (CNRM)
2.5	264 (4.44%)	290 (4.88%)
5	487	549
7.5	659	759
10	732	934
12.5	799	1091
15	789	1181
17.5	799	1271
20	782	1312
22.5	793	1269
25	761 ( <b>12.81%</b> )	1227 ( <b>20.66%</b> )

TABLE 2 – Points reliés aux terminaux pour la température moyenne annuelle en Europe (total = 5940 points) avec l’attribution *no\_st*

augmentant ainsi de manière croissante le coût total de la coupure.

Pourcentage de points pris (+/-)	Nombre de points reliés à la source (IPSL)	Nombre de points reliés à la target (CNRM)
2.5	271 (4.56%)	303 (5.10%)
5	513	599
7.5	719	881
10	845	1176
12.5	979	1451
15	1059	1708
17.5	1146	1968
20	1234	2189
22.5	1402	2302
25	1528 ( <b>25.72%</b> )	2426 ( <b>40.84%</b> )

TABLE 3 – Points reliés aux terminaux pour la température moyenne annuelle en Europe (total = 5940 points) avec l’attribution *st\_smallest*

### 3 Résultats

40 tests ont été effectués sur 10 échantillons de points pour la température et pour les précipitations, sous différentes formes et échelles spatio-temporelles :

- moyenne
- écart-type
- annuel
- saison
- France
- Europe

Chaque test présente des similitudes, mais aussi des différences, plus ou moins importantes. Ainsi, l'objectif est de comparer ces tests, afin de ressortir une ou des approches et un ou des tests optimaux, et de les comparer aux approches ensemblistes par moyenne.

#### 3.1 Températures

Comme dit précédemment, des tests ont été fait sur les simulations de températures des modèles, et ils seront étudiés en fonction de l'attribution des points aux biais faibles pour les deux modèles, du nombre de points attribués et, en ce qui concerne les tests sur les saisons, entre l'été et l'hiver. Ces différences sont plus importantes pour l'Europe que pour la France, dû à la taille des domaines, mais elles sont présentes dans les deux zones.

##### 3.1.1 Attribution des nœuds

En France, comme dit précédemment, il existe des différences entre les approches d'attribution, mais avec des ordres de grandeurs minimes. De ce fait, les résultats ne sont pas réellement différents à l'œil nu, mais présentent bel et bien ces différences.

Les premiers tests ont été réalisés sur les températures moyennes annuelles. Pour ces tests, c'est l'approche *no\_st* qui est la plus optimale, bien que les deux approches suivent la même trame, avec notamment une forte erreur pour Q10, qu'on ne retrouve pas pour Q05 par exemple (figure 8). C'est l'attribution d'un point au *target* (CNRM), modèle étant globalement moins bon que le modèle de l'IPSL pour la France, qui fit qu'une grande zone est attribuée au modèle CNRM, augmentant grandement le biais du résultat du graph cut. Cette dégradation montre encore une fois que le graph cut privilégie la continuité spatiale et la minimisation du coût total du cut à la minimisation des biais.

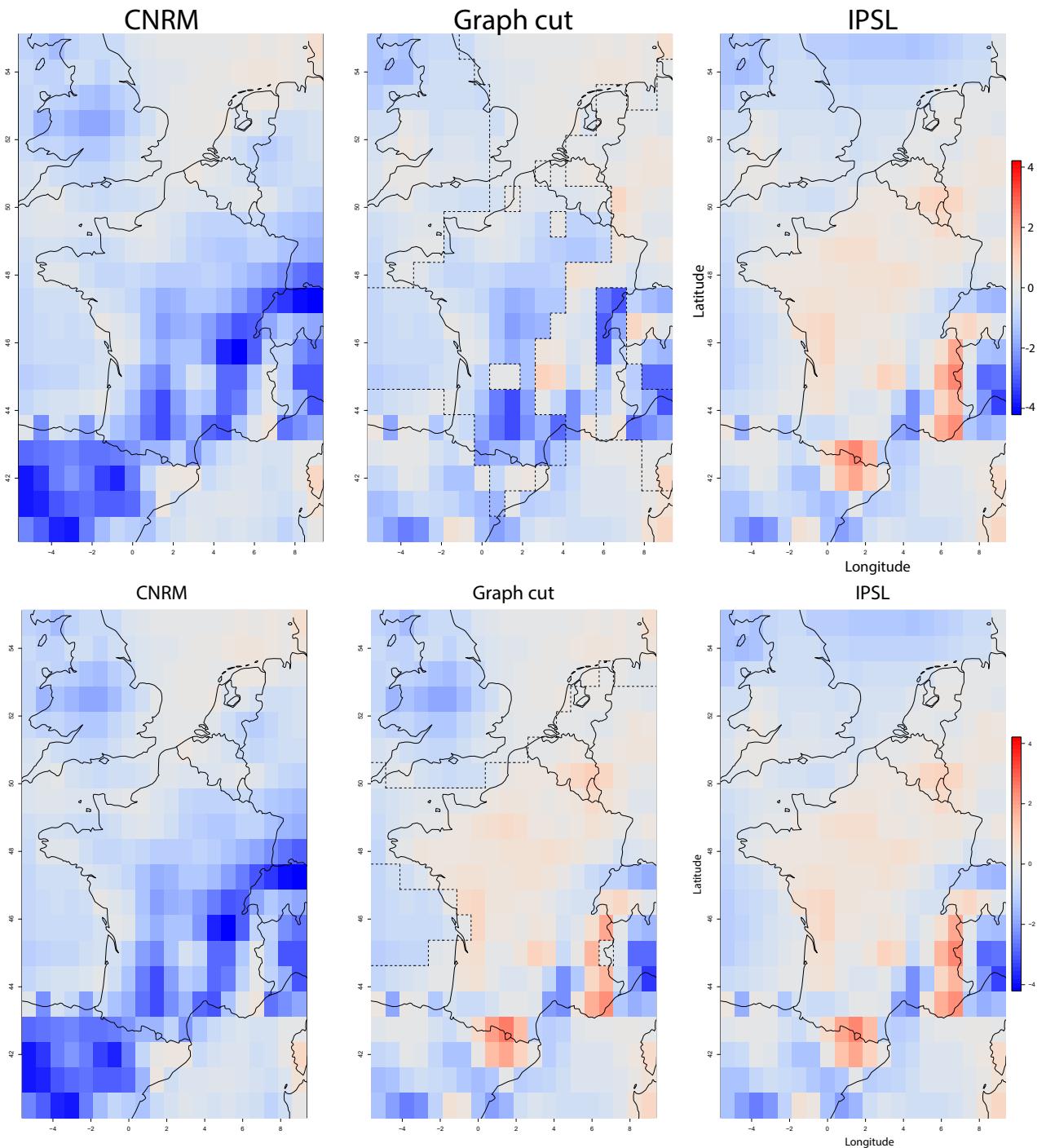


FIGURE 8 – Carte des biais de température ( $^{\circ}\text{C}$ ) CNRM (gauche), du graph cut (centre) et IPSL (droite) en attribuant 10% (haut) et 5% (bas) des biais les plus faibles pour la France

En ce qui concerne les écarts-types, cette erreur pour Q10 n'est pas présente (figure 9), montrant ainsi que le changement d'attribution de nœuds, aussi faible qu'il soit, peut totalement changer la qualité du graph cut.

Étant donné que l'erreur de Q10 est présente dans les deux approches, il n'est pas vraiment possible de définir une approche optimale pour la France (bien que *no\_st* soit très légèrement meilleure pour la moyenne). Cependant, si l'on s'intéresse à l'Europe, les approches se distinguent davantage.

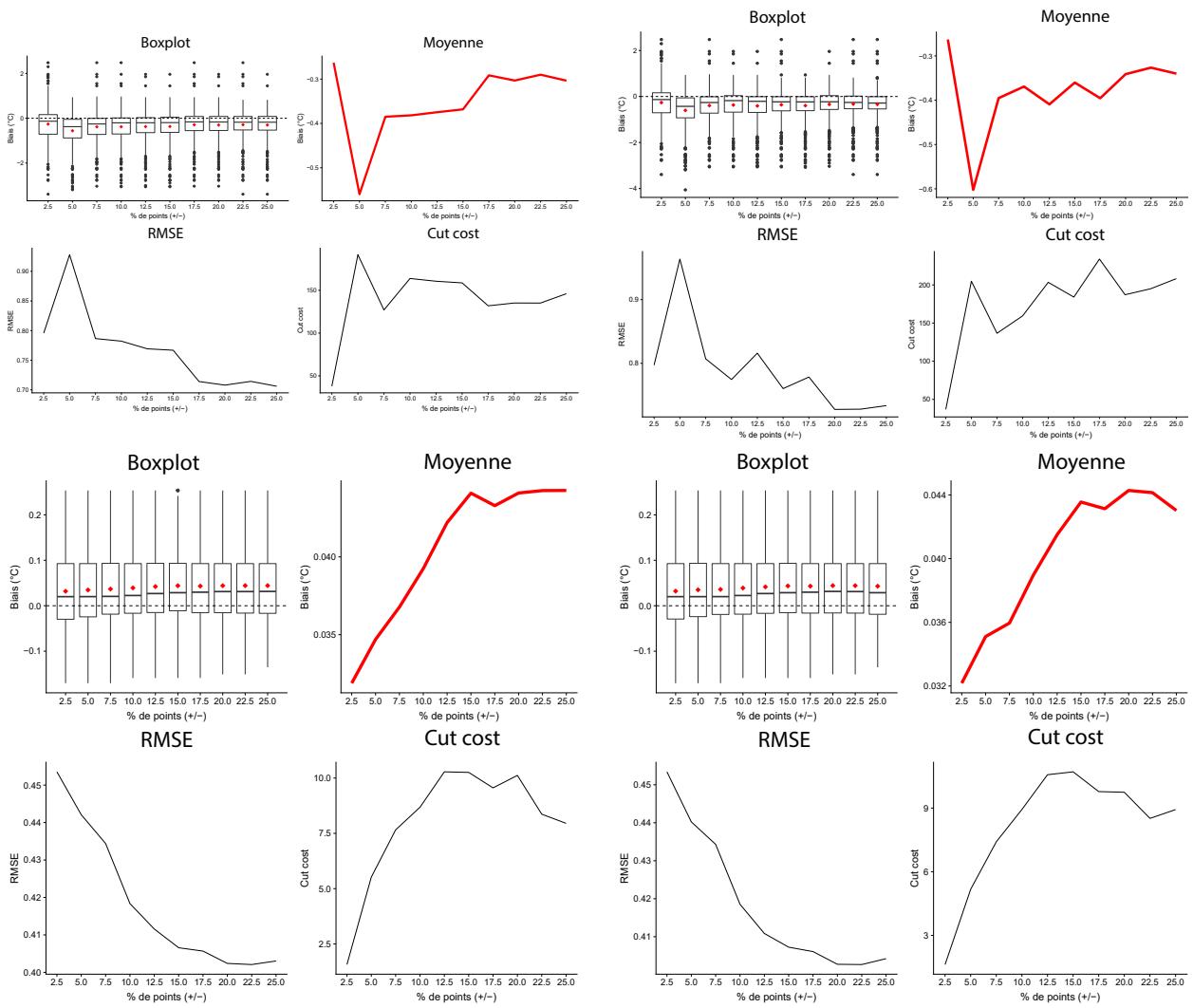


FIGURE 9 – Plots (boxplots, moyennes, RMSE, coût total du cut) de *mean\_no\_st* (en haut à gauche), *mean\_st\_smallest* (en haut à droite), *sd\_no\_st* (en bas à gauche) et *sd\_st\_smallest* (en bas à droite) en France

Si l'on compare les mêmes données (température annuelle moyenne et en écart-type) appliquées à l'Europe, on voit que les approches donnent des résultats beaucoup plus différents. Ces résultats sont mis sous forme de graphiques (figures 9 et 10) présentant les critères d'évaluation (boxplots, moyenne des biais, RMSE ["Root Mean Squared Error", RMSE en anglais] et coût du cut) du graph cut. Ces critères représentent différents aspects du cut. Les boxplots, la moyenne et la RMSE permettent d'évaluer la minimisation des biais, tandis que le coût du cut permet d'évaluer l'homogénéité du résultat. Ainsi, les 3 premiers critères ont été d'abord pris en compte, car le but est de minimiser les biais. Cependant, lors de résultats très similaires, le coût (le plus faible possible) du cut a permis de définir le meilleur résultat d'une approche.

Ainsi, si l'on compare les résultats pour la France et pour l'Europe, l'approche *no\_st* semble être la plus optimisée, notamment pour l'Europe. La raison principale est l'attribution des points aux terminaux.

L'approche *st\_smallest* attribuera de plus en plus de points aux terminaux (table 3), et forcera donc

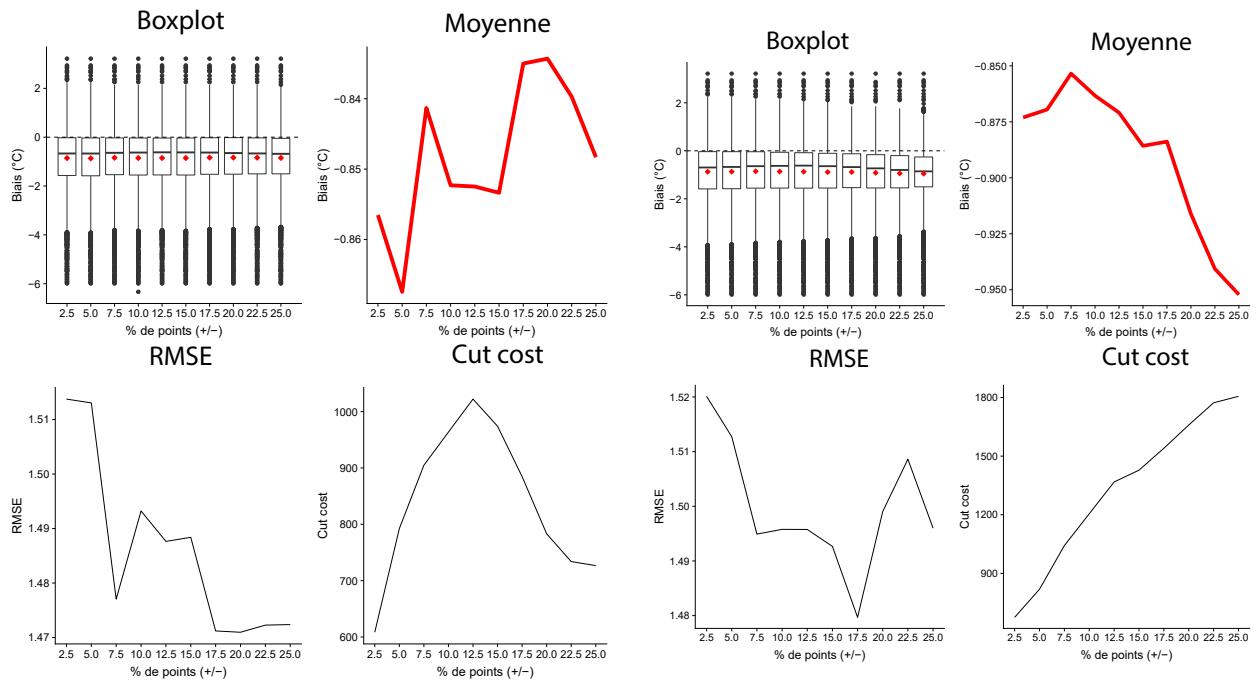


FIGURE 10 – Plots de *mean\_no\_st* (gauche) et de *mean\_st\_smallest* (droite) en Europe

le cut à passer par des points possédant des biais plus élevés que le cut de l’approche *no\_st*. En effet, si l’on choisit 50% des points ayant les biais les plus faibles, l’approche *st\_smallest* attribuera à un modèle ces 50%. Ainsi, le graph cut attribuera les 50% des points restants, possédant de ce fait des biais forts. Hors, les modèles présentent des différences en terme de biais, et donc une forte discontinuité, se traduisant par un coût élevé.

