**Supplementary Informations**

|  |  |  |  |
| --- | --- | --- | --- |
| Location | Position (latitude, longitude) | Altitude (m) | Climate (Joly et al., 2010) |
| Carquefou | 47.319461 -1.491685 | 7.4 | Oceanic |
| Etiolles | 48.667431 2.479903 | 84.2 | Degraded oceanic |
| Gardouch | 43.370952 1.673424 | 238.3 | Southwestern basin |
| La Tour-de-Salvagny | 45.797595 4.712050 | 286.2 | Degraded oceanic |
| Saint-Genès-Champanelle | 45.705787 2.971739 | 966.8 | Mountainous |
| Velaine-en-Haye | 48.704871 6.082540 | 334.1 | Semi-continental |

Table S1: Position, altitude and corresponding climate of the different sampling locations

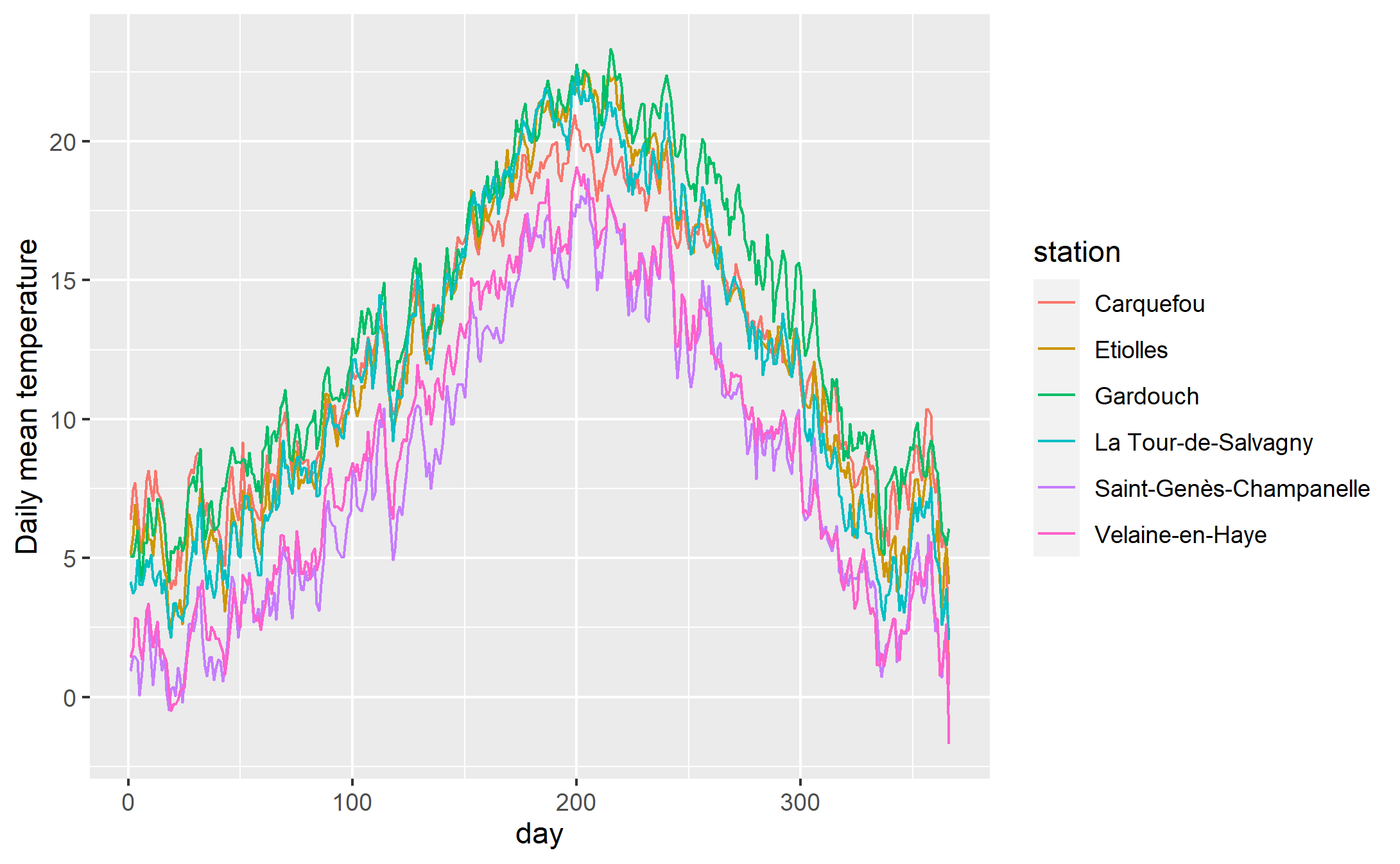
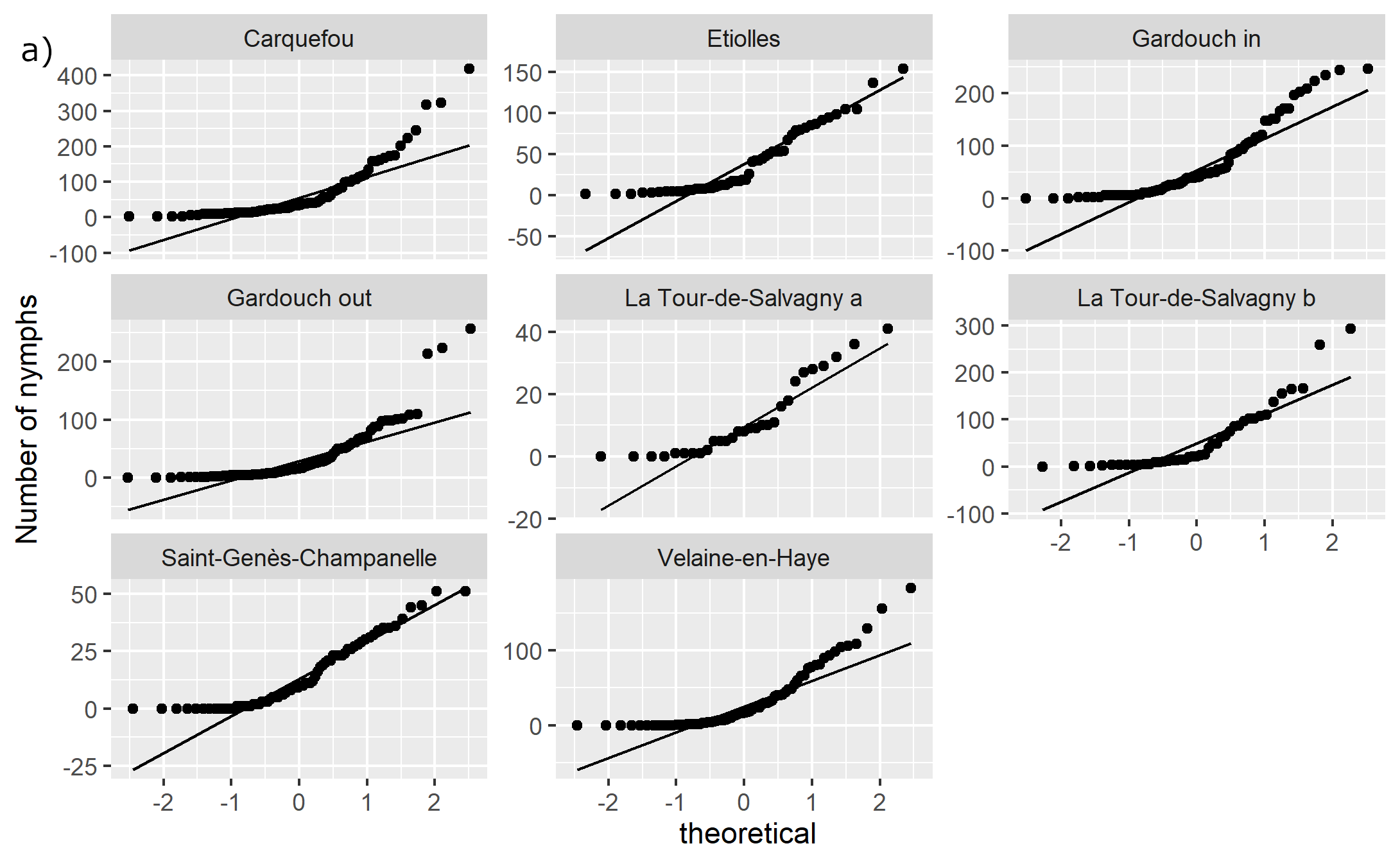


