TP d'assimilation de données

Utilisaiton du nudging sur la température de surface (SST)

décembre 2020

Auteur : DUPONT Ronan

Enseignant : M. Ourmieres

Table des matières

1 Introduction

Depuis l'arrivée du numérique, une émergence de modèles numériques sont arrivés pour modéliser les phénomènes physiques qui nous entours.
Afin d'avoir des estimations au plus fiable de ces phénomènes, les scientifiques ont élaboré des méthodes d'assimilation de données permettant de mettre en corrélation des observations avec des modèles numériques.
Durant ce TP que l'on codera en Matlab, on utilisera la méthode de nudging direct sur des données observés ainsi que le modèle numérique *GLAZUR64* : un modèle simulant la méditerranée à (1/64° soit 1,2 km). Pour réaliser cette méthodes ,on verra dans un premier temps comment mettre toutes les données à la même échelle (regriller) afin de pouvoir les traiter. Ensuite on exploitera les résultats et on s'intéressera aux erreurs et écarts-type.

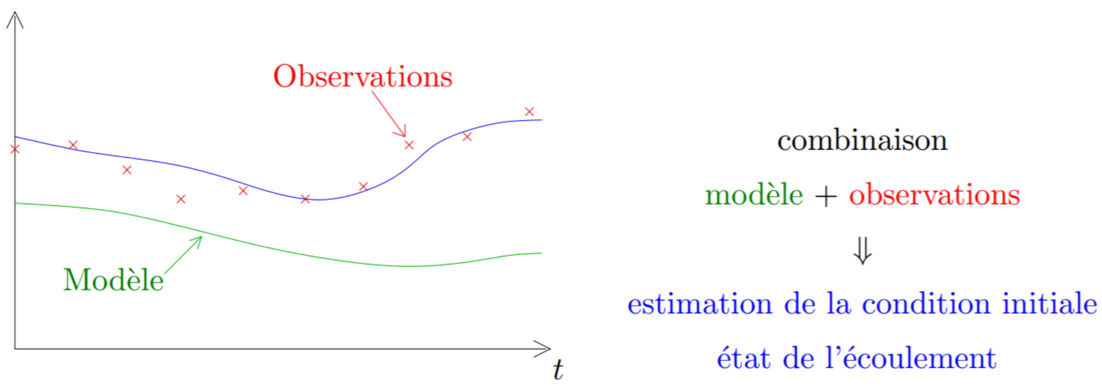


FIGURE 1 – source :math.unice.fr

2 Adapter les données en "regrillant" pour avoir tout à la même échelle

Dans un premier temps, on a besoin de "regriller" les données d'observation et de vérité afin de les corréler avec les données de référence qui proviennent ici du modèle *GLAZUR64*.
On créer donc la fonction ci-dessous en prenant en argument d'entrée une matrice initiale (celle que l'on va regriller) ainsi qu'une matrice de référence qui donnera les dimenssions de la matrice voulue. La fonction sort une matrice qui a les dimensions de la matrice de référence.

```
function [Mat_interp]=grid(Mat_ini,Mat_reference)
[sy,sx] = size(Mat_ini); %On recupere la taille de la matrice initiale
[sY,sX] = size(Mat_reference); %On recupere la taille de la matrice a interpoler

[X,Y] = meshgrid(linspace(1,sX,sx),linspace(1,sY,sy)); %abs/ord des petites matrice
[Xq,Yq] = meshgrid(1:sX,1:sY); %abs/ord de ma matrice de ref

Mat_interp = griddata(X,Y,Mat_ini,Xq,Yq);
end
```

On créer les matrices X, Y en `meshgrid` indiquant les dimensions initiales en abscisse et ordonnées. De même pour les matrices Xq et Yq pour les dimensions de la matrice de référence.
Ensuite, la fonction `griddate` s'occupe de faire l'interpolation pour passer de la grille initiale jusqu'à la grille de référence.