Si la différence entre les approches n’est pas marquante pour la France, elle l’est pour l’Europe, car le domaine est beaucoup plus étendu, et cela se traduit dans le coût du cut, qui croît de manière continue pour *st\_smallest*. Cela n’empêche pas l’approche *st\_smallest* d’être légèrement plus optimale dans certains cas (données annuelles mises sous forme d’écart-type par exemple).

Cependant, bien que les méthodes diffèrent, elles semblent toutes deux avoir le même comportement lorsqu’on compare le nombre de points attribués.

### 3.1.2 Nombre de points attribués

En effet, sur les plots présentés dans ce document, mais aussi dans ceux qui seront présents dans l’annexe, presque l’intégralité des tests (sauf 2 exceptions) montrent un meilleur résultat avec un grand nombre de points attribués aux terminaux (entre 35% et 50%, comme pour la figure 10). Le résultat paraît contre-intuitif, car en attribuant un plus grand nombre de points aux terminaux, l’espace disponible pour le cut est restreint, et de ce fait, le cut est obligé de passer par des biais plus importants. Pourtant, le fait d’attribuer un grand nombre de points aux terminaux force l’algorithme à choisir des biais faibles, tandis qu’en lui attribuant un nombre de points plus faible, l’algorithme va privilégier la continuité spatiale, pouvant poser un problème avec des biais forts. En effet, des arêtes peuvent avoir un faible coût car les noeuds ont tous deux des biais forts (le coût

de l'arête étant la différence des valeurs absolues des biais), et de ce fait être des coupes privilégiées. C'est pourquoi en attribuant beaucoup de biais faibles aux terminaux, le résultat sera plus proche des données de références, tout en ayant un résultat avec peu de discontinuités.

Pourtant, si l'on prend en compte les variances (et ici les écarts-types) au lieu des moyennes, les écarts entre les méthodes s'estompent fortement, voire *st\_smallest* devient meilleur, notamment à grande échelle spatiale et temporelle (saisons en France par exemple). Comme dit dans la section 2.3, *st\_smallest* a l'avantage d'attribuer un point de grille à un terminal, et donc un modèle, même lorsque ce point possède un biais faible pour les deux modèles. La limite de cette méthode serait qu'un modèle soit, pour chaque point de grille, meilleur que l'autre modèle. Mais dans ce cas de figure, la combinaison n'aurait pas d'intérêt, puisque le moins bon modèle n'apporterait aucune amélioration au meilleur modèle.

Pourtant, il existe une autre différence majeure, non pas entre les méthodes et approches, mais plutôt entre les échelles spatio-temporelles. En effet, les résultats entre les valeurs annuelles et saisonnières varient beaucoup (ainsi qu'entre la France et l'Europe), car les valeurs annuelles ont tendance à cacher le bruit des valeurs saisonnières et à masquer les différences fortes, notamment entre l'été et l'hiver (les exemples cités dans la suite de cette section seront présents en annexe pour le printemps et l'automne). De plus, c'est en s'intéressant à cette échelle inter-saisonnière que les approches atteignent leurs limites.

### 3.1.3 Échelles temporelles

En effet, les biais des modèles sont assez différents entre ces deux saisons, avec une tendance au refroidissement pour l'hiver et au réchauffement pour l'été (figure 11).

Si l'on compare les moyennes des biais entre ces deux saisons, les différences sont moindres pour la France, mais beaucoup plus importantes pour l'Europe. Cependant, en ce qui concerne le graph cut, ce n'est pas un bon critère à évaluer. En effet, bien que les moyennes d'été soient bien meilleures que les moyennes d'hiver, ces premières sont très hétérogènes. Pour l'hiver, les biais sont homogènes, mais tous négatifs, de ce fait, la moyenne l'est aussi. Cependant, en été, bien que la moyenne globale soit quasi nulle, les biais sont hétérogènes, mais étant donné que les biais positifs et négatifs ont approximativement la même valeur, la moyenne est donc nulle, ce qui explique cette forte différence de moyenne. En revanche, la RMSE est un meilleur critère d'évaluation, car les biais les plus extrêmes (-5.58°C pour le plus extrême) pour l'été sont plus faible que pour l'hiver (-10.31°C), de ce fait, la RMSE est meilleure. Toutefois, on observe une discontinuité pour l'été (présente sur tous les échantillons de points) qui va de la Pologne (5%) à l'Ukraine (50%) (figure 12).

Cette discontinuité est due principalement à la méthodologie du graph cut et la nécessité de garder une continuité spatiale. Le modèle CNRM présente des biais plus faibles dans tout l'océan Arctique, ainsi qu'en Russie. Cependant, les biais sont à la fois négatifs (océan) et positifs (conti-

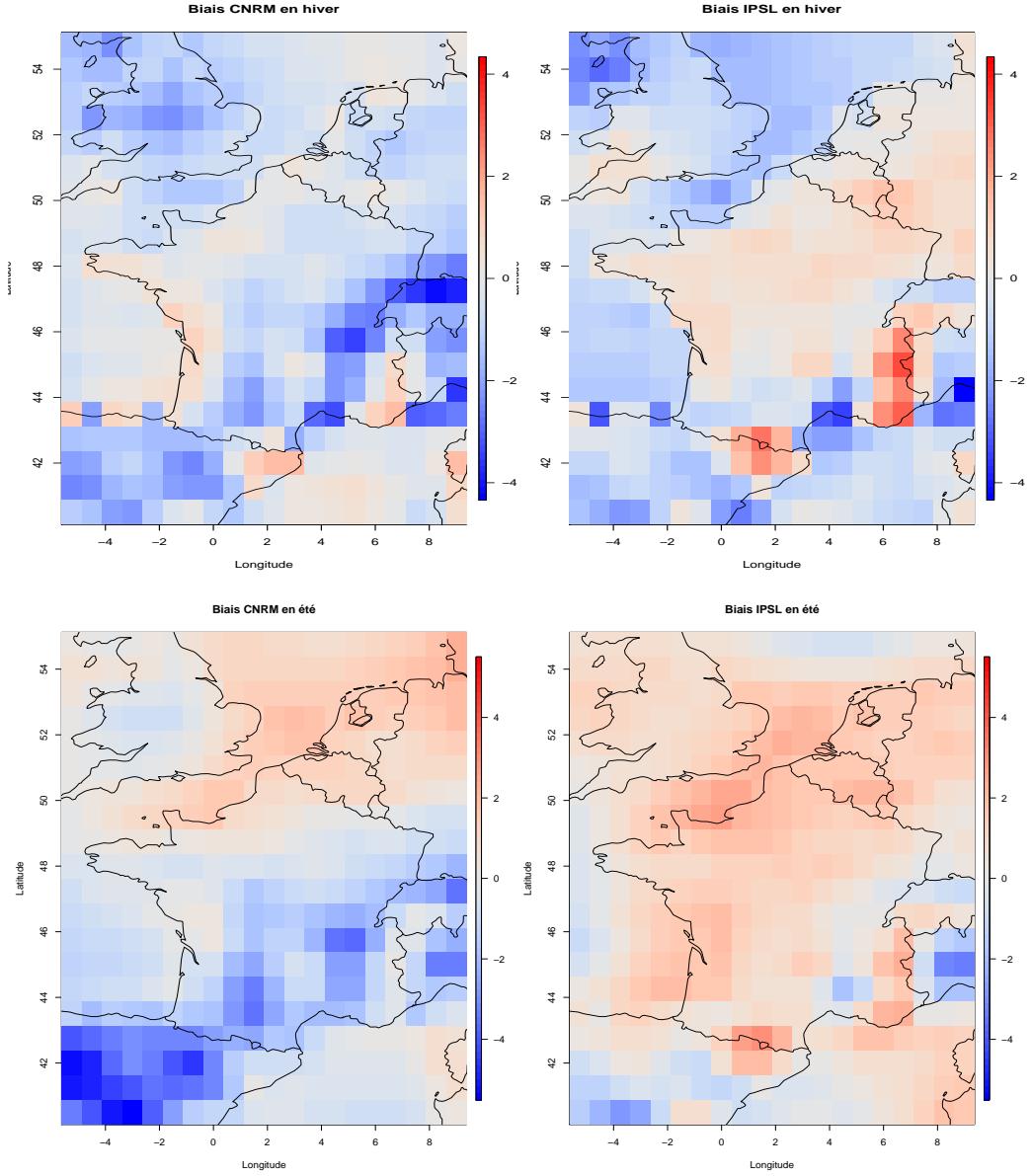


FIGURE 11 – Cartes des biais de température ( $^{\circ}\text{C}$ ) CNRM et IPSL en hiver (haut) et en été (bas) en France

ment). Le modèle IPSL lui présente un biais relativement faible, mais positif partout. Les nœuds attribués au modèle CNRM sont présents entre les latitudes  $60^{\circ}\text{N}$  et  $65^{\circ}\text{N}$  en Russie, mais aussi au niveau de la frontière avec la Géorgie. C'est pourquoi le cut privilégie le modèle CNRM, bien qu'il présente un biais légèrement plus important en se rapprochant de la discontinuité, jusqu'à pouvoir effectuer un cut, étant donné que le biais CNRM devient plus important que le biais IPSL. Cette discontinuité est la preuve que le graph cut privilégie la continuité spatiale, même en le forçant à prendre en compte les biais, et que ce choix peut altérer la qualité du résultat. De ce fait, en fonction du critère que l'on souhaite privilégier (continuité ou biais faible), l'interprétation peut varier.

Ainsi, après comparaison des approches, des échelles et des échantillons, un optimum a été défini pour chaque test, présenté dans la table 4.

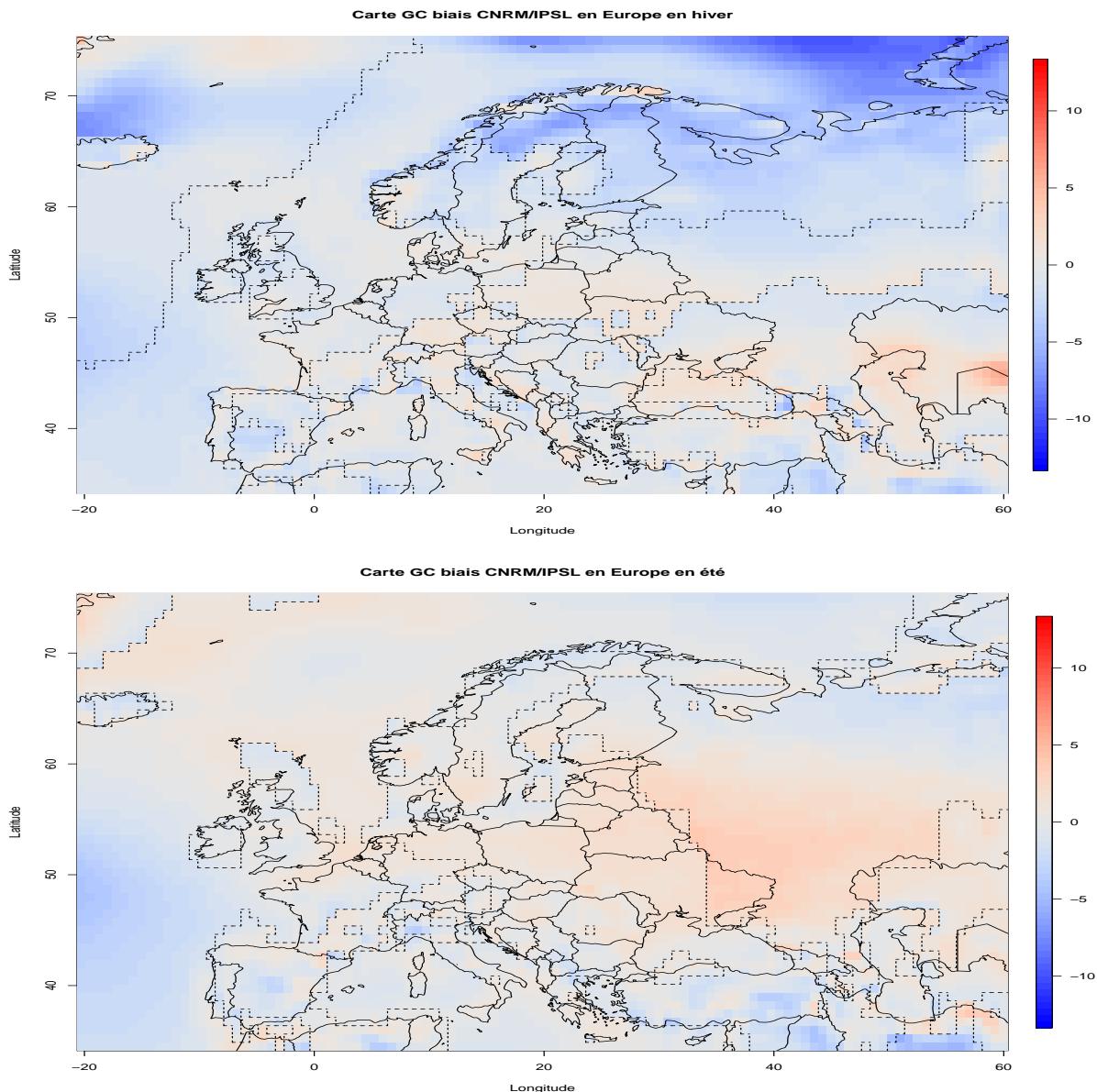


FIGURE 12 – Cartes des biais de température ( $^{\circ}\text{C}$ ) du graph cut en hiver (haut) et en été (bas) en Europe

TABLE 4 – Récapitulatif des optimum de chaque test sur les températures

Méthode	Optimal
tas_mean_annual_eur	40% des points attribués pour méthode <i>no_st</i>
tas_mean_annual_fr	35% des points attribués pour méthode <i>no_st</i>
tas_mean_season_eur	40% des points attribués pour méthode <i>no_st</i> en été
tas_mean_season_fr	25% des points attribués pour les deux méthodes en été
tas_sd_annual_eur	45% des points attribués pour les deux méthodes
tas_sd_annual_fr	5 % des points attribués pour la méthode <i>st_smallest</i>
tas_sd_season_eur	50% des points attribués pour la méthode <i>st_smallest</i> en été
tas_sd_season_fr	40% des points attribués pour la méthode <i>no_st</i> en hiver

### 3.1.4 Comparaison et projection

Ainsi, si l'on compare ces optima aux mêmes tests, mais cette fois-ci avec une approche ensembliste par moyenne, le résultat du graph cut est dans l'ensemble plus proche des données de

référence (figure 13 et 14).