Figure S1: Daily average observed temperature data for the different locations.



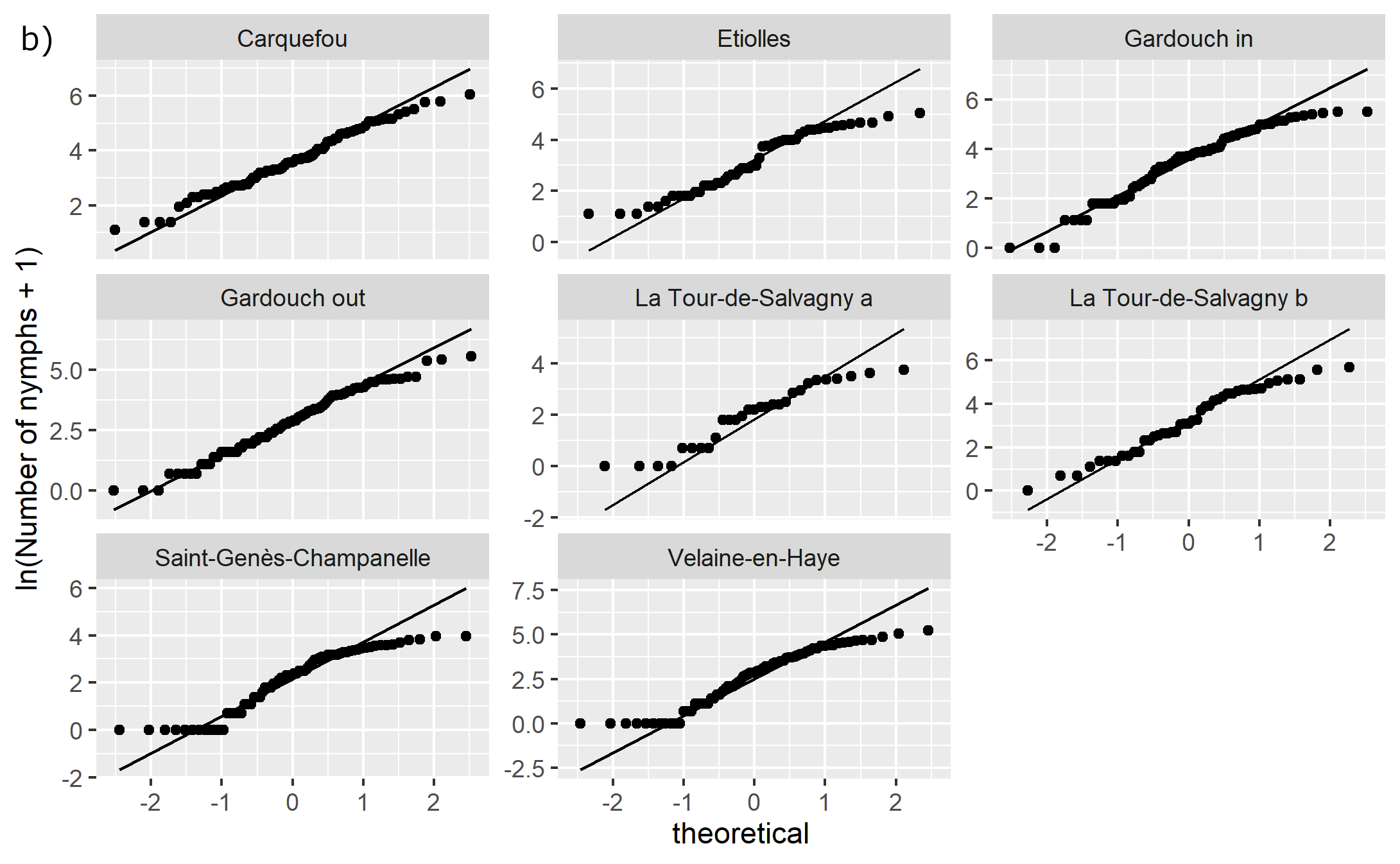


Figure S2: QQ plots of the data for the different sites without (a) or with (b) logarithmic transform.