On utilisera donc cette fonction dans notre `main` sur le programme ci-dessous afin d'avoir les bonnes grilles pour les données d'observation et de vérité.

```
model= ncread("GLAZUR64_20180707_20180720_grid_T.nc ","votemper") ;
obs= ncread("SST_MED_OBS_0707-2007-2018.nc ","analysed_sst")-273 ;
truth= ncread("reanalyse_SST_CMEMS.nc ","analysed_sst")-273 ;

Mat_reference=model(:,:,1,1); %Matrice de reference pour sa taille

Tm=zeros(377,230,14); %Tableau modele
Tobs=zeros(377,230,14); %Tableau observation
Tt=zeros(377,230,14); %Tableau verite

for i=1:14
    Tobs(:,:,i)=grid(obs(:,:,i),Mat_reference);
    Tt(:,:,i)=grid(truth(:,:,i),Mat_reference);
    Tm(:,:,i)=model(:,:,1,i);
end
```

On trace donc les grilles initiales et les grilles transformés à l'aide du programme ci-dessous et nous obtenons les graphiques figure 2 ci-dessous :

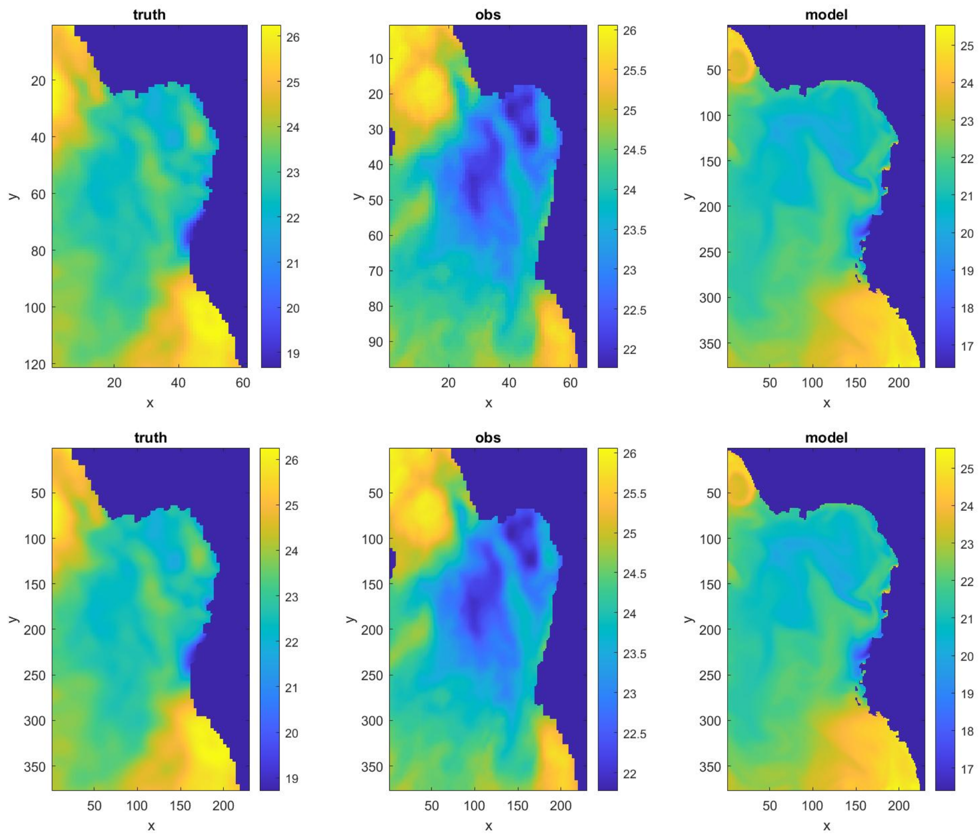


FIGURE 2 – Données au premier jour une fois "regrillé"

On remarque que les côtes ont été comme "lissés" grave à l'interpolation. Sinon on observe sensiblement les mêmes données. On peut donc valider notre opération de regrillage.

3 Obtention des états analysés par nudging

Le principe du nudging met en corrélation des données d'observation avec des données de modèle. Il sert à obtenir des données plus fiables.

On défini $\frac{\partial T}{\partial t} = M(T)$ où $M(t)$ est l'évolution du modèle de température qui aura donc pour valeur $M(T_k) - M(T_{k-1})$ soit $\frac{T_k^M - T_{k-1}^M}{dt}$.