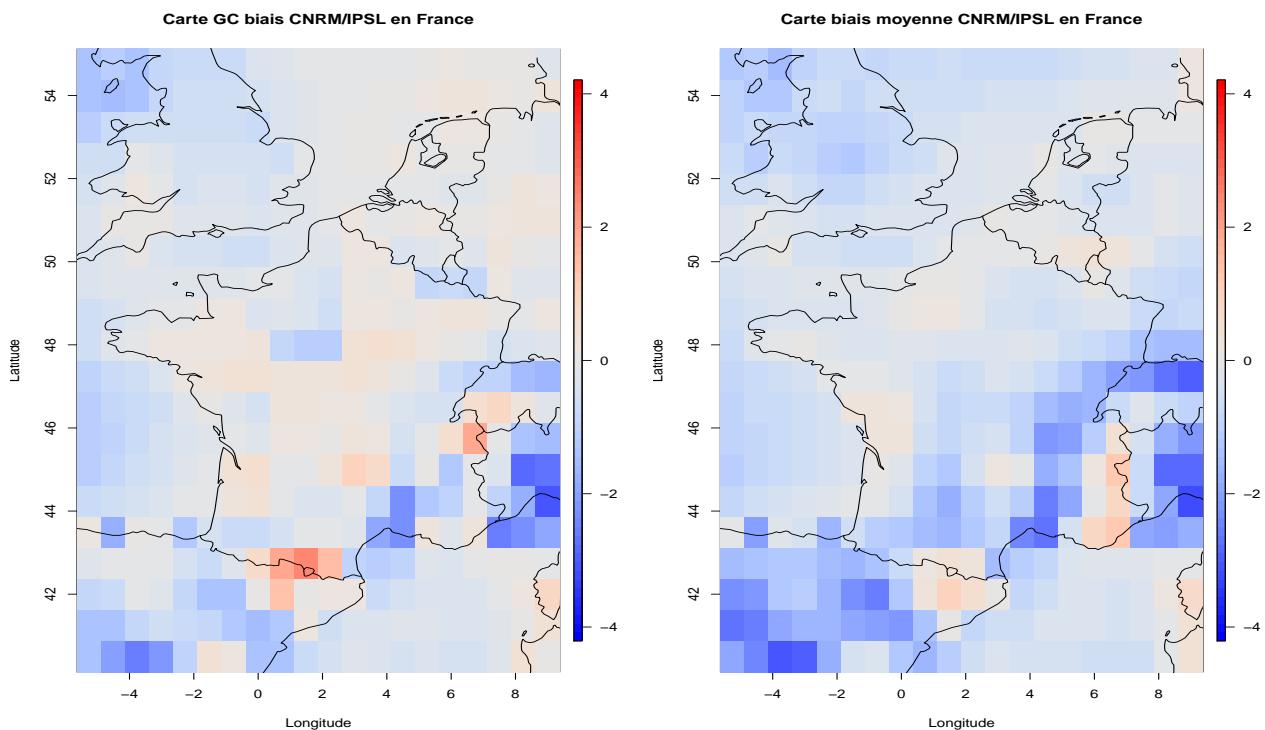


FIGURE 13 – Cartes des biais de température ( $^{\circ}\text{C}$ ) du graph cut (gauche) et de moyennes (droite) en France

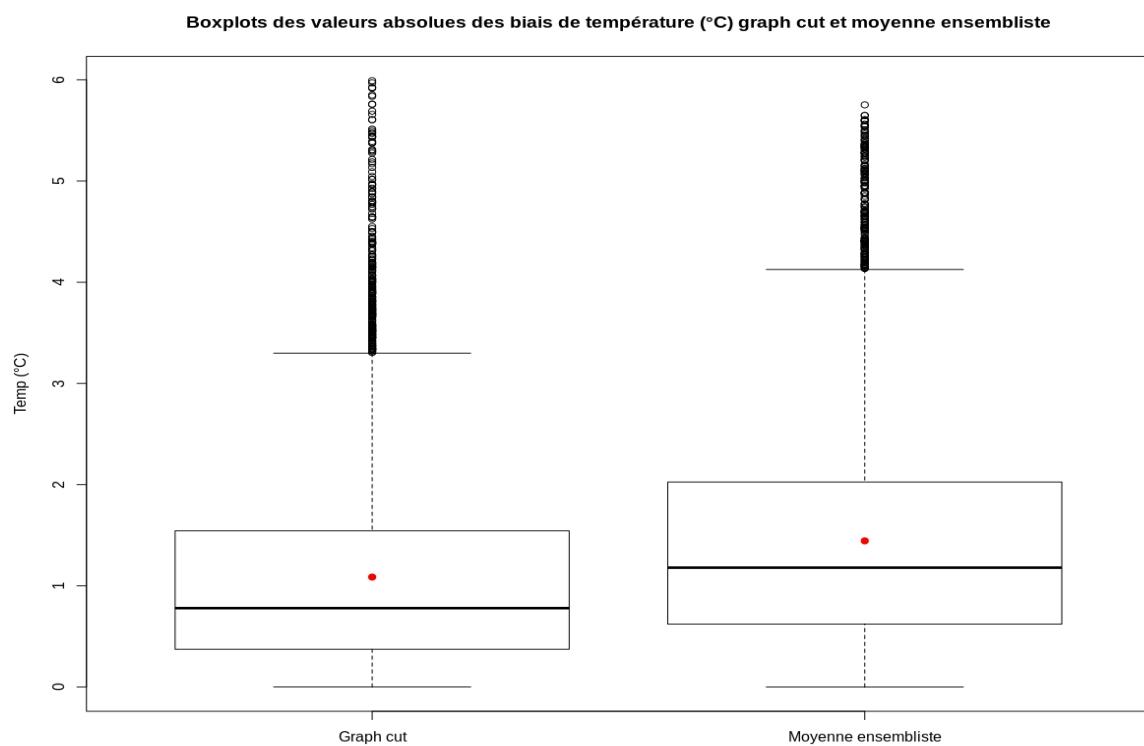


FIGURE 14 – Boxplot des valeurs absolues des biais de température ( $^{\circ}\text{C}$ ) du graph cut (gauche) et des biais d'une moyenne arithmétique des modèles CNRM et IPSL (droite) en Europe

De plus, si l'on applique le résultat du graph cut sur les séries temporelles des projections (2041-2070 et 2071-2100), le résultat semble correct et l'homogénéité de la carte est toujours présente (figure 15). En effet, le but de cette application était de voir si les modèles faisaient des projections cohérentes, bien qu'il n'est pas possible de savoir si les biais des modèles diminueront ou augmenteront entre les projections.

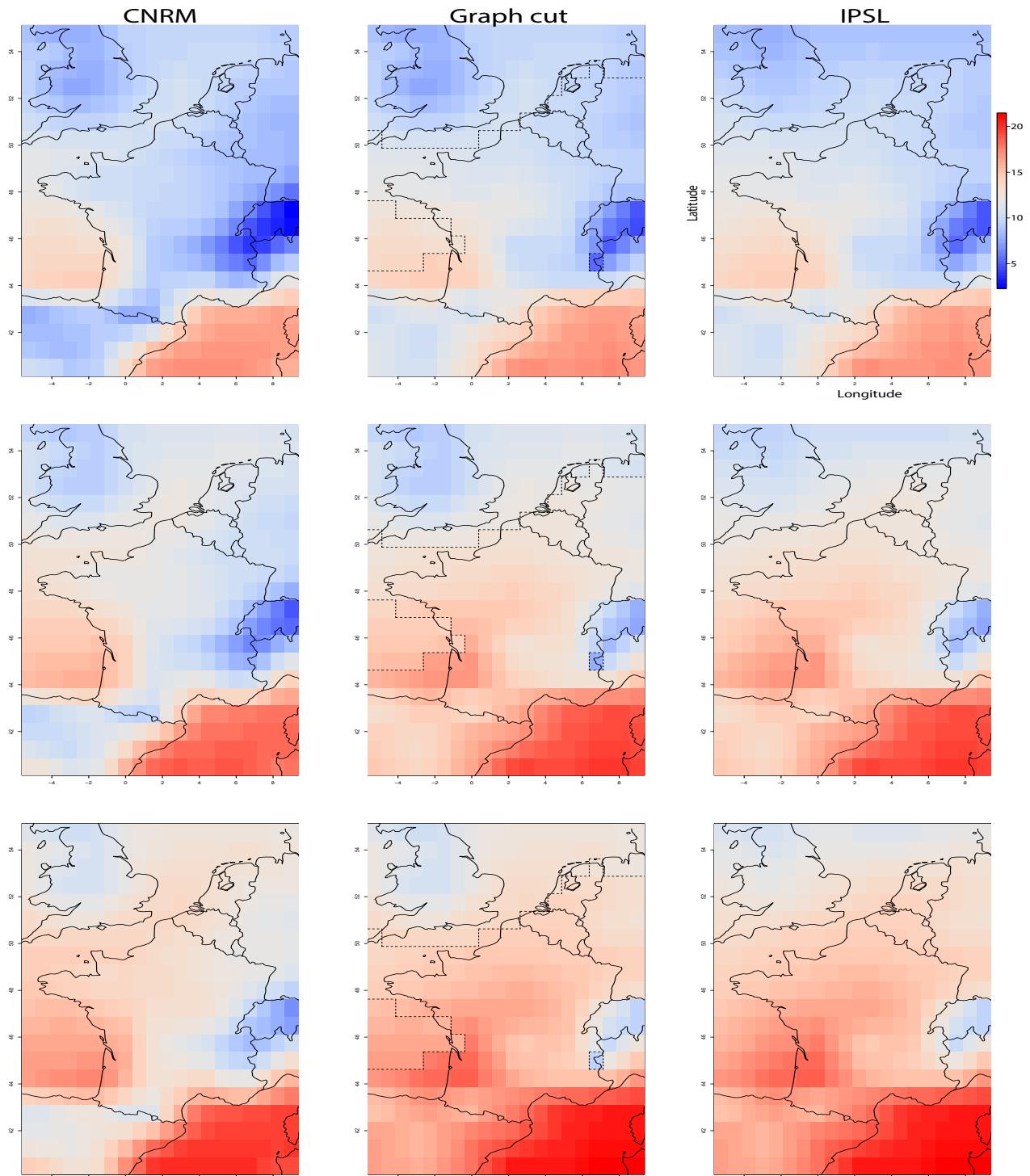


FIGURE 15 – Cartes des projections de température ( $^{\circ}\text{C}$ ) du modèle CNRM (gauche), du graph cut (centre) et du modèle IPSL (droite) en France pour les périodes 1979-2008 (haut), 2041-2070 (centre) et 2071-2100 (bas)

## 3.2 Précipitations

Après avoir appliqué la méthodologie du graph cut et fait les tests sur la température, l'idée est de voir comment se comporte cette méthodologie pour d'autres variables climatiques, comme les précipitations. Globalement, les résultats sont similaires, cependant, les précipitations confirment différentes hypothèses présentées dans la section précédente : la limite de l'approche *no\_st* et l'importance de comparer les variables mises sous différentes formes (moyennes ou en écarts-types) et différentes temporalités (annuelle ou saisonnière).

### 3.2.1 Attribution des nœuds

Bien que les modèles représentent la précipitation de manière assez différentes, ils possèdent un très grand nombre de biais faibles communs, pour la France, mais surtout pour l'Europe. En effet, si l'on s'intéresse aux moyennes annuelles de précipitations, ce n'est qu'avec l'échantillon Q50 que l'on obtient une attribution, mais qui reste très faible (2 points pour la *source* et 8 pour la *target* correspondant respectivement à 0.03% et à 0.13%). Cela prouve que, bien que l'approche *no\_st* présente de meilleurs résultats, elle n'est pas applicable dans toutes les situations, tandis que l'approche *st\_smallest* l'est, puisqu'elle attribuera forcément un terminal à un biais faible. Cependant, si l'on s'intéresse aux moyennes saisonnières, l'approche *no\_st* attribue des points de grille aux terminaux, même avec l'échantillon le plus faible, à savoir Q05 (sauf pour l'automne, où l'attribution ne se fait qu'à partir de Q35). Cela prouve que, bien que les biais faibles soient identique à l'échelle annuelle, ils diffèrent beaucoup à l'échelle saisonnière, et donc que les modèles climatiques simulent les variables avec plus ou moins de différence.

### 3.2.2 Nombre de points attribués

En ce qui concerne la taille des échantillons, le graph cut se comporte de la même manière pour la précipitation que pour la température. En effet, la majorité des résultats présentent des optima avec de grands échantillons (entre Q40 et Q50). Cependant, lorsqu'on s'intéresse aux moyennes en hiver et en été pour la France, le résultat est optimal pour de petits échantillons (Q10 pour l'hiver et Q15 pour l'été). Ce résultat est dû à mauvaise représentation du modèle IPSL pour ce test. En forçant l'algorithme à considérer un plus grand nombre de points à attribuer, le résultat se retrouve dégradé (figure 16).