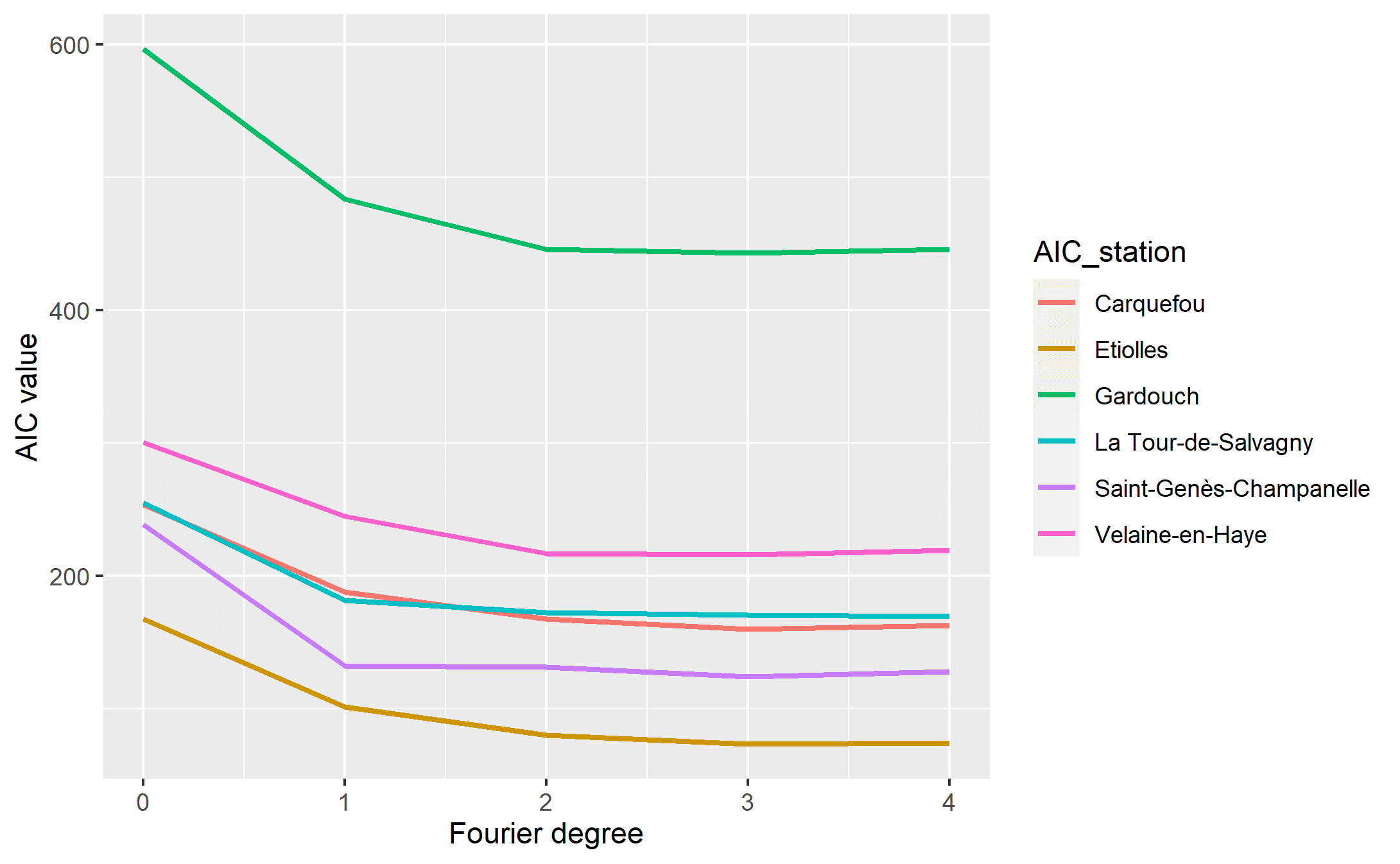


Figure S3: AIC values of harmonic regressions as a function of Fourier degree (parameter *K*) for the different sites. A unique AIC value was computed for La Tour-de-Salvagny and Gardouch sites.

R script

## ---- message = 'FALSE'------------------------------------------------------------------------------------------------

library(tidyverse)

library(ggplot2)

library(ggrepel)

## ----------------------------------------------------------------------------------------------------------------------

ticks <- as\_tibble(

read.csv("data/data\_for\_regression.csv",

stringsAsFactors = FALSE))

ticks <- ticks[-which(ticks$IDsite %in% c("9","10","11")),]

## ----------------------------------------------------------------------------------------------------------------------

ticks <- ticks %>%

mutate(date = as.Date(date, "%Y-%m-%d"),

day\_year = as.integer(format(date, "%j")),

day\_radian = day\_year / 365 \* 2 \* pi,

year = (as.integer(format(date, "%Y"))),

log\_nymph = log(NbNymphe + 1))

## ---- message = FALSE--------------------------------------------------------------------------------------------------

ls\_site\_station <- ticks %>%

group\_by(IDsite) %>%

summarise(

IDsite = unique(IDsite)) %>%

mutate(id = row\_number(),

IDstation = c(1,1,2,4,5,7,7,6),

IDstation\_1 = c("S1", "S1", "S2", "S4", "S5", "S6S7", "S6S7", "S8"),

site = c("La Tour-de-Salvagny a", "La Tour-de-Salvagny b", "Saint-Genès-Champanelle", "Etiolles", "Carquefou", "Gardouch in", "Gardouch out", "Velaine-en-Haye"),

station = c("La Tour-de-Salvagny", "La Tour-de-Salvagny", "Saint-Genès-Champanelle", "Etiolles", "Carquefou", "Gardouch", "Gardouch", "Velaine-en-Haye")) %>%

select(id, IDsite, IDstation, IDstation\_1, site, station)

ticks <- left\_join(ticks, ls\_site\_station)

## ----------------------------------------------------------------------------------------------------------------------

ticks %>%

group\_by(site, station) %>%

summarise(

n\_collectes = length(unique(date)),

min\_date = min(date),

max\_date = max(date)

)

## ----densities abundances----------------------------------------------------------------------------------------------

ggplot(ticks, aes(x = NbNymphe)) +

geom\_density() +

facet\_wrap(~site, scale = "free\_x")

ggsave("figures/abundance\_densities.png")

