Guide pour les travaux pratiques sur les modèles dynamiques.

**François Brun and contributions of other teachers**

**Version 2015-11-28**

**Table of Contents**

[1. Liste des ressources : données et script 1](#_Toc436484982)

[2. Un modèle de phénologie d'un insecte, le charançon de la carotte, avec une fonction, une boucle et une condition. 2](#_Toc436484983)

[3. Un modèle SEIR simple basé sur le modèle de Zadoks appliqué à la Rouille Brune. 3](#_Toc436484984)

[a. Description du modèle de Zadoks (1971). 3](#_Toc436484985)

[b. Structure du modèle. 3](#_Toc436484986)

[c. Equations du modèle d'origine. 4](#_Toc436484987)

[d. Equations d'une version simplifiée du modèle. 4](#_Toc436484988)

[e. Ajouter un effet climatique sur les processus. 5](#_Toc436484989)

[f. Plus de réalisme avec les files d'attente et la version d'origine du modèle de Zadoks (1971). 6](#_Toc436484990)

[g. Comment ces différents modèles représentent le système réel ? 6](#_Toc436484991)

[4. Utilisation du modèle de Zadoks pour analyser et comparer des scénarios. 7](#_Toc436484992)

[a. Objectif 7](#_Toc436484993)

[b. Comment prendre en compte les pratiques dans le modèle de Zadoks ? 7](#_Toc436484994)

[c. Effet du changement climatique sur les épidémies de rouille brune en France. 8](#_Toc436484995)

# Liste des ressources : données et script

Dans ce document, on va utiliser différents jeux de données et script R pour les travaux pratique.

Vert : corrections disponibles

|  |  |  |
| --- | --- | --- |
| **type** | **fichiers** | **utilisation** |
| rscript | models/logistic.model\_correction.r | Jour 1 |
| rscript | models/carrot.weevil.model\_correction.r | Jour 4 |
| rscript | models/zadoks1971.simple.model\_correction.r | Jour 4 |
| rscript | models/zadoks1971.original.model\_correction.r | Jour 4 |
| rscript | models/zadoks1971.simpleweather\_correction.model.r | Jour 4 |
| rscript | explore.data/explore\_leaf\_rust\_wheat.dataBancal2008\_correction.r |  |
| data | weather/DRIAS/SCRATCH08\_arpege\_v4\_retic.rda |  |
| data | weather/DRIAS/SCRATCH08\_arpege\_v4\_retic\_a1b\_1.rda |  |
| data | weather/DRIAS/SCRATCH08\_arpege\_v4\_retic\_a2\_1.rda |  |
| data | weather/DRIAS/SCRATCH08\_v4\_retic\_ref\_1.rda |  |
| rscript | projects/ClimateChange.read.weather.drias.r |  |
| rscript | projects/CimateChange\_correction/ClimateChange.leafrust.r |  |
| rscript | projects/CimateChange\_correction/zadoks.simpleweather.model.r |  |
| rscript | projects/CompareModelData\_correction/compareModelData\_correction.r |  |

# Un modèle de phénologie d'un insecte, le charançon de la carotte, avec une fonction, une boucle et une condition.

Un concept très couramment utilisé en agronomie et en protection des cultures est le temps thermique. Il est basé sur le fait que la vitesse de développement de nombreux organismes dépend de la température. Ainsi, un modèle phénologique prédit un calendrier du développement d'un organisme sur la base du cumul du temps thermique. En protection des cultures, il est utilisé pour calculer le développement de ravageurs tels que les insectes, dont le développement dépend de la température de leur environnement.

Dans cet exemple, on s'intéresse au Charançon de la carotte (*Listronotus oregonensis*) qui se développe lorsque la température est au-dessus de 7.0°C. Ce seuil minimum de développement est nommé température de base (Tbase).