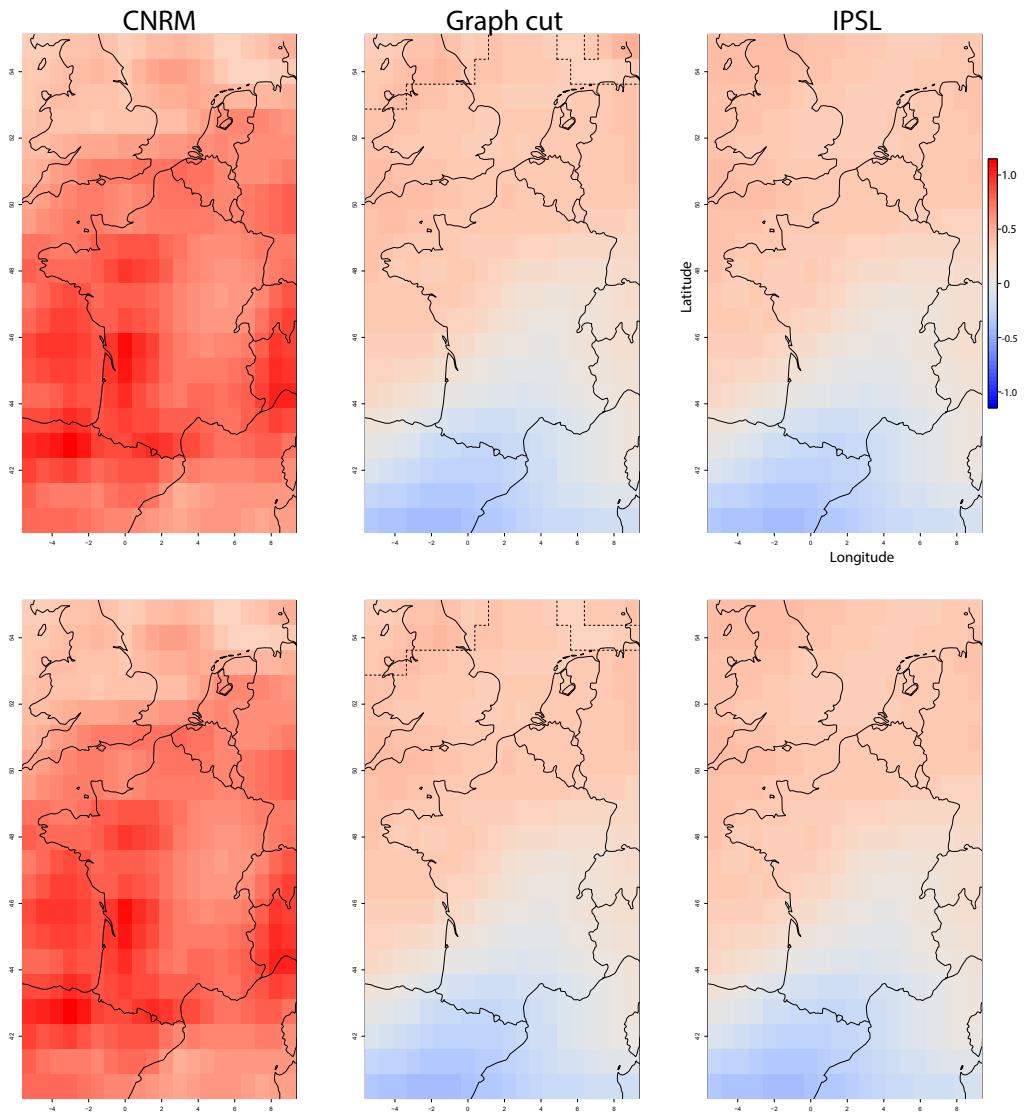


FIGURE 16 – Cartes des biais de précipitations (mm/j) du modèle CNRM (gauche), du graph cut (centre) et du modèle IPSL (droite) en attribuant 30% (haut) et 50% (bas) pour la France

### 3.2.3 Échelles temporelles

Si l'on s'intéresse maintenant aux moyennes saisonnières, les biais faibles semblent différer entre les modèles, permettant ainsi de comparer les résultats entre les deux approches. De plus, les biais des simulations climatiques changent de comportement entre la température et les précipitations, notamment en été et en hiver. Les deux modèles avaient tendance à présenter des biais de température positifs (donc sur-estimer) en été, et des biais négatifs (et donc sous-estimer) en hiver, tandis que pour les précipitations, les biais sont inversés (sous-estimés en été et sur-estimés en hiver). Ainsi, les modèles ont tendance sur-estimer le changement climatique en été (températures plus élevées et moins de précipitations) et le sous-estimer en hiver (températures moins élevées et plus de précipitations) (figure 17).

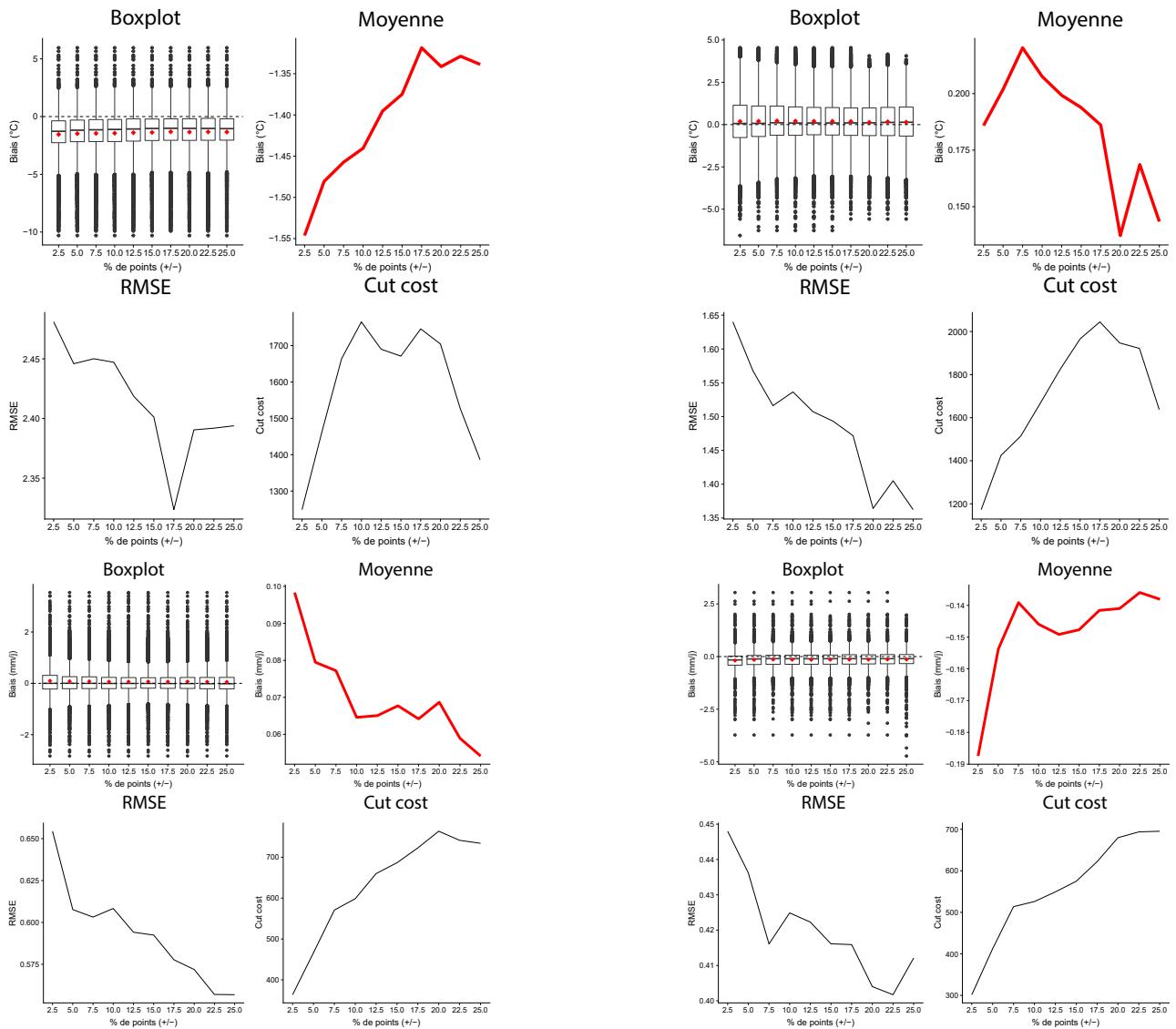


FIGURE 17 – Plots (boxplots, moyennes, RMSE, coût total du cut) des températures (haut) et des précipitations (bas) en hiver (gauche) et en été (droite) en Europe

### 3.2.4 Comparaison et projection

De la même manière que la température, l’application du graph cut donne dans l’ensemble de meilleurs résultats qu’une moyenne de biais. Cependant, avec les données fournies, la différence est plus flagrante. En effet, le modèle IPSL étant bien meilleur que le modèle CNRM en terme de précipitations dans le domaine France, l’approche par moyenne donne un résultat moins bon (figure 18). Ainsi, le graph cut n’est que très légèrement amélioré par le modèle CNRM (moyenne améliorée de 0.003°C). Cela prouve que le graph cut ne peut exploiter que le meilleur des modèles, mais pas les améliorer.

**Boxplot des valeurs absolues des biais de précipitations (mm/j) graph cut, moyenne ensembliste, modèle IPSL et CNRM**

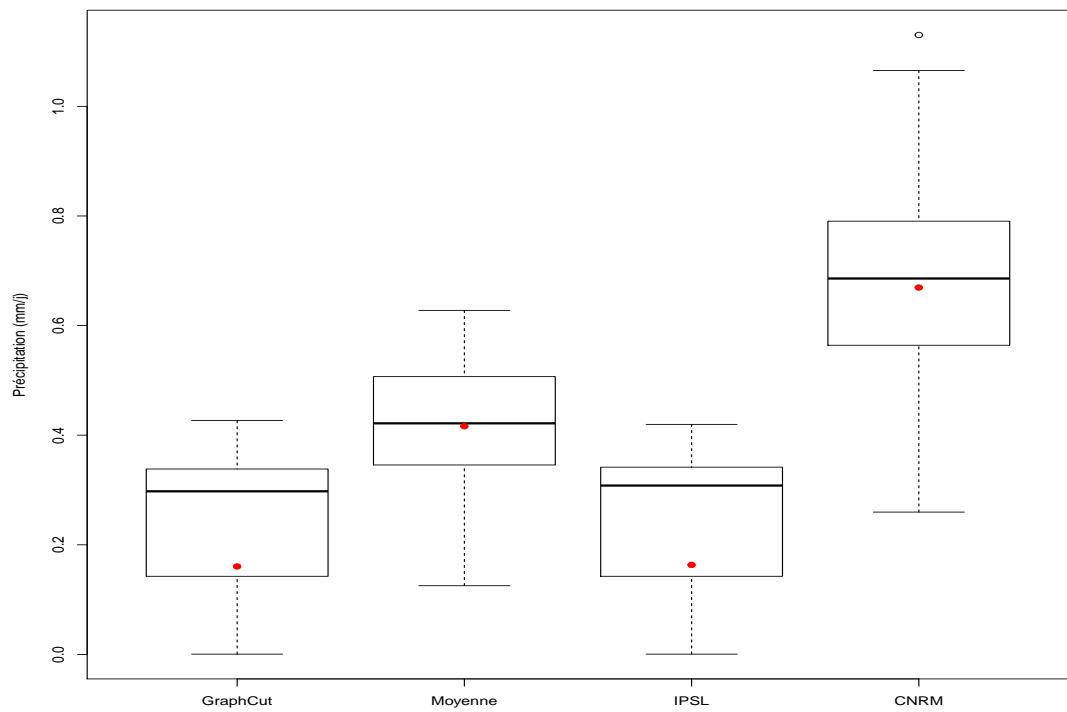


FIGURE 18 – Boxplot des valeurs absolues des biais de précipitation (mm/j) en France avec de gauche à droite : graph cut, moyenne arithmétique des modèles CNRM et IPSL, modèle IPSL, modèle CNRM

Les projections n'ont pas pu être générées par manque de temps.

## 4 Conclusions et perspectives

Les différents tests de ce stage ont permis de prouver que l'application du graph cut sur des simulations climatiques a permis d'obtenir de meilleurs résultats qu'avec une approche ensembliste par moyenne. En effet, en terme de biais, le modèle résultant de l'algorithme du graph cut a donné de meilleurs résultats. Les approches définies durant la mise en place de la méthodologie ont montré leurs forces, mais aussi leurs faiblesses. Les différents tests ont montré que l'augmentation du nombre de points de grille attribués permet au graph cut de prendre en compte la minimisation des biais, mais que l'algorithme privilégie la continuité spatiale, puisqu'il est paramétré pour répondre au théorème du *flow* de Ford et Fulkerson. Enfin, les différentes échelles temporelles ont prouvé qu'un résultat peut fortement différer, et cacher des biais forts, mais aussi des biais très faibles.

Cependant, les tests, bien que nombreux, restent assez similaire, puisque les données ne prennent en compte qu'une seule variable climatique. C'est cette problématique qui ouvre une des perspectives principales du stage.

En effet, durant le temps restant du stage, le principal objectif serait de voir comment l'algorithme du graph cut se comporterait avec des données comportant plusieurs variables climatiques. Un autre objectif, concernant l'implémentation du code, consisterait à rendre le code itératif, le but étant de couvrir une plus grande zone (idéalement le globe) tout en gardant une zone d'étude petite. Un des freins de la méthodologie créée était le temps de calcul, augmentant très fortement en augmentant la taille (quelques secondes de calcul pour la France, entre 4 et 20 heures pour l'Europe en fonction du test, et plus de 15 jours pour le globe). Une autre perspective, plus complexe, serait de passer de 2 à  $n$  modèles, le but étant de reformuler l'algorithme afin de pouvoir obtenir un graphe avec  $n$  terminaux. Cependant, le stage se terminera sûrement avant d'atteindre un résultat.

### 4.1 Apport personnel

Ce stage a eu un très fort impact sur mes attentes et mes envies professionnelles. Avant d'avoir été contacté par Soulivanh, je ne m'étais jamais vraiment intéressé aux statistiques appliquées à la climatologie et à ces questions de correction de biais et de combinaison de simulations. J'ai donc été très intrigué et perplexe lorsqu'il m'a contacté pour ce stage, et ne savais pas trop à quoi m'attendre et pourquoi Mathieu, Soulivanh et Grégoire avaient décider de me choisir.

Cependant, je pense que ce stage a vraiment changé la vision que je me fais de mon parcours professionnel. Avant de venir ici, je souhaitais effectuer une thèse dans le but d'essayer de devenir enseignant-chercheur, car je ne pensais pas que la recherche était quelque chose que je souhaitais faire à plein temps. Cependant, avoir passé ces deux mois au sein de l'équipe ESTIMR, le cadre étant vraiment très chaleureux, et le sujet de mon stage passionnant, je me suis rendu compte que la recherche était finalement quelque chose de vraiment passionnant, et même d'amusant. En effet, ce fut un réel plaisir de concevoir ce code, en me basant sur des recherches et

des conseils de Mathieu de Soulivanh, et j'ai hâte de pouvoir continuer de l'améliorer, et de pousser ce sujet plus loin.

Tous ces éléments font que mon projet professionnel a changé, et cela m'enchante. J'espère fortement avoir la chance de revenir dans cette équipe pour effectuer d'autres travaux.