## ----densities log abundances------------------------------------------------------------------------------------------

ggplot(ticks, aes(x = log(NbNymphe+1))) +

geom\_density() +

facet\_wrap(~site)

ggsave("figures/abundance\_log\_densities.png")

## ----qqplot abundance untransformed------------------------------------------------------------------------------------

ggplot(ticks, aes(sample = NbNymphe)) +

stat\_qq() +

stat\_qq\_line() +

facet\_wrap(~site, scales="free\_y") +

ylab("Number of nymphs")

ggsave("figures/abundance\_qqplots.png")

## ----qqplot ln abundance-----------------------------------------------------------------------------------------------

ggplot(ticks, aes(sample = log(NbNymphe + 1))) +

stat\_qq() +

stat\_qq\_line() +

facet\_wrap(~site, scales="free\_y") +

ylab("ln(Number of nymphs + 1)")

ggsave("figures/abundance\_log\_qqplots.png")

## ----------------------------------------------------------------------------------------------------------------------

## Months in English

Sys.setlocale("LC\_ALL","English")

ticks %>%

mutate(day\_of\_year = paste0("2020-", format(date, "%m-%d"))) %>%

ggplot(., aes(x = as.Date(day\_of\_year), y = log\_nymph, col = factor(year))) +

geom\_point() +

facet\_wrap(~ site) +

scale\_x\_date(breaks = function(x) seq.Date(from = as.Date("2020-01-01"),

to = as.Date("2020-12-31"),

by = "1 month"),

date\_labels = "%b") +

theme(axis.text.x = element\_text(angle = 90, hjust = 1)) +

xlab("") +

ylab("ln(Number of nymphs + 1)") +

scale\_color\_discrete(name = "Year")

ggsave("figures/log\_nymph\_year\_site.png")

## ----------------------------------------------------------------------------------------------------------------------

ticks <- ticks %>% select(IDsite,date,NbNymphe,day\_year,day\_radian,year,log\_nymph,id,IDstation,IDstation\_1,station)

## ----------------------------------------------------------------------------------------------------------------------

## Without sites 1a et 1b (Salvagny a et Salvagny b)

ls\_stations <- unique(as.character(ticks$IDstation))

# Same colour for all sites throughout

colrs <- c(1,2,3,4,6,7)

# Test of the models and AIC computation

AIC\_table <- data.frame(

degre\_Fourier=rep(c(0:4),6),

AIC\_station = rep(unique(ticks$station),each=5),

mdL = NA)

## For Salvagny a et Salvagny b one model with distinct intercepts

mdL0\_1 <- lm(log\_nymph ~ 1 + factor(IDsite),

data = ticks, subset = IDsite %in% c("1a", "1b"))

mdL1\_1 <- lm(log\_nymph ~ factor(IDsite) +

cos(day\_radian) + sin(day\_radian),

data = ticks, subset = IDsite %in% c("1a", "1b"))

mdL2\_1 <- lm(log\_nymph ~ factor(IDsite) +

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian),

data = ticks, subset = IDsite %in% c("1a", "1b"))

mdL3\_1 <- lm(log\_nymph ~ factor(IDsite) +

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian) +

cos(3 \* day\_radian) + sin(3 \* day\_radian),

data = ticks, subset = IDsite %in% c("1a", "1b"))

mdL4\_1 <- lm(log\_nymph ~ factor(IDsite) +

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian) +

cos(3 \* day\_radian) + sin(3 \* day\_radian) +

cos(4 \* day\_radian) + sin(4 \* day\_radian),

data = ticks, subset = IDsite %in% c("1a", "1b"))

AIC\_table$mdL[1] <- AIC(mdL0\_1)

AIC\_table$mdL[2] <- AIC(mdL1\_1)

AIC\_table$mdL[3] <- AIC(mdL2\_1)

AIC\_table$mdL[4] <- AIC(mdL3\_1)

AIC\_table$mdL[5] <- AIC(mdL4\_1)

## Same for Gardouch

mdL0\_5 <- lm(log\_nymph ~ 1 + factor(IDsite),

data = ticks, subset = IDsite %in% c("6", "7"))

mdL1\_5 <- lm(log\_nymph ~ factor(IDsite) +

cos(day\_radian) + sin(day\_radian),

data = ticks, subset = IDsite %in% c("6", "7"))

mdL2\_5 <- lm(log\_nymph ~ factor(IDsite) +

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian),

data = ticks, subset = IDsite %in% c("6", "7"))

mdL3\_5 <- lm(log\_nymph ~ factor(IDsite) +

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian) +

cos(3 \* day\_radian) + sin(3 \* day\_radian),

data = ticks, subset = IDsite %in% c("6", "7"))

mdL4\_5 <- lm(log\_nymph ~ factor(IDsite) +

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian) +

cos(3 \* day\_radian) + sin(3 \* day\_radian) +

cos(4 \* day\_radian) + sin(4 \* day\_radian),

data = ticks, subset = IDsite %in% c("6", "7"))