Une définition simple du temps thermique (TT) est la somme des températures quotidiennes moyennes (Tmean) moins Tbase. Si la différence est négative (si Tmean < Tbase) alors la somme de température ce jour-là est égale à 0. Le calcul se fait sur une période donnée. L'unité de temps thermique couramment utilisée est °C.jour (voir le cours de JN Aubertot sur l'analyse dimensionnelle).

Pour le charançon de la carotte, nous avons également des informations, issues d'observations, concernant le temps thermique requis pour finir chaque stade de développement

|  |  |
| --- | --- |
| **Stade** | **TT (°C.jour) pour le stade** |
| O - œufs | 130 |
| L - larves | 256 |
| P1 - pré-nymphes | 114 |
| P2 - pupes | 130 |
| temps de génération - œuf à l'âge adulte | 630 |

Référence :

* Simonet, D. E., and B. L. Davenport. 1981. Temperature requirements for development and oviposition of the carrot weevil. Ann. Entomol. Soc. Amer. 74: 312-315.
* <http://www.ipm.ucdavis.edu/PHENOLOGY/ma-carrot_weevil_loreg.html>

### Questions

1. *Ecrire une fonction R prenant comme argument un vecteur de températures moyennes et retournant la somme des TT correspondant au dernier jour de cette série. Utiliser une boucle (for ...) et une condition (if ... else ...) pour cela. Écrire un programme principal qui exécute la fonction.*
2. *Modifier la fonction précédente pour stocker les valeurs de TT à chaque pas de temps dans un vecteur appelé TT. Avant de stocker les valeurs TT, vous devez définir le vecteur de TT. Vous pouvez créer un TT vectoriel où l'aide de l'instruction TT=rep(NA, duration). Modifier la fonction pour que Tbase en soit un argument. Bravo ! Vous avez écrit votre premier modèle dynamique avec R !*
3. *Utiliser les informations sur les besoins en temps thermique pour charançon de la carotte pour proposer un modèle de phénologie complet.*
4. *Utiliser ce modèle sur un scénario de température constante, puis sur des scénarios climatiques réelle en utilisant les données météos de weather\_FranceWest du package ZeBook.*

*library(ZeBook)*

*meteo = subset(weather\_FranceWest,idsite==1&WEYR==2000)*

1. *Proposer une fonction plus simple qui n'utilise ni boucle et ni condition if.*

# Un modèle SEIR simple basé sur le modèle de Zadoks appliqué à la Rouille Brune.

## Description du modèle de Zadoks (1971).

Ce modèle proposé par Zadoks (1971) est un modèle classique SEIR pour simuler des épidémies de maladies des cultures. Les modèles SEIR (*Susceptible-Exposed-Infectious-Removed*) forment une famille de modèles très utilisés dans différents domaines (médicale, vétérinaire, phytopathologie,…).

La culture est considérée comme étant composée d'un nombre fini de sites infectieux. Les dimensions physiques d'un site infectieux coïncident à peu près avec l'unité de reproduction du parasite étudié.

Chaque site est caractérisé par son état :

* S : *Susceptible*. Unité saine, pouvant être infectée.
* E : *Exposed*. Unité infectée, en phase de latence (pour une certaine durée) et non infectieuse.
* I : *Infectious*. Unité infectée et infectieuse (pour une certaine durée).
* R : *Removed*. Unité qui a fini d'être infectieuse, post infectieuse, ne pouvant plus être infectée.

Ce modèle simple est basé sur les concepts épidémiologiques de "durée de latence" (durée de l'état E), de "durée infectieuse" (durée de l'état I) et de "facteur de multiplication" (nombre de sites contaminables à partir d'une unité infectieuse par jour). Différents pathosystèmes (avec différentes définitions pour les sites infectieux) peuvent être considérés avec ce modèle.

Une description complète est disponible dans la publication d'origine qui vous est fournie :

*Zadoks, J.C. 1971. Systems Analysis and the Dynamics of Epidemics. Phytopathology. 61:441-598.*

Il s'agit d'un modèle relativement simple, voire simpliste. En effet, il existe de nombreux autres modèles plus récents qui incluent plus de connaissance sur les mécanismes épidémiologiques en jeux. Néanmoins, les concepts du modèle de Zadoks sont à la base de certains de ces modèles plus complexes. Il s'agit donc d'un bon point de départ pour comprendre le fonctionnement d'un modèle dynamique.