## 5 Bibliographie

- Auger, J.D.; Birkel, S.D.; Maasch, K.A.; Mayewski, P.A.; Schuenemann, K.C. An Ensemble Mean and Evaluation of Third Generation Global Climate Reanalysis Models. *Atmosphere* 2018, 9, 236.
- Berrisford, P., Dee, D., Poli, P., Brugge, R., Fielding, K., Fuentes, M., Kallberg, P., Kobayashi, S., Uppala, S. and Simmons, A., (2011) The ERA-Interim archive, version 2.0. ERA report series. 1. Technical Report. ECMWF pp23.
- Boykov, Y., O. Veksler and R. Zabih (2001). "Fast approximate energy minimization via graph cuts." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23(11) : 1222-1239.
- Boykov, Y. and G. Funka-Lea (2006). "Graph cuts and efficient N-D image segmentation." *International Journal of Computer Vision* 70(2) : 109-131.
- Fontenla M. (2014), opTrees, R package.
- Ford, L.R., and D. R. Fulkerson (1956), Maximal Flow through a network, *Can. J. Math.*, 8(3), 399-404
- Knutti, R., R. Furrer, C. Tebaldi, J. Cermak and G. A. Meehl (2010). "Challenges in combining projections from multiple climate models." *Journal of Climate* 23(10) : 2739-2752; 22(3) : 277-286.
- Kwatra, N., A. Schödl, I. Essa, G. Turk and A. Bobick (2003). "Graphcut textures : Image and video synthesis using graph cuts." *ACM transactions on Graphics* 22(3) : 277-286.
- Li, X., G. Mariethoz, D. Lu and N. Linde (2016). "Patch-based iterative conditional geostatistical simulation using graph cuts." *Water Resources Research*.
- Sanderson, B. M., and Coauthors, 2008 : Constraints on model response to greenhouse gas forcing and the role of subgrid-scale processes. *J. Climate*, 21, 2384–2400.
- Solomon, S., G. K. Plattner, R. Knutti, and P. Friedlingstein, 2009 : Irreversible climate change due to carbon dioxide emissions. *Proc. Natl. Acad. Sci. USA*, 106, 1704–1709.
- Vrac, M. (2018) Multivariate bias adjustment of high-dimensional climate simulations : the Rank Resampling for Distributions and Dependences (R2D2) Bias Correction. *Hydrol. Earth Syst. Sci.*, 22, 3175-3196, <https://doi.org/10.5194/hess-22-3175-2018>

# Annexes

## A Code

Présentation de l'implémentation de la méthodologie sur R (test : moyenne annuelle en Europe en attribuant 5% des points avec la méthode *no\_st*) :

```
list.of.packages <- c("optrees", "ncdf4", "fields", "maps", "mapdata", "Metrics")
new.packages <- list.of.packages[!(list.of.packages %in% installed.packages())[,]
if(length(new.packages)) install.packages(new.packages, repos = "http://cran.us.r
lapply(list.of.packages, library, character.only = TRUE)

# Variables : 1.10
# Settings : 1.43 + 1.401
# Attribution : 1.215

### Variables ###

nc <- nc_open("t2_erai_clim_1979_2008.nc")      # Ouvrir fichier netcdf et permet
tas_erai = ncvar_get(nc, varid = "t2")           # Importer la variable contenant l
lat_erai = ncvar_get(nc, varid = "lat")          # V rifier si dans ordre croissan
lon_erai = ncvar_get(nc, varid = "lon")
tas_erai = tas_erai[, length(lat_erai) : 1]      # Derni re colonne devient la pr
lat_erai = rev(lat_erai)                         # Inverser ordre de la matrice
nc_close(nc)

nc2 <- nc_open("tas_cnrm_clim_1979_2008.nc")
tas_cnrm = ncvar_get(nc2, varid = "tas")
lat_cnrm = ncvar_get(nc2, varid = "lat")
lon_cnrm = ncvar_get(nc2, varid = "lon")
tas_cnrm = tas_cnrm[, length(lat_cnrm) : 1]
lat_cnrm = rev(lat_cnrm)
nc_close(nc2)

nc3 <- nc_open("tas_ipsl_clim_1979_2008.nc")
tas_ipsl = ncvar_get(nc3, varid = "tas")
lat_ipsl = ncvar_get(nc3, varid = "lat")
lon_ipsl = ncvar_get(nc3, varid = "lon")
tas_ipsl = tas_ipsl[, length(lat_ipsl) : 1]
lat_ipsl = rev(lat_ipsl)
nc_close(nc3)

cnrm_bias <- tas_cnrm-tas_erai
```

```

ipsl_biais <- tas_ipsl-tas_erai
colorTable<- designer.colors(64, c( "blue","grey90", "red"))

#### Settings biais ####

europe = map(xlim=c(-20.25,60),
              ylim=c(34.5,75),
              plot=FALSE)

lon_fr_cnrm = c((lon_cnrm[454:480]-360),lon_cnrm[1:81]) # lon comprises entre 1
lat_fr_cnrm = lat_cnrm[167:221] # lat comprises entre 34,5 N
tas_fr_cnrm <- rbind(cnrm_biais[454:480,167:221],cnrm_biais[1:81,167:221])

lon_fr_ipsl = c((lon_ipsl[454:480]-360),lon_ipsl[1:81])
lat_fr_ipsl = lat_ipsl[167:221]
tas_fr_ipsl <- rbind(ipsl_biais[454:480,167:221],ipsl_biais[1:81,167:221])

lon_fr_erai = c((lon_erai[454:480]-360),lon_erai[1:81])
lat_fr_erai = lat_erai[167:221]
tas_fr_erai = rbind(tas_erai[454:480,167:221],tas_erai[1:81,167:221])-273.15

fr_biais = c(tas_fr_cnrm,tas_fr_ipsl)
fr_temp = c(tas_fr_cnrm,tas_fr_ipsl,tas_fr_erai)

# Crit re sur le biais vs r f rence
cnrm_fr_10 = quantile(abs(fr_biais),0.05,na.rm = TRUE)
crit1C_fr <- abs(tas_fr_cnrm) <=cnrm_fr_10
# Crit re sur la diff rence des biais r f rence
seuil_50C_fr <- quantile(abs(tas_fr_ipsl) - abs(tas_fr_cnrm), 0.5, na.rm = TRUE)
crit2C_fr <- abs((abs(tas_fr_ipsl) - abs(tas_fr_cnrm))) > seuil_50C_fr
mat_cnrm_10_fr <- tas_fr_cnrm
mat_cnrm_10_fr[!(crit1C_fr)] <- NA

# Critere sur le biais vs reference
ipsl_10_fr = quantile(abs(fr_biais),0.05,na.rm = TRUE)
crit1_fr <- abs(tas_fr_ipsl) <= ipsl_10_fr

split_col <- function(mat,y,n.col){
  mat2 <- matrix(mat[,y:(y+n.col)-1]],nrow=nrow(mat),ncol=n.col)
  return(mat2)
}

```

```

}

overlap_cnrm = tas_fr_cnrm
overlap_ipsl = tas_fr_ipsl
overlap = overlap_cnrm-overlap_ipsl

poids <- overlap
indices <- matrix(1:length(poids), ncol=ncol(poids))

mat.pad <- function(mat1){
  na1 <- matrix(NA, ncol=1, nrow=nrow(mat1))
  mat2 <- matrix(c(na1, mat1, na1), nrow=nrow(mat1), ncol=ncol(mat1)+2)
  na2 <- matrix(NA, nrow=1, ncol=ncol(mat2))
  mat3 <- rbind(na2, mat2, na2)
  return(mat3)
}

location <- function(matrice){
  matrice2 <- mat.pad(matrice)
  x <- 1:length(matrice)
  ind_row <- 2:(nrow(matrice)+1)
  ind_col <- 2:(ncol(matrice)+1)
  neigh = rbind(N = as.vector(matrice2[ind_row - 1, ind_col]),
                E = as.vector(matrice2[ind_row , ind_col + 1]),
                S = as.vector(matrice2[ind_row + 1, ind_col ]),
                W = as.vector(matrice2[ind_row , ind_col - 1]))
  return(neigh[,x])
}

indice <- function(matrice,x){
  n.row2 <- which(matrice==x, arr.ind=TRUE)
  n.col2 <- which(matrice==x, arr.ind=TRUE)
  n.row <- n.row2[,1]
  n.col <- n.col2[,2]
  N <- which(matrice==x-1,arr.ind=TRUE)
  if (n.row==1) N=c(NA,NA)
  E <- which(matrice==x+nrow(matrice),arr.ind=TRUE)
  if (n.col==ncol(matrice)) E=c(NA,NA)
  S <- which(matrice==x+1,arr.ind=TRUE)
  if (n.row==nrow(matrice)) S=c(NA,NA)
  W <- which(matrice==x-nrow(matrice),arr.ind=TRUE)
  if (n.col==1) W=c(NA,NA)
  location_neigh_x <- matrix(c(N,E,S,W),nrow=4, byrow = TRUE)
}

```

```

row.names(location_neigh_x) <- c("N", "E", "S", "W")
colnames(location_neigh_x) <- c("x", "y")
return(location_neigh_x)
}

location <- location(indices)
colnames(location) <- 1:ncol(location)

test_matgc <- function(matrice){
  n <- rep(1:length(matrice), each=4)
  x <- rep(1:4, times=length(matrice))
  test_matgc <- matrix(c(n, location[x, n]))
  return(test_matgc)
}

location_poids <- function(matrice){
  matrice2 <- mat.pad(matrice)
  x <- 1:length(matrice)
  ind_row <- 2:(nrow(matrice)+1)
  ind_col <- 2:(ncol(matrice)+1)
  N = abs(as.vector(matrice2[ind_row - 1, ind_col])) + as.vector(matrice2[ind_row, ind_col])
  E = abs(as.vector(matrice2[ind_row, ind_col + 1])) + as.vector(matrice2[ind_row, ind_col - 1])
  S = abs(as.vector(matrice2[ind_row + 1, ind_col])) + as.vector(matrice2[ind_row, ind_col])
  W = abs(as.vector(matrice2[ind_row, ind_col - 1])) + as.vector(matrice2[ind_row, ind_col + 1])
  if (length(N) == 0) N=c(NA,NA)
  if (length(E) == 0) E=c(NA,NA)
  if (length(S) == 0) S=c(NA,NA)
  if (length(W) == 0) W=c(NA,NA)
  neigh = rbind(N,E,S,W)
  return(neigh[,x])
}

poids2 <- location_poids(poids)

n <- rep(1:ncol(location), each=4)
n2 <- rep(1:ncol(poids2), each=4)
x <- seq(1:(length(location)))
x2 <- seq(1:(length(poids2)))

test_matgc <- matrix(c(n, location[x]), ncol=2)
test_poidsgc <- matrix(c(n2, poids2[x]), ncol=2)
mat_gc_test <- cbind(test_matgc, test_poidsgc[, 2])

```

```

mat_gc <- mat_gc_test[-(which(is.na(mat_gc_test[,2])),]
colnames(mat_gc) <- c("Noeud_1", "Noeud_2", "Poids")

matriceGC2 <- subset(mat_gc, mat_gc[,1] < mat_gc[,2])

nodes <- 1:(length(poids)+2)

location <- function(matrice){
  matrice2 <- mat.pad(matrice)
  x <- 1:length(matrice)
  ind_row <- 2:(nrow(matrice)+1)
  ind_col <- 2:(ncol(matrice)+1)
  neigh = rbind(N = as.vector(matrice2[ind_row - 1, ind_col]),
                 E = as.vector(matrice2[ind_row , ind_col + 1]),
                 S = as.vector(matrice2[ind_row + 1, ind_col]),
                 W = as.vector(matrice2[ind_row , ind_col - 1]))
  return(neigh[,x])
}

source_test_3=matrix(1:length(overlap),ncol=1)
mat_ipsl_10_test <- matrix(mat_ipsl_10_fr,ncol=1)
for(i in 1:length(overlap)){
  if(!is.na(mat_ipsl_10_test[i])){
    mat_ipsl_10_test[i]=source_test_3[i]
  }
}
source_nodes <- mat_ipsl_10_test[-(which(is.na(mat_ipsl_10_test)),arr.ind=TRUE)],

sink_test_3=matrix(1:(length(overlap)),ncol=1)
mat_cnrm_10_test <- matrix(mat_cnrm_10_fr,ncol=1)
for(i in 1:(length(overlap))){
  if(!is.na(mat_cnrm_10_test[i])){
    mat_cnrm_10_test[i]=sink_test_3[i]
  }
}
sink_nodes <- mat_cnrm_10_test[-(which(is.na(mat_cnrm_10_test)),arr.ind=TRUE)],

diff = matrix((mat_ipsl_10_fr)-(mat_cnrm_10_fr),ncol=1)

### Attribution s_nodes_Q05_mean_eur et t_nodes_Q05_mean_eur ###

for(i in 1:length(overlap)){

```

```

if((i %in% source_nodes) & (i %in% sink_nodes)){
  source_nodes[source_nodes==i] <- NA
  sink_nodes[sink_nodes==i] <- NA
}
}

source_nodes <- source_nodes[-which(is.na(source_nodes))]
sink_nodes <- sink_nodes[-which(is.na(sink_nodes))]
source <- cbind(c(rep((length(nodes)-1),each=(length(source_nodes)))), 
                 c(source_nodes),
                 10000)
colnames(source) <- c("Noeud_1", "Noeud_2","Poids")
sink <- cbind(c(rep((length(nodes)),each=(length(sink_nodes)))), 
               c(sink_nodes),
               10000)
colnames(sink) <- c("Noeud_1", "Noeud_2","Poids")

matriceGC <- rbind(source,matriceGC2,sink)

MinCut <- findMinCut(nodes, matriceGC, source.node=(length(nodes)-1), sink.node=

```

## B Cartes

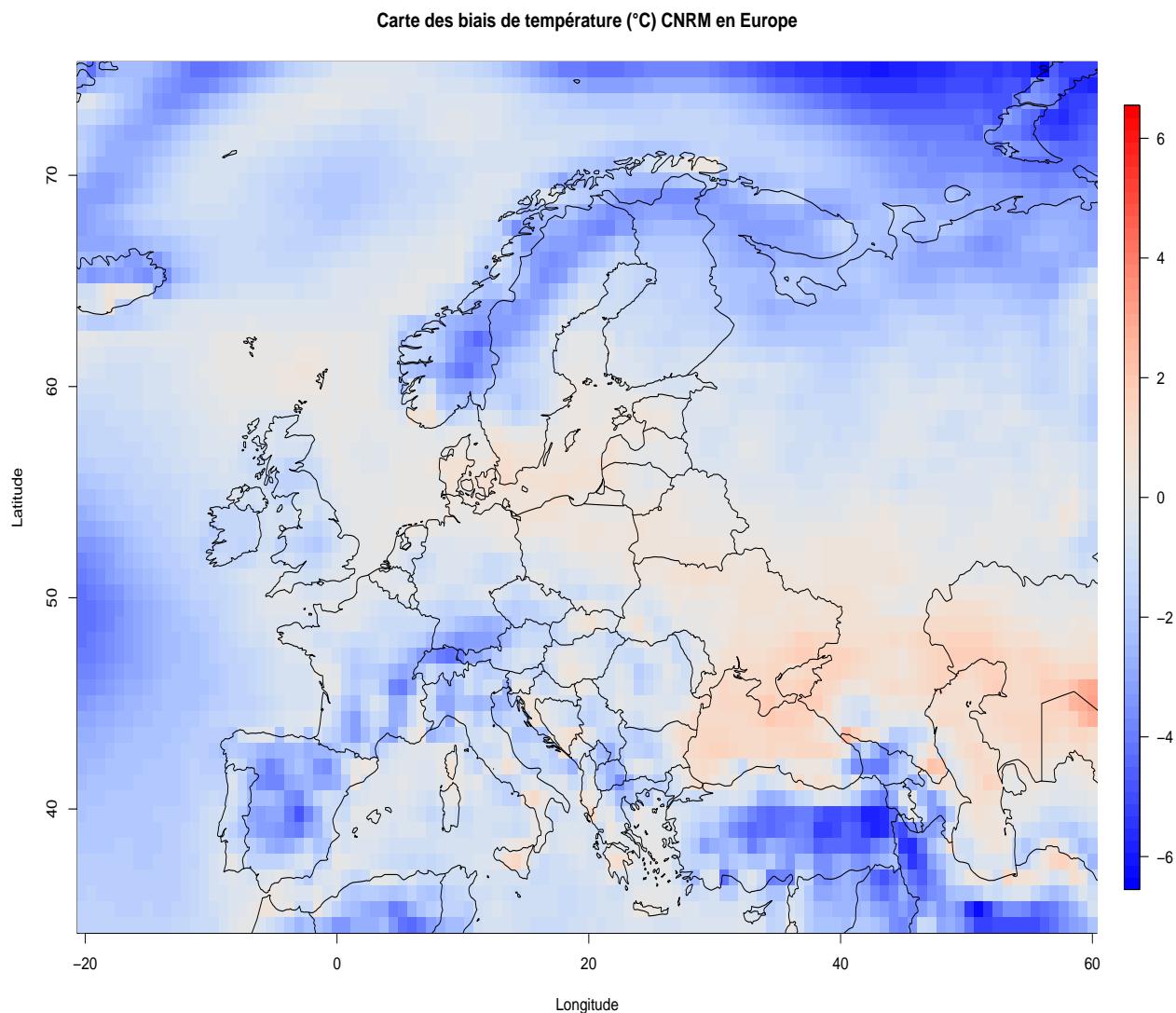


FIGURE 19 – Carte des biais de température ( $^{\circ}\text{C}$ ) du modèle CNRM en Europe