On obtient donc :

$$T_k^M = T_{k-1}^M + dt(M(T_k) - M(T_{k-1})$$

On applique un nudging de la forme :

$$\frac{\partial T}{\partial t} = M(T) + \alpha(T_{obs} - T)$$

où $\alpha \in [0,2]$. On obtient donc :

$$T_k^A = T_{k-1}^A + dt[(M(T_k) - M(T_{k-1})) + \alpha(T_{k-1}^{obs} - T_{k-1}^A)] \tag{1}$$

Avec $M(T_k) - M(T_{k-1}) = \frac{T_k^M - T_{k-1}^M}{dt}$, (1) devient :

$$T_k^A = T_{k-1}^A + (T_k^M - T_{k-1}^M) + dt.\alpha(T_{k-1}^{obs} - T_{k-1}^A) \tag{2}$$

On peut donc obtenir la matrice T_a des températures analysés avec le programme ci-dessous où nous utiliserons l'équation (2).

```
% Initialisation
Ta=zeros(377,230,14); %Tableau analyse par nudging
Ta(:,:,1)=Tm(:,:,1); %Condition initiale du tableau
a=0.5; %alpha
dt=1; %pas de temps

for k=2:14 %Formule du nudging
   Ta(:,:,k)=Ta(:,:,k-1)+Tm(:,:,k)-Tm(:,:,k-1)+a*dt*(Tobs(:,:,k-1)-Ta(:,:,k-1));
end
```

4 Résultats

4.1 Comparaison des différents modèles

Une fois nos formule analysé obtenue, on a cherché à comparer les résultats des différents modèles. Par exemple figure 3 ci-dessous, on a tracé les différents modèles avec $\alpha = 0.75$ au jour 14.

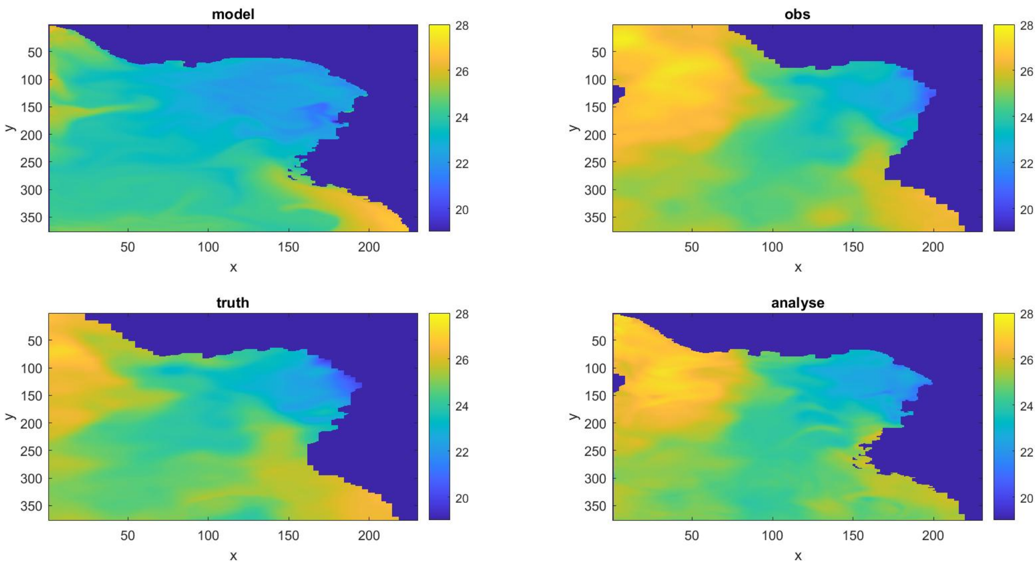


FIGURE 3 – Comparaison des modèles pour $\alpha = 0.75$ au jour 14

On remarque que le modèle analyse se rapproche fortement du modèle réel (truth). On a donc cherché à évaluer le bon coefficient α afin d'obtenir un résultat cohérent.

4.2 Recherche du meilleur coefficient α

Pour trouver le meilleur α, on a comparé nos valeurs réels de SST avec des modèles par nudging avec différents α au jour 7 et 14 comme on peut voir figure 4 et 5 ci-dessous.