## Structure du modèle.

Le modèle a quatre variables d'état (nécessaires et suffisantes pour décrire le système modélisé) représentant le nombre de sites dans chaque état.

|  |  |  |
| --- | --- | --- |
| Variable d'état  (à utiliser dans R) | Origine du nom | Etat SEIR correspondant |
| XVAC | vacant | S : *Susceptible*. |
| XLAT | latent | E : *Exposed*. |
| XINF | infectieux | I : *Infectious*. |
| XCTR | cumulative total of removal | R : *Removed*. |

Deux variables supplémentaires basées sur les variables d'état sont utilisées pour plus de facilité :

* XTO1 = XLAT + XINF + XCTR
* XSEV = XINF + XCTR

Les flux ou taux d'évolution des variables d'état sont définis :

* rocc (occupation)
* rapp (apparition
* rrem (removal)

Le modèle peut être représenté selon un diagramme Forrester (issu de Zadoks, 1971).



Cette dynamique est définie avec un pas de temps journalier (dt = 1 jour). Le système étudié représente un hectare d'une culture de blé.

## Equations du modèle d'origine.

Les équations à différence décrivant le modèle sont décrites ci-dessous.

|  |
| --- |
| **Définition de la structure du modèle avec des équations à différence.**  XVAC(day+1) = XVAC(day)-rocc  XLAT(day + 1) = XLAT(day) +rocc - rapp  XINF(day+1) = XINF(day) + rapp - rrem  XCTR(day+1) = XCTR(day) + rrem  **Durée de latence et d'état infectieux**  Afin de représenter les processus de délais des états de latence et infectieux, les deux variables d'état XLAT et XINF sont définis comme deux files d'attente (boxcar trains) : BOXL (latent) et BOXI (infectieux). Par exemple, la structure BOXL stocke des informations sur chaque cohorte, à savoir le nombre de sites qui entrent en latence chaque jour, et détermine le moment où se termine la période de latence (paramètre nlpd) de cette la cohorte de (de NLPD).  **Définition des flux**  # rocc: taux d'occupation : nb de sites Vacant=>Latent  rocc = cofr \* DMFR \* XINF  avec  # rapp: taux d'apparition : nb de sites Latent=>Infectant  rapp= outflow(XLAT)= outflow(BOXL)  # rrem: taux de "disparition" : nb de sites Infectant=>removed  rrem= outflow(XINF) = outflow(BOXI)    Variables d'intérêt auxiliaires :  XTO1 = XLAT+XINF+XCTR  XSEV = XINF+XCTR |

## Equations d'une version simplifiée du modèle.

Prendre en compte les processus de délais comme dans le modèle original complexifie la programmation sous R. Aussi, nous proposons une version avec quelques simplifications concernant la modélisation de ces délais pour XLAT et XINF.

|  |
| --- |
| **Définition de la structure du modèle avec des équations à différence.**  XVAC(day+1) = XVAC(day)-rocc  XLAT(day + 1) = XLAT(day) +rocc - rapp  XINF(day+1) = XINF(day) + rapp - rrem  XCTR(day+1) = XCTR(day) + rrem  **Définition des flux**  Vous devez ajouter des règles pour éviter d'avoir des variables d'état négatives. L'ordre de calcul est important.  # rocc: taux d'occupation : nb de sites Vacant=>Latent  rocc = min(cofr\*dmfr\*XINF(day), XVAC(day))  with  # rapp: taux d'apparition : nb de sites Latent=>Infectant  rapp= min(XLAT[day]\*1/nlpd, XLAT[day]+rocc)  # rrem: taux de "disparition" : nb de sites Infectant=>removed  Variables d'intérêt auxiliaires :  XTO1 = XLAT+XINF+XCTR  XSEV = XINF+XCTR |