AIC\_table$mdL[21] <- AIC(mdL0\_5)

AIC\_table$mdL[22] <- AIC(mdL1\_5)

AIC\_table$mdL[23] <- AIC(mdL2\_5)

AIC\_table$mdL[24] <- AIC(mdL3\_5)

AIC\_table$mdL[25] <- AIC(mdL4\_5)

pred <- expand.grid(

date = seq(as.Date("2014-01-01"), as.Date("2019-12-31"), by = "1 day"),

id = 1:8

)

pred <- full\_join(pred, ls\_site\_station)

pred <- pred %>%

mutate(day = as.numeric(format(date, "%j")),

day\_radian = day / 365 \* 2 \* pi)

pred$ln\_nNymph <- rep(NA)

pred$ln\_nNymph <- rep(NA)

## Models for all sites except Salvagny and Gardouch

for(i in c(2:4,6)){

i\_station <- ls\_stations[i]

mdL0 <- lm(log\_nymph ~ 1,

data = ticks, subset = IDsite == i\_station)

mdL1 <- lm(log\_nymph ~ cos(day\_radian) + sin(day\_radian),

data = ticks, subset = IDsite == i\_station)

mdL2 <- lm(log\_nymph ~ cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian),

data = ticks, subset = IDsite == i\_station)

mdL3 <- lm(log\_nymph ~ cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian) +

cos(3 \* day\_radian) + sin(3 \* day\_radian),

data = ticks, subset = IDsite == i\_station)

mdL4 <- lm(log\_nymph ~ cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian) +

cos(3 \* day\_radian) + sin(3 \* day\_radian) +

cos(4 \* day\_radian) + sin(4 \* day\_radian),

data = ticks, subset = IDsite == i\_station)

AIC\_table$mdL[(i - 1)\*5+1] <- AIC(mdL0)

AIC\_table$mdL[(i - 1)\*5+2] <- AIC(mdL1)

AIC\_table$mdL[(i - 1)\*5+3] <- AIC(mdL2)

AIC\_table$mdL[(i - 1)\*5+4] <- AIC(mdL3)

AIC\_table$mdL[(i - 1)\*5+5] <- AIC(mdL4)

}

## ----------------------------------------------------------------------------------------------------------------------

AIC\_table %>%

ggplot(., aes(x = degre\_Fourier, y = mdL, colour=AIC\_station)) +

geom\_line(size = 1) +

ylab("AIC value") +

xlab("Fourier degree") +

scale\_color\_manual(values=c("#F8766D", "#CD9600", "#00BE67", "#00BFC4", "#C77CFF", "#FF61CC"))

ggsave("figures/AIC\_degree.png")

## ----------------------------------------------------------------------------------------------------------------------

mdL2\_1 <- lm(log\_nymph ~

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian),

data = ticks, subset = IDsite %in% c("1a", "1b"))

mdL2\_2 <- lm(log\_nymph ~ factor(IDsite) +

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian),

data = ticks, subset = IDsite %in% c("1a", "1b"))

mdL2\_3a <- lm(log\_nymph ~

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian),

data = ticks, subset = IDsite == "1a")

mdL2\_3b <- lm(log\_nymph ~

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian),

data = ticks, subset = IDsite == "1b")

pred\_1ab <- pred[pred$IDsite %in% c("1a", "1b"),]

pred\_1ab$mdL2\_1 <- predict(mdL2\_1, pred\_1ab)

pred\_1ab$mdL2\_2 <- predict(mdL2\_2, pred\_1ab)

pred\_1ab$mdL2\_3 <- rep(NA)

pred\_1ab$mdL2\_3[pred\_1ab$IDsite == "1a"] <- predict(mdL2\_3a, pred\_1ab[pred\_1ab$IDsite == "1a",])

pred\_1ab$mdL2\_3[pred\_1ab$IDsite == "1b"] <- predict(mdL2\_3b, pred\_1ab[pred\_1ab$IDsite == "1b",])

## ----------------------------------------------------------------------------------------------------------------------

mdL3\_1 <- lm(log\_nymph ~

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian),

data = ticks, subset = IDsite %in% c("6", "7"))

mdL3\_2 <- lm(log\_nymph ~ factor(IDsite) +

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian),

data = ticks, subset = IDsite %in% c("6", "7"))

AIC(mdL3\_1)

AIC(mdL3\_2)

## ----------------------------------------------------------------------------------------------------------------------

rm(list = ls()[grep("mdL", ls())])

rm(list = ls()[grep("pred", ls())])