Carte des biais de température ( $^{\circ}\text{C}$ ) IPSL en Europe

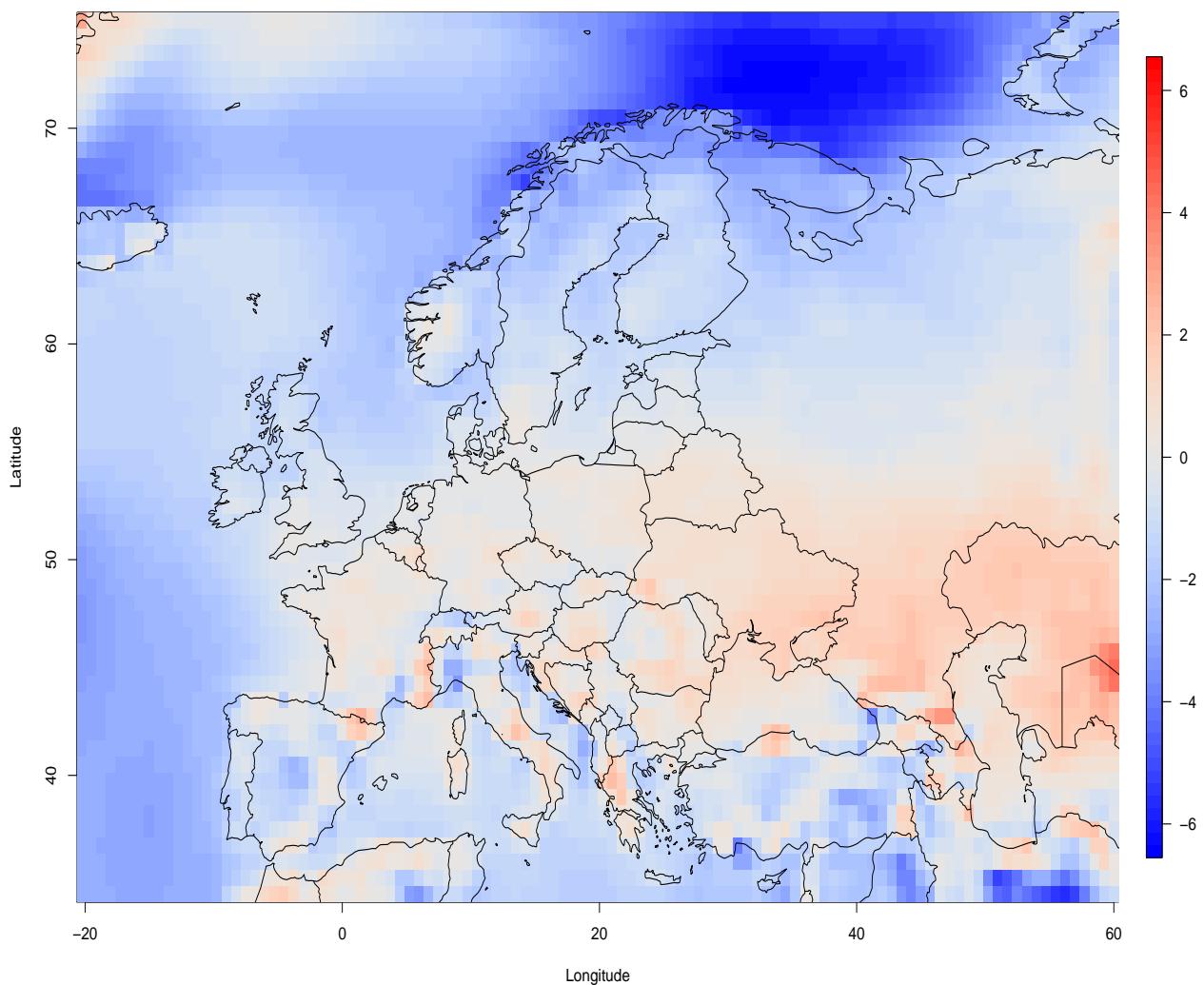
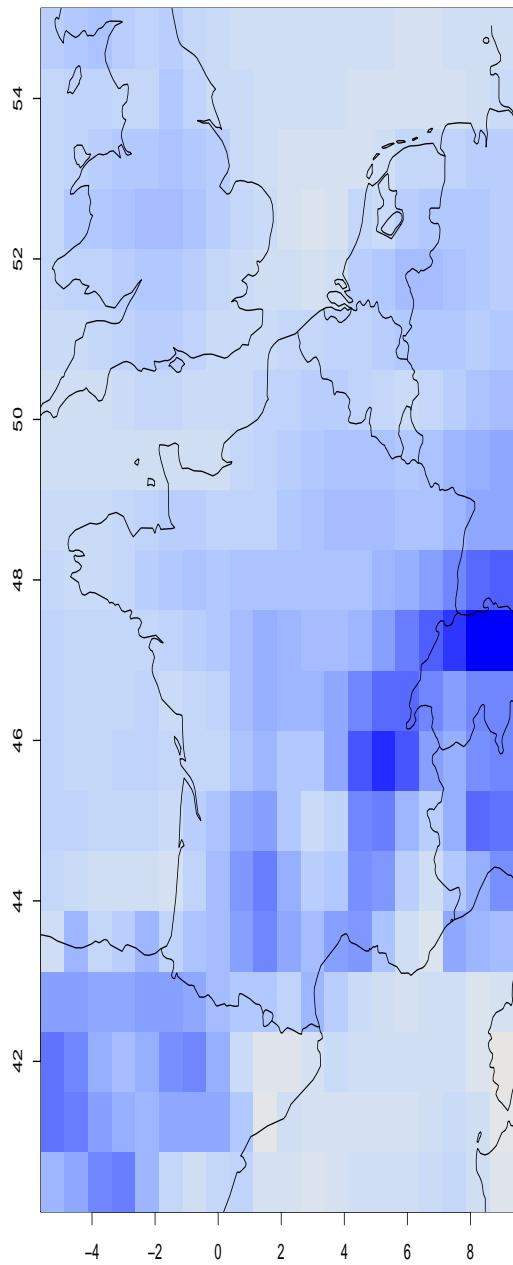


FIGURE 20 – Carte des biais de température ( $^{\circ}\text{C}$ ) du modèle IPSL en Europe

Biais des températures ( $^{\circ}\text{C}$ ) du modèle CNRM en France au printemps



Biais des températures ( $^{\circ}\text{C}$ ) du modèle IPSL en France au printemps

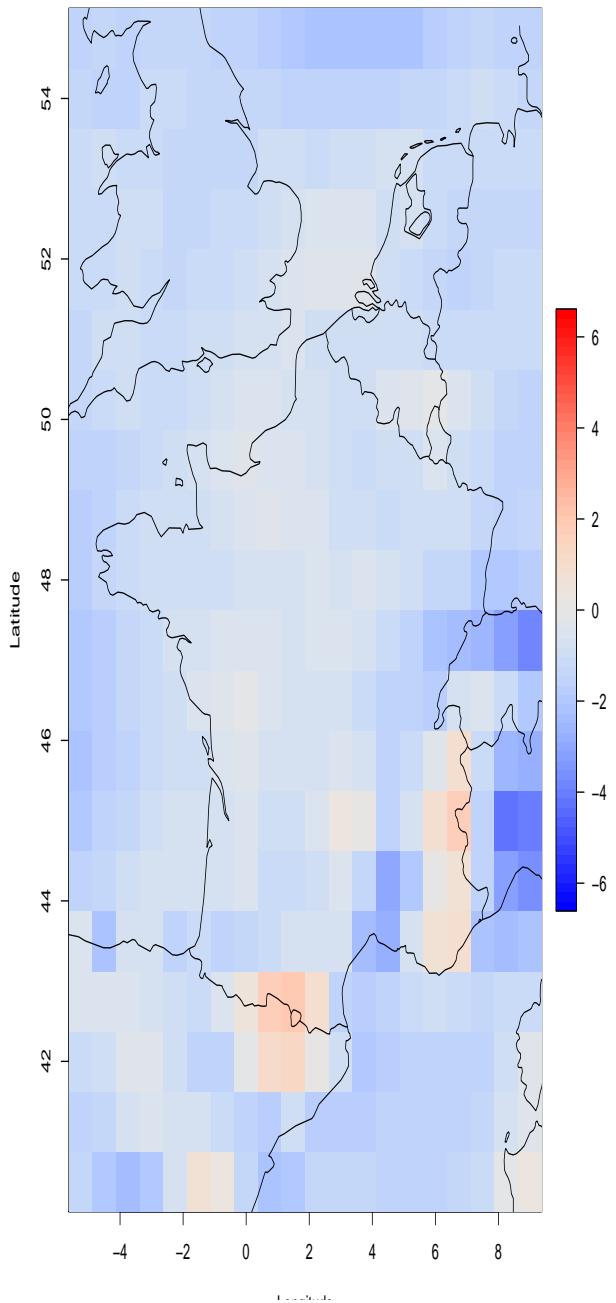
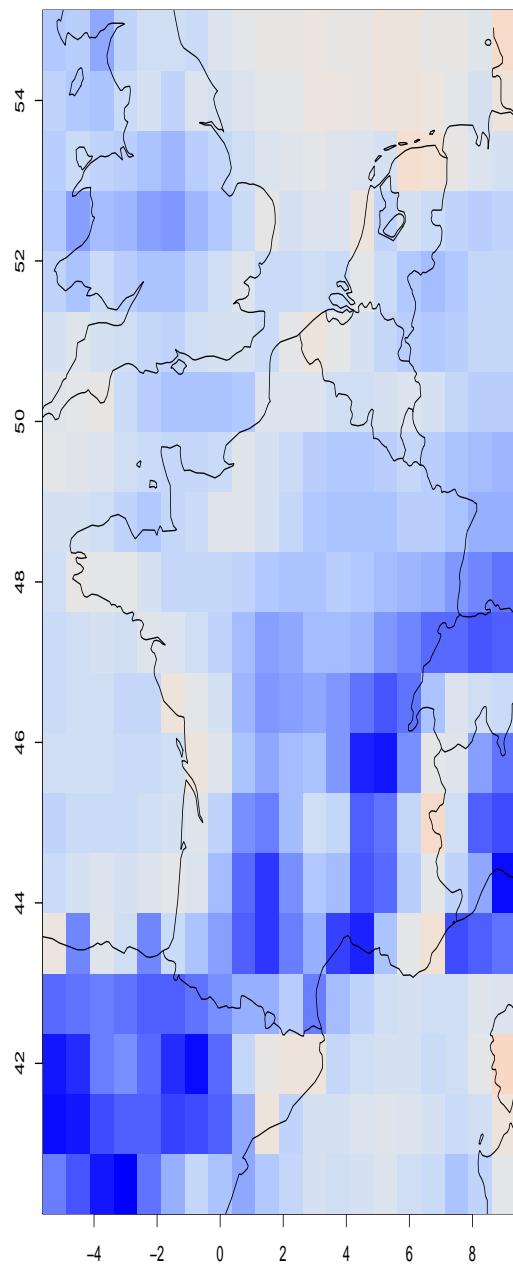


FIGURE 21 – Carte des biais de température ( $^{\circ}\text{C}$ ) du modèle CNRM (gauche) et du modèle IPSL (droite) en France au printemps

Biais des températures ( $^{\circ}\text{C}$ ) du modèle CNRM en France à l'automne



Biais des températures ( $^{\circ}\text{C}$ ) du modèle IPSL en France à l'automne

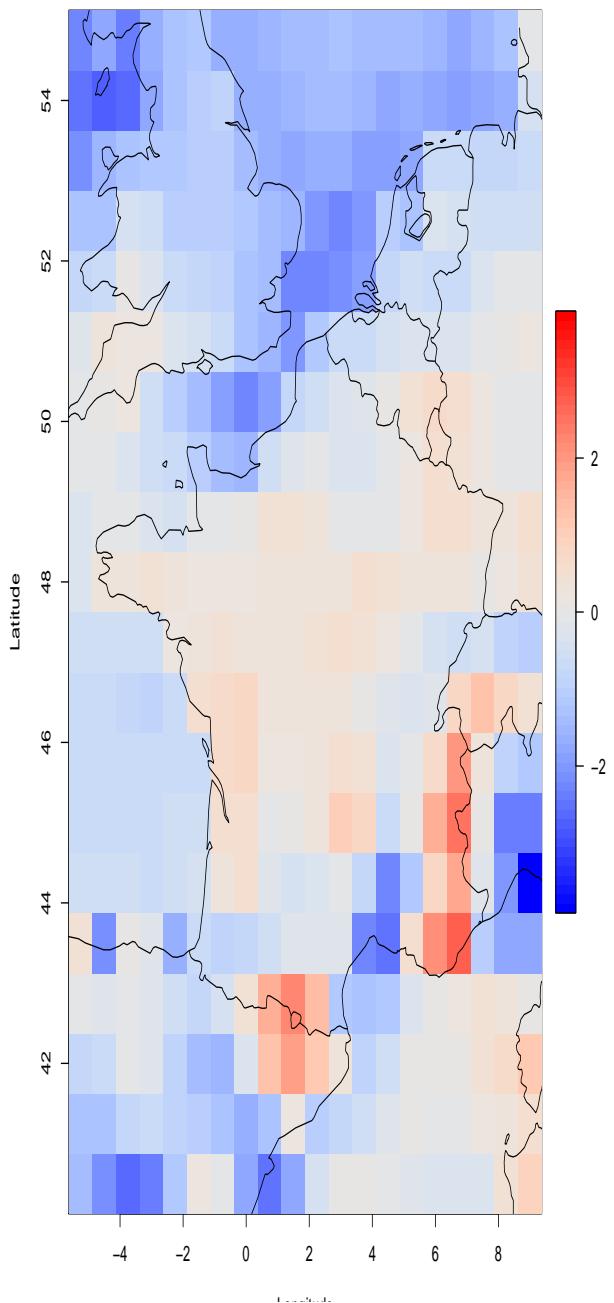


FIGURE 22 – Carte des biais de température ( $^{\circ}\text{C}$ ) du modèle CNRM (gauche) et du modèle IPSL (droite) en France en automne

Biais de température ( $^{\circ}\text{C}$ ) du modèle CNRM en Europe au printemps