Table 21. Proposition of parameter values for SEIR model and input variable for leaf rust on wheat.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value** | **Description** | **Unit** |
| nlpd | 8 | Duration pour la période de latence | day |
| nipd | 30 | Duration pour la période infectieuse | day |
| dmfr | 16 | Coefficient de multiplication | - |
| XVAC0 | 1e+10 | Nombre initial de sites vacants | sites |
| XLAT0 | 1 | Nombre initial de sites latents (détermine la première contamination) | sites |

### Questions

*Modèle simple dérivé de Zadoks.*

*Écrire sur papier la structure générale d'un algorithme permettant de simuler le modèle.*

*Ecrire un script R pour simuler ce modèle.*

*Encapsuler la procédure dans une fonction prenant comme argument*

*zakoks.simple.model = function (nlpd=4,nipd=1,dmfr=16,SITE0 = 5\*10^9,weather=NULL, sdate = 1, ldate = 140)*

*and returning a list with the values of the state variables each day as a data.frame and information on the parameters.*

et retournant une liste avec les valeurs des variables d'état chaque jour (comme un data.frame) et des informations sur les paramètres.

*return(list(sim=data.frame(day = sdate:ldate, XVAC = XVAC[sdate:ldate], XLAT = XLAT[sdate:ldate], XINF = XINF[sdate:ldate],XCTR = XCTR[sdate:ldate],XTO1=XTO1[sdate:ldate], XSEV= XSEV[sdate:ldate], severity=severity[sdate:ldate]), param=c(nlpd=nlpd,nipd=nipd,dmfr=dmfr,SITE0 = SITE0)))*

*Simuler le modèle avec les valeurs proposées pour les paramètres et vérifier si vous obtenez mêmes résultats que les autres participants.*

## Ajouter un effet climatique sur les processus.

L'ampleur des épidémies dépend de plusieurs conditions climatiques, en fonction du champignon considéré. Les conditions climatiques peuvent affecter des processus différents comme la dispersion, l'infection, la période de latence, la période infectieuse,... Dans le modèle de Zadoks (1971) et dans sa version simplifiée que vous avez encodée sous R, il n'y a aucun effet climatique. Nous vous proposons de rajouter un premier effet de la température, connu pour être important dans le cas de la rouille brune du blé.

### Questions

*Adding temperature effect on latency and infectious period*

*Nous allons considérer que la période de latence dépend de la température. A 10°C, la période de latence typique est d'environ 8 jours, de sorte que nous pouvons définir un nouveau paramètre nlpd\_T.*

*Le taux rapp peut alors être définie comme :*

*XLAT[day] \* TMEAN[day]/nlpd\_T*

*Pour les données météo, vous pouvez utiliser les données contenues dans le paquet ZeBook R en utilisant l'instruction*

*library(ZeBook)*

*sdate1 = 1 #as.numeric(format(strptime("1997-05-25", "%Y-%m-%d", tz=""), "%j"))*

*weather1=subset(weather\_FranceWest, WEYR==1997 & idsite==39)*

*Vous pouvez repartir de la fonction de zadoks.simple.model.r précédente (en prenant soin de l'enregistrer comme un nouveau fichier).*

*Comparer les résultats de différentes années sur le même site.*

## Plus de réalisme avec les files d'attente et la version d'origine du modèle de Zadoks (1971).

### Questions

*A partir de la fonction précédente, chercher une solution pour prendre en compte les délais comme une véritable file d'attente sous R. Vous devrez utiliser un vecteur ou une matrice au lieu d'un simple scalaire pour suivre le devenir de chaque cohorte entrant en phase de latence (et d'infection).  
Comparer le résultat avec le modèle zadoks.simple.model.*