## ---- message = FALSE--------------------------------------------------------------------------------------------------

pred <- expand.grid(

date = seq(as.Date("2014-01-01"), as.Date("2021-12-31"), by = "1 day"),

id = 1:8

)

pred <- full\_join(pred, ls\_site\_station)

pred <- pred %>%

mutate(day = as.numeric(format(date, "%j")),

day\_radian = day / 365 \* 2 \* pi)

pred$ln\_nNymph <- rep(NA)

## ----------------------------------------------------------------------------------------------------------------------

ls\_sites <- unique(ticks$IDsite)

## Models for La Tour de Salvagny

mdL\_S1 <- lm(log\_nymph ~ factor(IDsite) +

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian),

data = ticks, subset = IDsite %in% c("1a", "1b"))

qqnorm(resid(mdL\_S1), main = paste("1a", "1b"))

abline(a = 0, b = 1)

pred$ln\_nNymph[pred$IDsite %in% c("1a", "1b")] <- predict(mdL\_S1, pred[pred$IDsite %in% c("1a", "1b"),])

## Models for Gardouch

mdL\_S67 <- lm(log\_nymph ~ factor(IDsite) +

cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian),

data = ticks, subset = IDsite %in% c("6", "7"))

qqnorm(resid(mdL\_S67), main = paste0("6 - 7"))

abline(a = 0, b = 1)

pred$ln\_nNymph[pred$IDsite %in% c("6", "7")] <- predict(mdL\_S67, pred[pred$IDsite %in% c("6", "7"),])

## Models for sites except Gardouch and La Tour de Salvagny

for(i in c(3:5, 8)){

i\_site <- ls\_sites[i]

nm <- paste0("mdL\_S", i\_site)

mdL <- lm(log\_nymph ~ cos(day\_radian) + sin(day\_radian) +

cos(2 \* day\_radian) + sin(2 \* day\_radian),

data = ticks, subset = IDsite == i\_site)

assign(nm, mdL)

pred$ln\_nNymph[pred$IDsite == i\_site] <- predict(mdL, pred[pred$IDsite == i\_site,])

qqnorm(resid(mdL), main = ls\_sites[i])

abline(a = 0, b = 1)

}

pred$ln\_nNymph[pred$ln\_nNymph < 0] <- 0

## ----------------------------------------------------------------------------------------------------------------------

curvs <- pred %>%

filter(date > "2018-12-31" & date < "2020-01-01") %>%

group\_by(IDsite) %>%

mutate(dif\_1 = ln\_nNymph - lag(ln\_nNymph, default = NA),

dif\_2 = dif\_1 - lag(dif\_1, default = NA ),

dif\_1\_sign = sign(dif\_1) - lag(sign(dif\_1), default = NA ),

dif\_2\_sign = sign(dif\_2) - lag(sign(dif\_2), default = NA )

)

dat\_pic <- curvs %>%

filter(dif\_1\_sign == -2) %>%

select( date, id, IDsite, IDstation, site, station,day, ln\_nNymph) %>%

mutate(type\_event = "max")

dat\_min <- curvs %>%

filter(dif\_1\_sign == 2) %>%

select( date, id, IDsite, IDstation, site, station,day, ln\_nNymph) %>%

mutate(type\_event = "min")

dat\_inflx <- curvs %>%

filter(dif\_2\_sign == -2) %>%

select( date, id, IDsite, IDstation, site, station,day, ln\_nNymph) %>%

mutate(type\_event = "inflx")

curvs\_events <- bind\_rows(dat\_pic, dat\_min, dat\_inflx) %>%

arrange(IDsite)

rm(list = c("dat\_pic", "dat\_min", "dat\_inflx"))

## ----------------------------------------------------------------------------------------------------------------------

events <- curvs\_events %>%

filter(type\_event == "max" & date < "2019-10-01")

events <- bind\_rows(

events,

curvs\_events %>%

filter(site %in% c("Lyon a", "Lyon b", "Sénart", "Nancy") &

type\_event == "inflx" & date > "2019-09-01" & date < "2019-11-01"),

curvs\_events %>%

filter(site %in% c("Carquefou", "Toulouse int", "Toulouse ext") &

type\_event == "min" & date > "2019-08-01" & date < "2019-11-01")

) %>%

arrange(IDsite)

## ----------------------------------------------------------------------------------------------------------------------