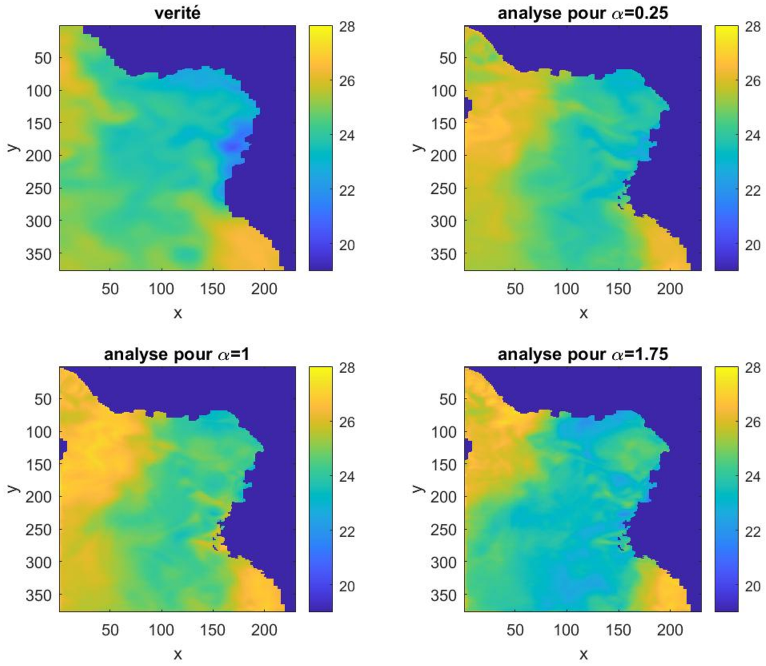


FIGURE 4 – Comparaison des modèles pour différents α au jour 7

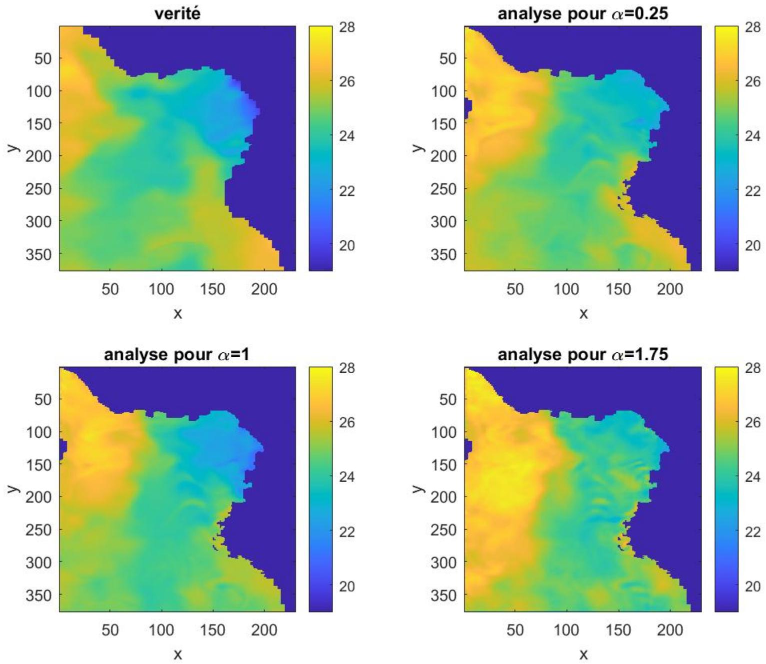


FIGURE 5 – Comparaison des modèles pour différents α au jour 14

Dans les deux cas, cela semble difficile à trancher sur lequel α est le meilleur. On peut voir qu'au jour 14, le α ne convient vraiment pas.
On a donc cherché à obtenir des données plus précises permettant de savoir quel α est le meilleur. Pour se faire, on a calculé à l'aide de 6.1 l'erreur moyenne sur la grille à chaque jours en fonction de différents α. On a obtenu la figure 6 suivante :

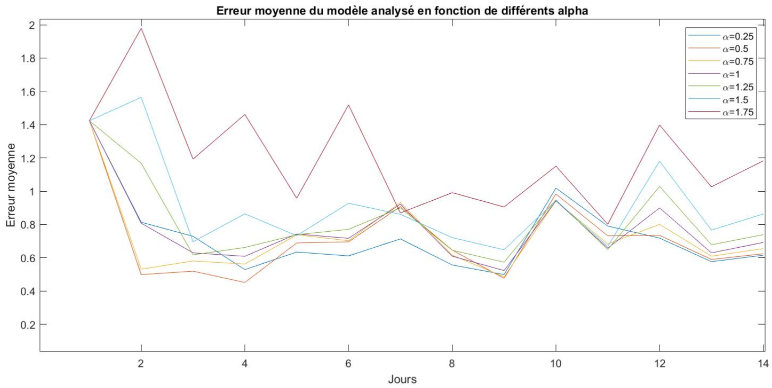


Figure 6 – Calcul d'erreur moyenne avec différents α en fonction des jours

On remarque sur cette figure que parfois certains α sont plus adaptés que d'autres sur certains jours. Mais globalement, on peut dire que le $\alpha = 1.75$ est mauvais.
On a ensuite calculé les erreurs moyennes sur les jours et obtenu les résultats suivants :

α	0.25	0.5	0.75	1	1.25	1.5	1.75
Erreur moyenne	0.73	0.71	0.73	0.77	0.82	0.91	1.20

Le α qui semble le meilleur est celui à 0.5 mais celui-ci ne se démarque pas tant que ça des autres. En prenant un $\alpha \in [0.25,1]$, celui-ci semble être un bon candidat.