## Comment ces différents modèles représentent le système réel ?

Précédemment, vous avez encodé 2 ou 3 versions du modèle proposé par Zadoks. Ces modèles sont construits à partir des connaissances épidémiologiques et cherche à reproduire en partie le comportement du pathosystème considéré. Dans le document de Zadoks, il y a également des analyses de sensibilité du modèle montrant qu'il semble reproduire des éléments connus du comportement du système dynamique considéré. Mais, aucune comparaison formelle avec des épidémies observées n'est faite. Si l'on compare les simulations avec des données existantes d'épidémie de rouille brune, on se rend compte des fortes divergences (travail de comparaison fait lors de la préparation de la formation). Sur ce point, nous vous rappelons l'importance du travail d'évaluation : voir le cours sur l'évaluation et les travaux pratiques sur les modèles statistiques. Néanmoins, nous proposons de mettre de côté ce point très important pour la suite, afin de nous focaliser sur des travaux de comparaison de scénarios.

# Utilisation du modèle de Zadoks pour analyser et comparer des scénarios.

## Objectif

Précédement, nous nous sommes concentrés sur la construction d'un modèle à partir de la littérature et quelques simulations simples. On va maintenant utiliser le modèle pour comparer des scénarios. Il s'agit d'analyse par exemple l'effet du changement climatique ou l'effet des pratiques.

Quelques éléments issus de références bibliographiques sont fournis, mais vous pouvez trouver plus d'éléments sur l'Internet, demander des informations aux intervenants ou faire des hypothèses.

Vous choisissez un sujet et vous pouvez travailler en binôme, on prendra le temps d'un échange collectif en fin de journée.

## Comment prendre en compte les pratiques dans le modèle de Zadoks ?

1. Utilisation de variétés plus ou moins résistantes.

Hypothèse que cela modifie des paramètres dans le modèle (durée de latence, d'infection ou taux de multiplication).

Faire des simulations avec différentes combinaisons de ces paramètres.

* nlpd=[8 à 11] (day) (ref : Herrera-Foessel *et al*, 2007)
* nipd=[1 à 30] (day)
* dmfr=[0 à 16] (day-1)

Sur une variable unique par simulation : AUDPC (*aera under disease progress curve*) sur 3 mois.

Astuce code R : pour créer une matrice de combinaison de ces paramètres.

expand.grid(nlpd=8:10, nipd=1:30, dmfr=0 :16)