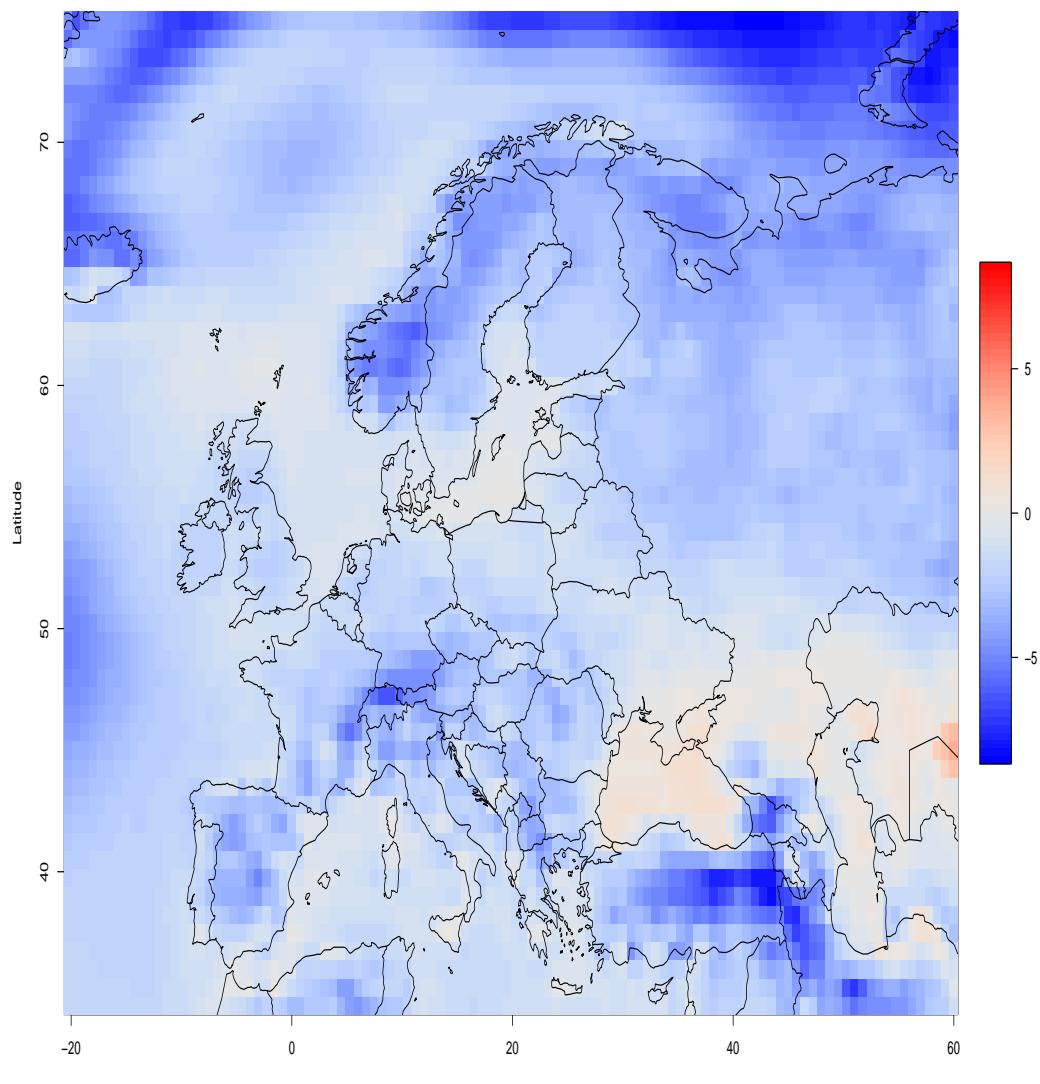


FIGURE 23 – Carte des biais de température ( $^{\circ}\text{C}$ ) du modèle CNRM en Europe au printemps

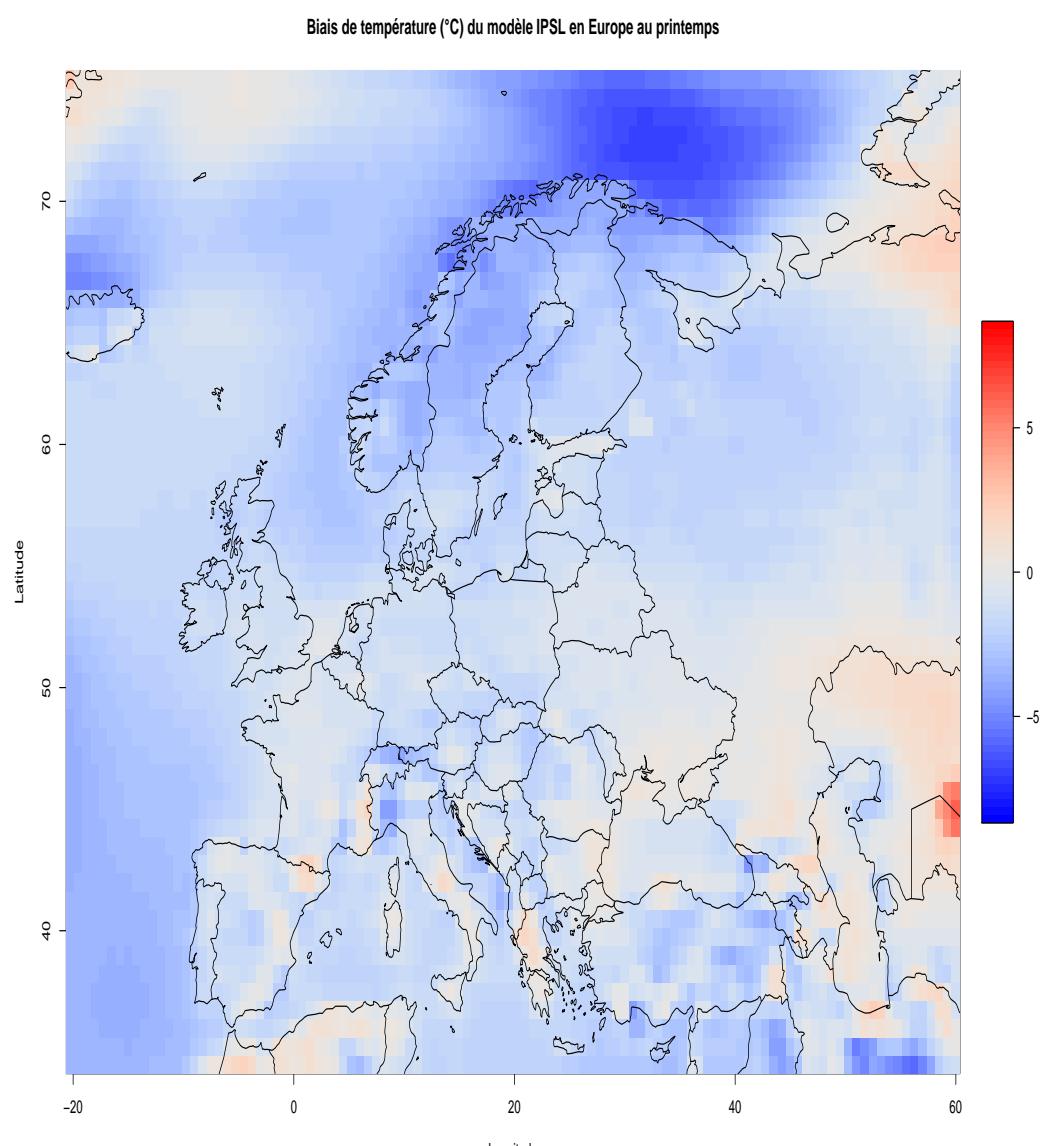


FIGURE 24 – Carte des biais de température ( $^{\circ}\text{C}$ ) du modèle IPSL en Europe au printemps

Biais de température ( $^{\circ}\text{C}$ ) du modèle CNRM en Europe à l'automne

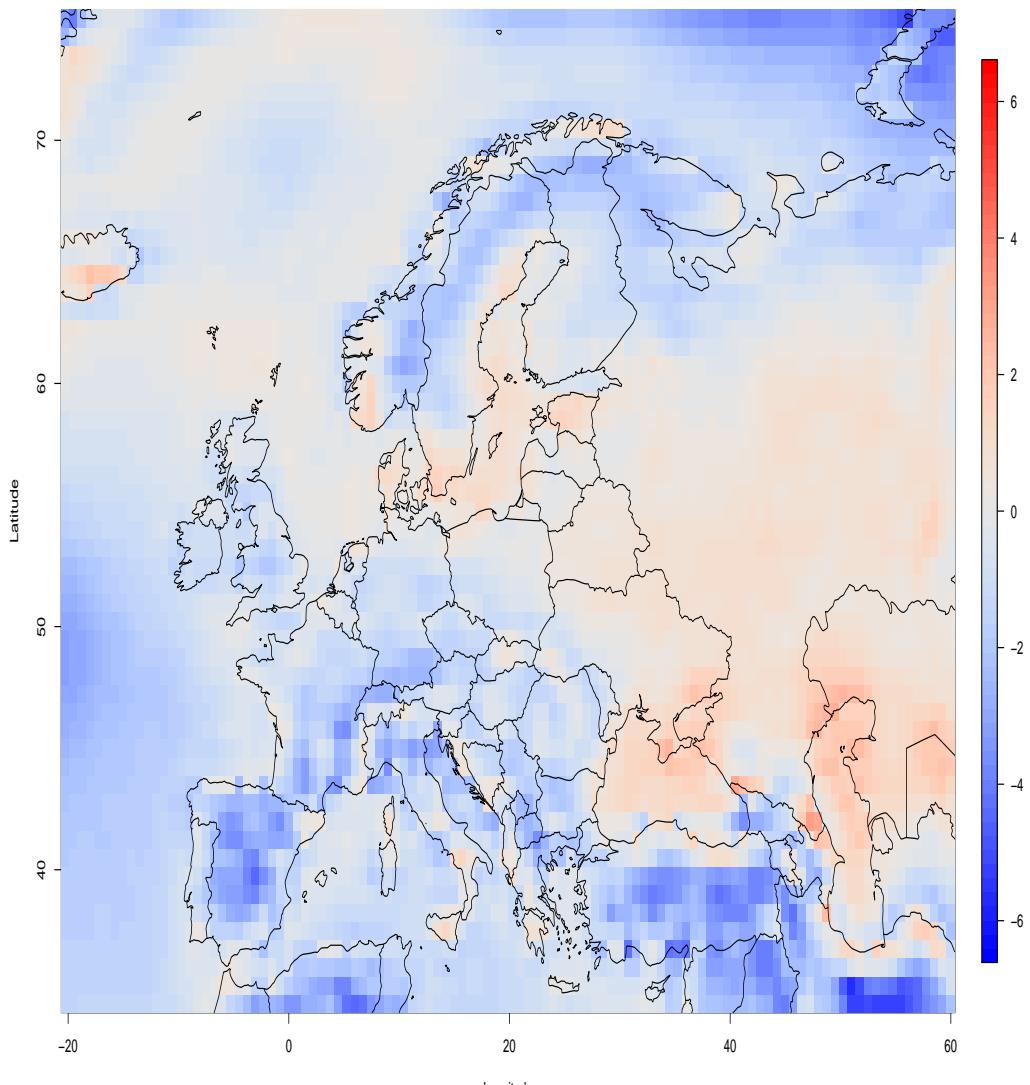


FIGURE 25 – Carte des biais de température ( $^{\circ}\text{C}$ ) du modèle CNRM en Europe en automne

Biais de température ( $^{\circ}\text{C}$ ) du modèle IPSL en Europe à l'automne

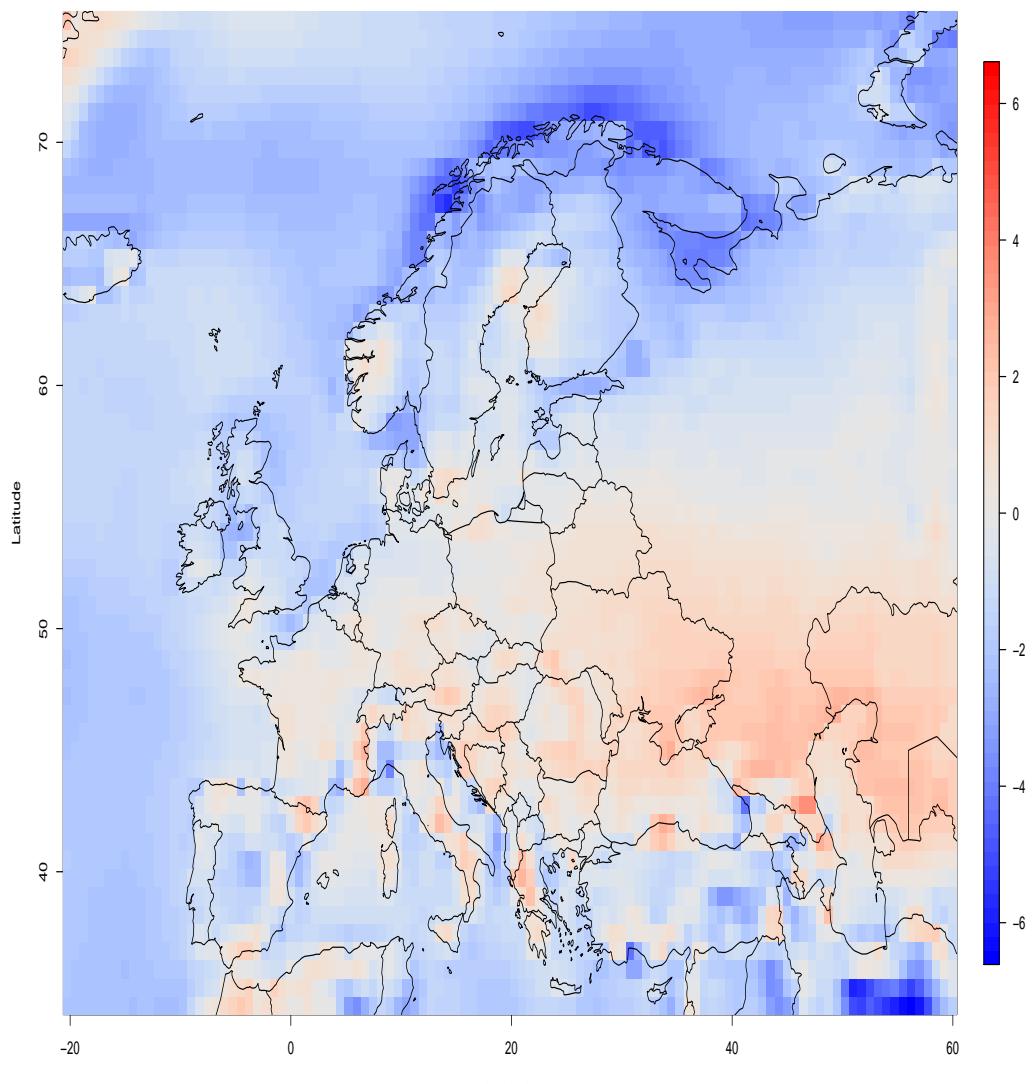


FIGURE 26 – Carte des biais de température ( $^{\circ}\text{C}$ ) du modèle IPSL en Europe en automne