Ensuite pour aller plus loin, on peut regarder où sont les zones à fortes erreurs en traçant les erreurs ci-dessous figure 7.

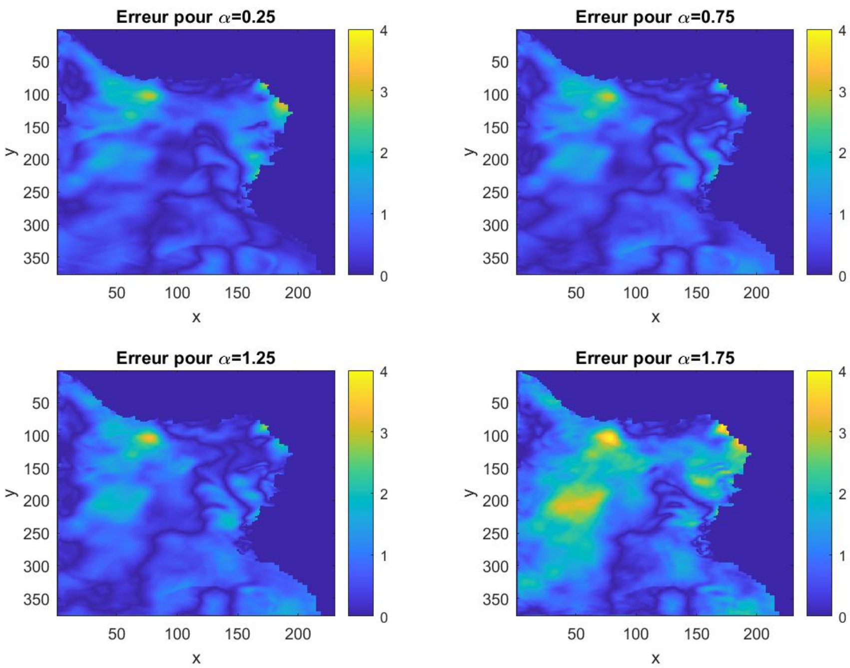


Figure 7 – Erreur absolue avec différents α au jour 14

On remarque sans surprise que l'approximation est mauvaise au niveau des côtes car celles-ci sont influencés par les conditions limites.

4.3 Graphiques de statistiques

Une fois notre modèle "validé", on a cherché dans un premier temps à tracer la moyenne de température journalière figure 8 ci-dessous.

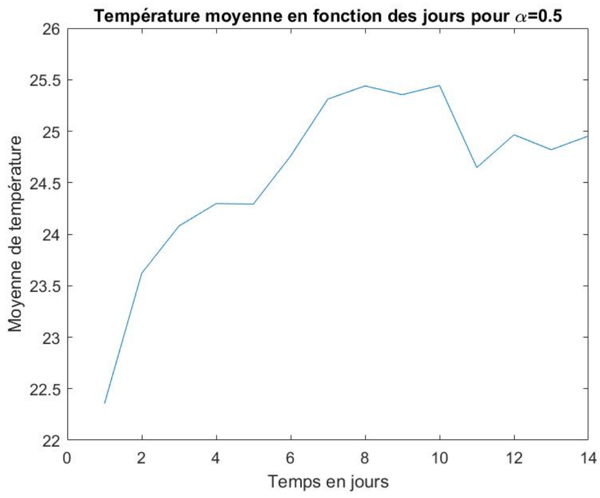
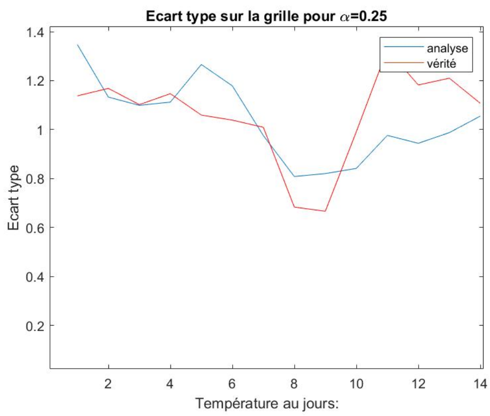


FIGURE 8 – Température moyenne en fonction des jours pour α=0.5

Puis ensuite on a comparé l'écart-type des valeurs de SST avec le programme 6.2 en utilisant la formule :

$$\sigma = \sqrt{V} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2}$$

où les $(x_1, ..., x_n)$ sont ici les SST en tous points de la grille.