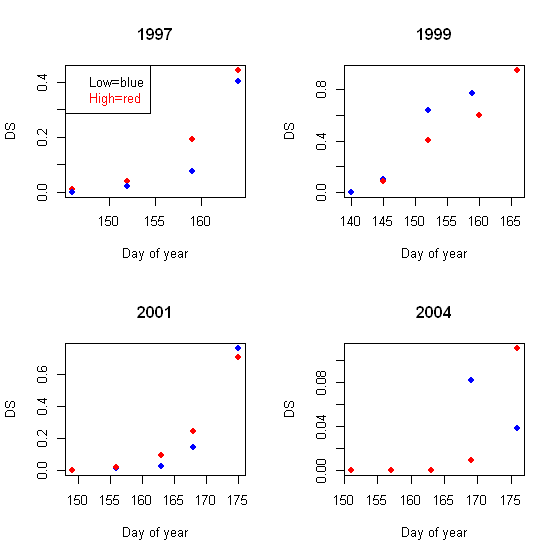
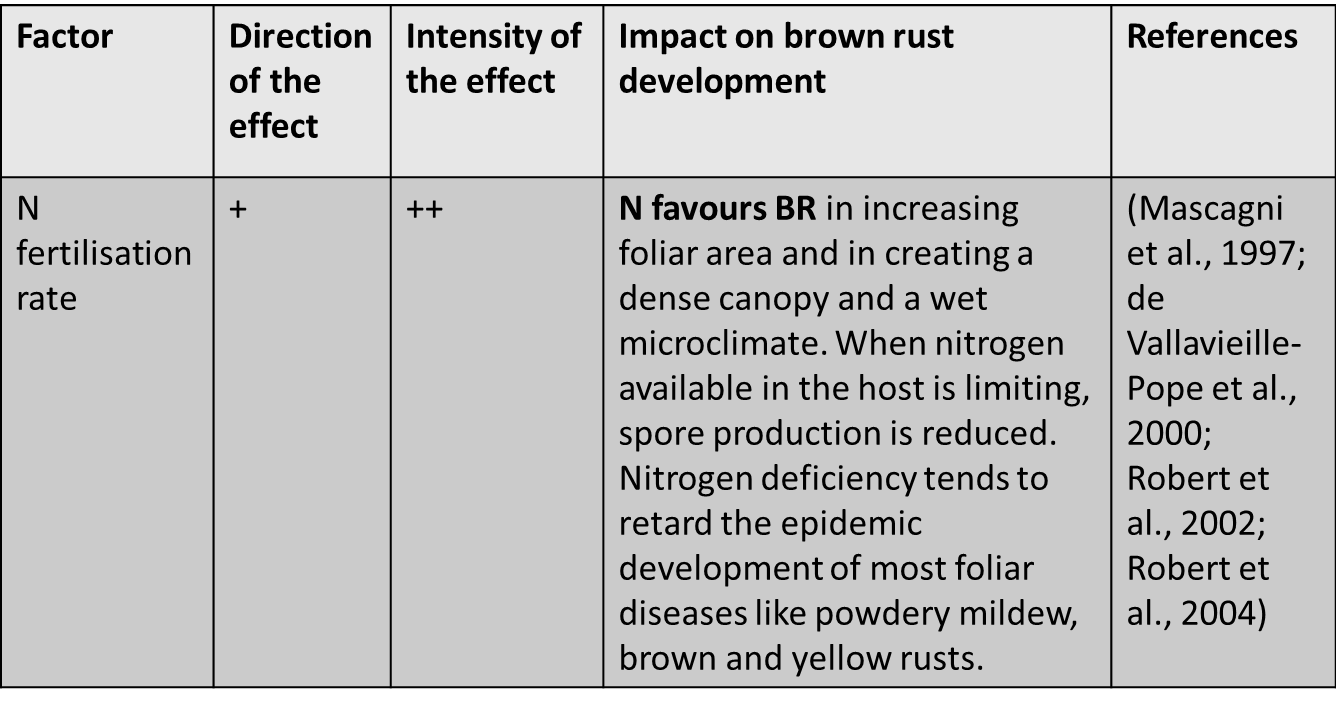
1. Effet de l'application de pesticides : chercher à optimiser la date d'application.

Hypothèse : effet protecteur des feuilles du fongicide : pas d’infection pendant duration\_fong jour après l’application

Astuce code R :

if (day>=day\_fong & day<day\_fong +duration\_fong) { rocc=0} else { rocc = rocc}

1. Effet de la fertilisation azotée.



Proposition :

* Dmfr high = 20( NEW)
* Dmfr low = 12 (NEW)

Astuce code R :

if (N\_fertilisation=="high") {dmfr= dmfrH} else {dmfr= dmfrL}

1. Effet de la date de semis.

Ceci nécessite l'ajout d'un modèle simple de croissance des plantes. La surface des feuilles est désormais dynamique. On propose d'utilise une fonction logistique pour cela.

## Effet du changement climatique sur les épidémies de rouille brune en France.

### Ressources disponibles

Nous avons des fichiers climatiques, avec le climat de référence ("actuel") et futur, pour 3 sites en France.

Vous devez utiliser une version du modèle qui représente l'effet de la température.

Il s'agit de projection de Météo-France de changement climatique (portail DRIAS, accès public).

* Climat de référence : 1950-2000.
* Climat futur 2001-2100 avec différents scénarios.

Pour 3 sites en France (zones de production de blé d'hiver) représentant des zones climatiques contrastées.

* près de Paris : INRA-Grignon
* près de Toulouse : INRA-Castanet-Tolosan
* près de Lyon : Arvalis-Satolas

Des scripts R pour commencer.

* zadoks.simpleweather.model.r
* read.weather.drias.r