## C Plots

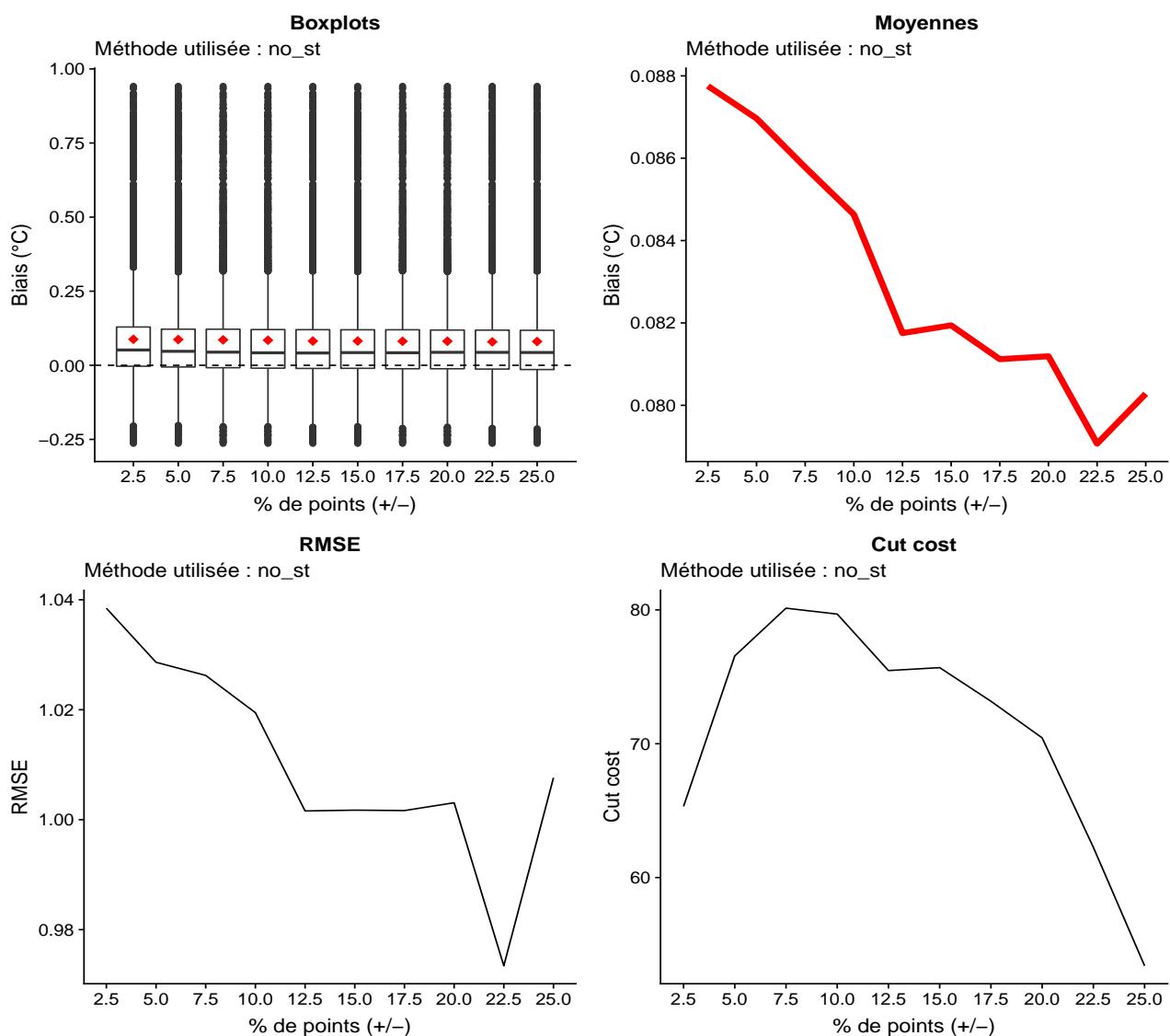


FIGURE 27 – Plots (boxplots, moyennes, RMSE, coût total du cut) de *sd\_no\_st* en Europe

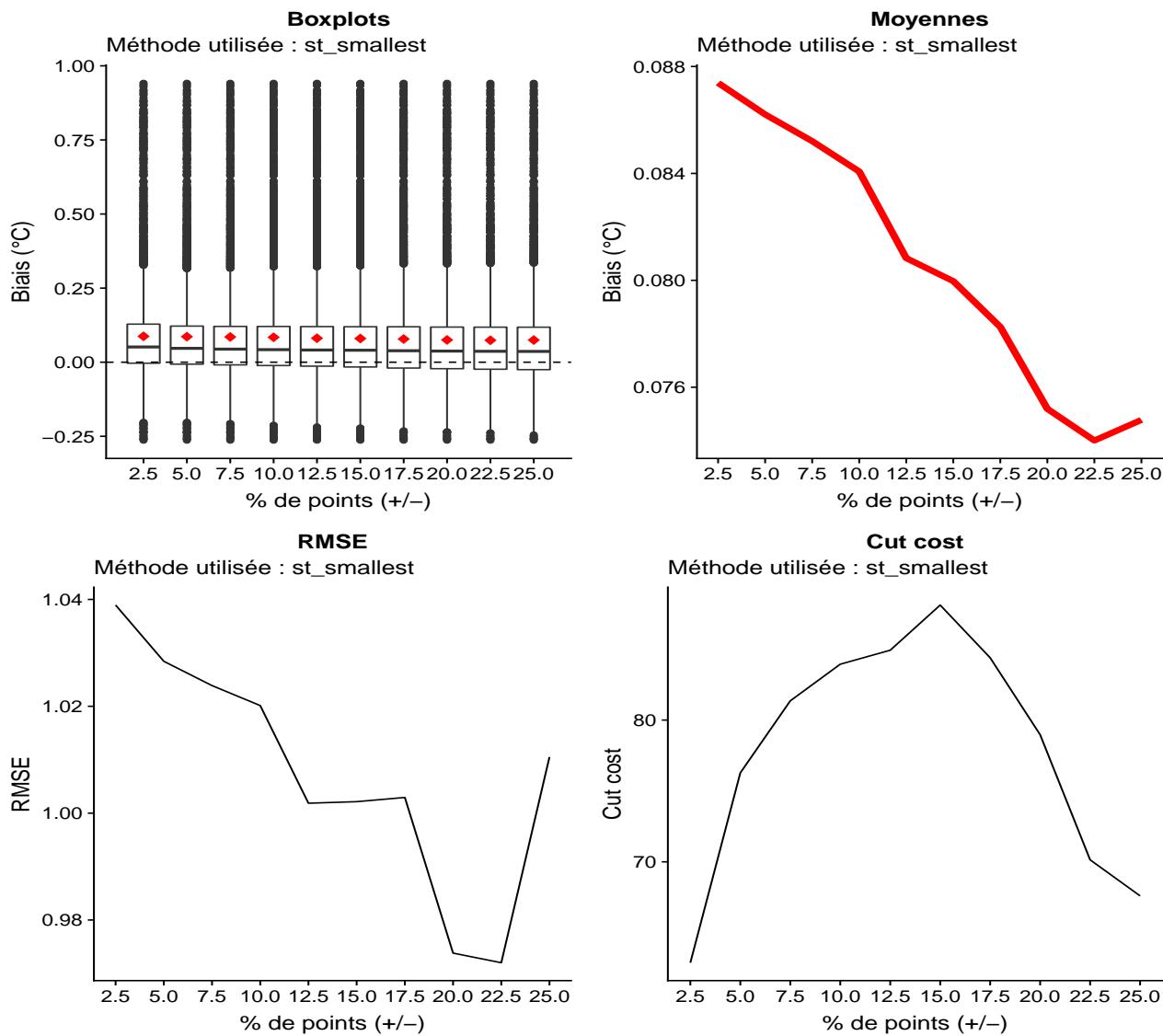


FIGURE 28 – Plots (boxplots, moyennes, RMSE, coût total du cut) de *sd\_st\_smallest* en Europe

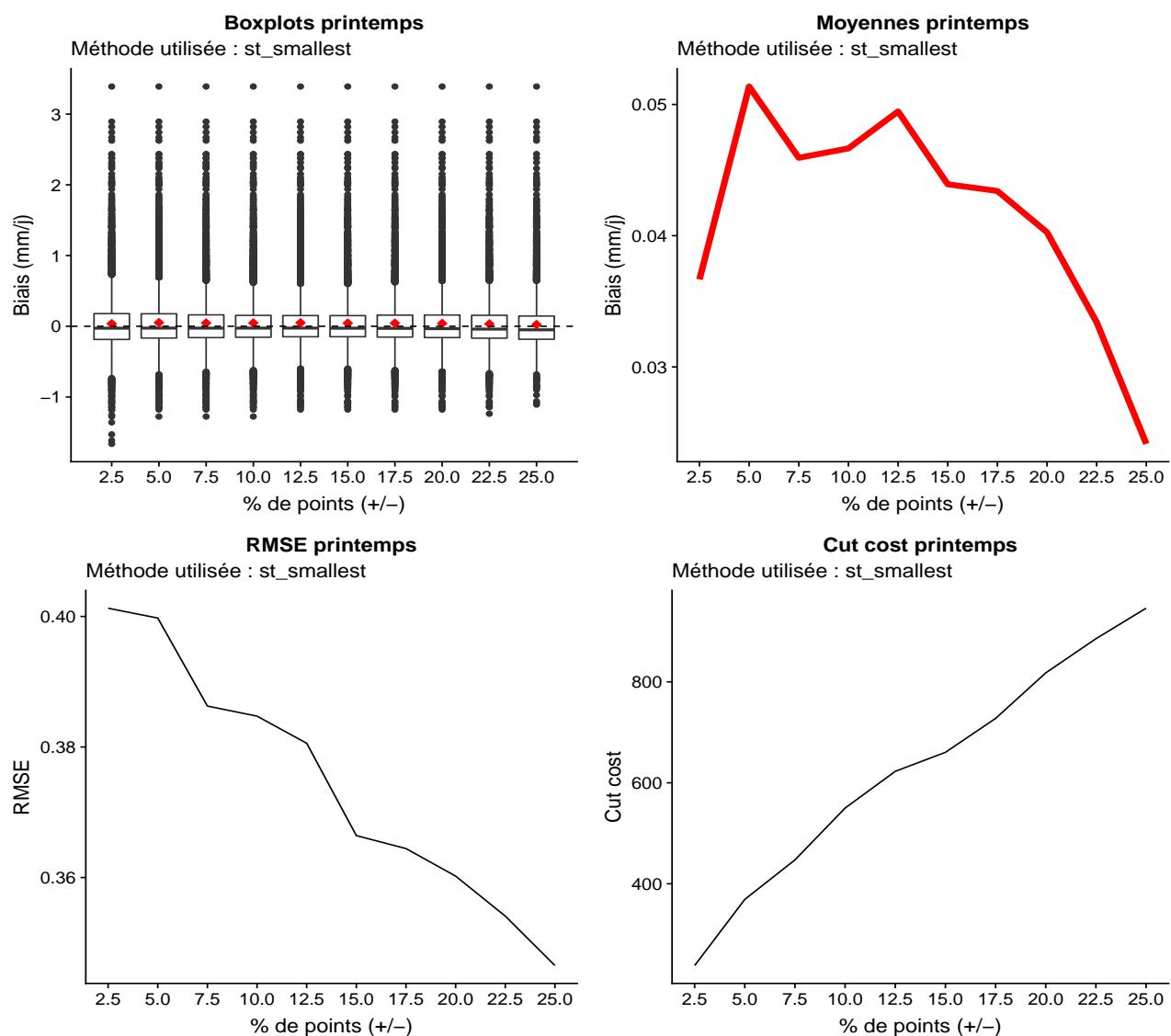


FIGURE 29 – Plots (boxplots, moyennes, RMSE, coût total du cut) de *sd\_st\_smallest* en Europe

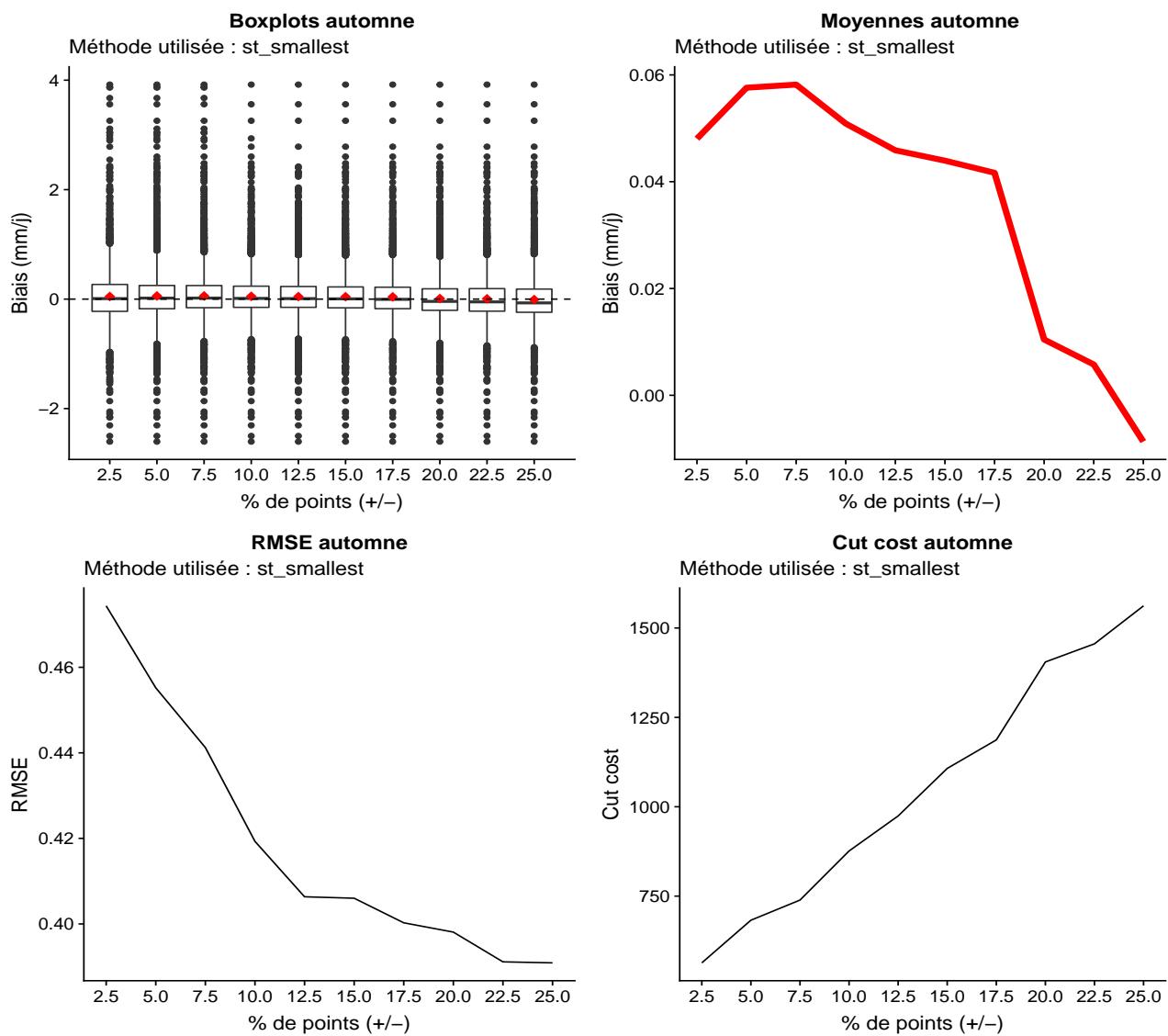


FIGURE 30 – Plots (boxplots, moyennes, RMSE, coût total du cut) de *sd\_st\_smallest* en Europe