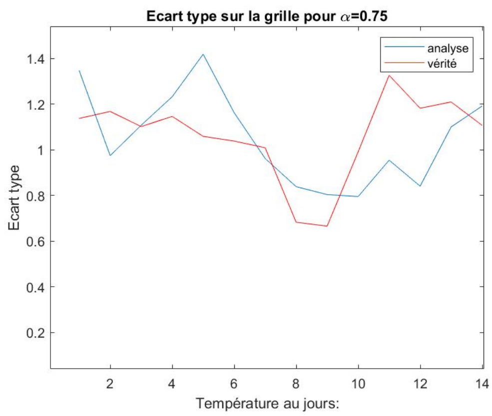
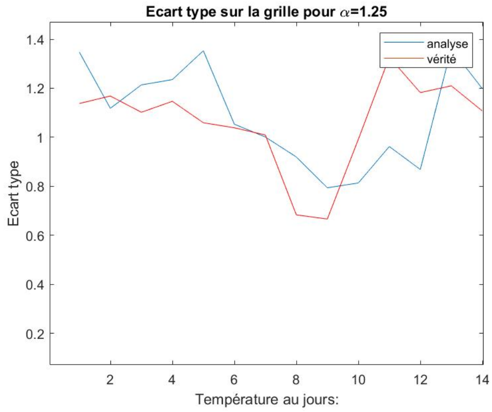


FIGURE 9 – $\alpha = 0.25, 0.75$



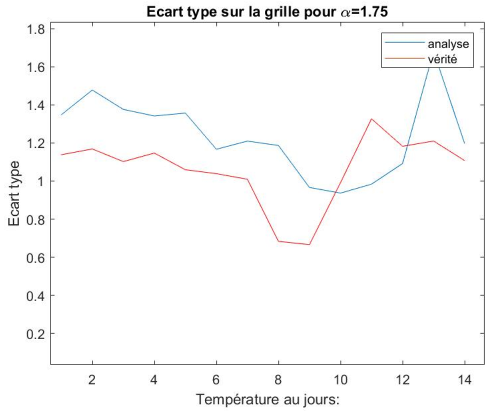


FIGURE 10 – $\alpha = 1.25, 1.75$

On remarque que globalement, l'écart type des SST oscille entre 1 et 1.5 quelque soit le modèle et quelque soit le α. Ceci montre donc qu'il y a une variation de températures assez forte en 2 semaines. En effet, d'après la figure 8, on passe d'une température à 22,5 °C à une température à 25 °C en seulement 1 semaine.

5 Conclusion

Durant ce TP réalisé avec Matlab, on a pu étudier une méthode d'assimilation de données appelée nudging direct. Celle-ci permet d'obtenir des résultats cohérent en utilisant des données d'observations ainsi que des données d'un modèle numérique.
On a pu essayer de trouver le meilleur coefficient de nudging α en calculant les erreurs en fonction de différents α. On a remarqué que celui-ci doit être compris entre 0.25 et 1 pour sortir un résultat cohérent.
Ensuite, on a pu observer les données de SST avec de l'analyse statistique en utilisant l'écart-type et la moyenne qui nous a montré que la SST dans la zone étudiée a évolué rapidement.

6 Annexe

6.1 Calcul d'erreurs

```
Er=zeros(14);
for k=1:14
    E=abs(Ta(:,:,k)-Tt(:,:,k));
    Er(k)=nanmean(nanmean(E));
end
```

6.2 Calcul ecarts-types

```
Sa=zeros(14);
St=zeros(14);
for k=1:14
    A=Ta(:,:,k);
    T=Tt(:,:,k);
    ma=nanmean(nanmean(A));
    mt=nanmean(nanmean(T));
    ca=0;
    sa=0;
    ct=0;
    st=0;
    for i=1:377
        for j=1:230
            if not(isnan(A(i,j)))
                sa=sa+(A(i,j)-ma)^2;
                ca=ca+1;
            end
            if not(isnan(T(i,j)))
                st=st+(T(i,j)-mt)^2;
                ct=ct+1;
            end
        end
    end
    Sa(k)=(1/ca*sa)^0.5;
    St(k)=(1/ct*st)^0.5;
end
```