Sys.setlocale("LC\_ALL","English")

plot\_labs <- curvs %>%

group\_by(site) %>%

summarise(date = max(date),

day = max(day),

ln\_nNymph = last(ln\_nNymph))

ggplot(curvs, aes(x = date, y = ln\_nNymph, colour = site)) +

geom\_line(size = 1) +

geom\_text\_repel(data = plot\_labs,

aes(x = date, y = ln\_nNymph, label = site),

nudge\_x = 30,

segment.linetype = 5) +

geom\_point(data = events[events$type\_event == "max",],

aes(x = date, y = ln\_nNymph, colour=site),

size = 3) +

scale\_x\_date(breaks = function(x) seq.Date(from = as.Date("2019-01-01"),

to = as.Date("2019-12-31"),

by = "1 month"),

date\_labels = "%b") +

ylab("ln(Number of nymphs + 1)") +

xlab("") +

theme(axis.text.x = element\_text(angle = 90, hjust = 1)) +

theme(legend.position = "none")

# scale\_color\_manual(values=c("#F8766D", "#CD9600", "#7CAE00","#00BE67", "#00BFC4", "#00A9FF", "#C77CFF", "#FF61CC"))

ggsave("figures/superimposed\_abundances.png")

## ----------------------------------------------------------------------------------------------------------------------

Sys.setlocale("LC\_ALL","English")

plot\_labs <- curvs %>%

group\_by(site) %>%

summarise(date = max(date),

day = max(day),

ln\_nNymph = last(ln\_nNymph))

ggplot(subset(curvs, IDsite %in% c("5", "6", "7")),aes(x = date, y = ln\_nNymph, colour = site)) +

geom\_line(size = 1) +

geom\_text\_repel(subset(plot\_labs,site %in% c("Carquefou","Gardouch in","Gardouch out")),mapping=aes(x = date, y = ln\_nNymph, label = site),

nudge\_x = 30, segment.linetype = 5) +

geom\_point(subset(events,events$type\_event == "max" & events$IDsite %in% c("5", "6", "7")), mapping=aes(x = date, y = ln\_nNymph,colour = site),size = 3) +

scale\_x\_date(breaks = function(x) seq.Date(from = as.Date("2019-01-01"),

to = as.Date("2019-12-31"),

by = "1 month"),

date\_labels = "%b") +

ylab("ln(Number of nymphs + 1)") +

xlab("") +

theme(axis.text.x = element\_text(angle = 90, hjust = 1)) +

theme(legend.position = "none")+

scale\_color\_manual(values=c("#F8766D", "#7CAE00", "#00BE67" ))

## ----------------------------------------------------------------------------------------------------------------------

Sys.setlocale("LC\_ALL","English")

plot\_labs <- curvs %>%

group\_by(site) %>%

summarise(date = max(date),

day = max(day),

ln\_nNymph = last(ln\_nNymph))

ggplot(subset(curvs, IDsite %in% c("1a", "1b", "4", "8")),aes(x = date, y = ln\_nNymph, colour = site)) +

geom\_line(size = 1) +

geom\_text\_repel(subset(plot\_labs,site %in% c("La Tour-de-Salvagny a","La Tour-de-Salvagny b","Etiolles","Velaine-en-Haye")),mapping=aes(x = date, y = ln\_nNymph, label = site),

nudge\_x = 30, segment.linetype = 5) +

geom\_point(subset(events,events$type\_event == "max" & events$IDsite %in% c("1a", "1b", "4", "8")), mapping=aes(x = date, y = ln\_nNymph,colour = site),size = 3) +

scale\_x\_date(breaks = function(x) seq.Date(from = as.Date("2019-01-01"),

to = as.Date("2019-12-31"),

by = "1 month"),

date\_labels = "%b") +

ylab("ln(Number of nymphs + 1)") +

xlab("") +

theme(axis.text.x = element\_text(angle = 90, hjust = 1)) +

theme(legend.position = "none")+

scale\_color\_manual(values=c("#CD9600", "#00BFC4","#00A9FF", "#FF61CC"))

## ----------------------------------------------------------------------------------------------------------------------

Sys.setlocale("LC\_ALL","English")

plot\_labs <- curvs %>%

group\_by(site) %>%

summarise(date = max(date),

day = max(day),

ln\_nNymph = last(ln\_nNymph))

ggplot(subset(curvs, IDsite=="2"),aes(x = date, y = ln\_nNymph, colour = site)) +

geom\_line(size = 1) +

geom\_text\_repel(subset(plot\_labs,site == "Saint-Genès-Champanelle"),mapping=aes(x = date, y = ln\_nNymph, label = site),

nudge\_x = 30, segment.linetype = 5) +

geom\_point(subset(events,events$type\_event == "max" & events$IDsite == "2"), mapping=aes(x = date, y = ln\_nNymph,colour = site),size = 3) +

scale\_x\_date(breaks = function(x) seq.Date(from = as.Date("2019-01-01"),

to = as.Date("2019-12-31"),

by = "1 month"),

date\_labels = "%b") +

ylab("ln(Number of nymphs + 1)") +

xlab("") +

theme(axis.text.x = element\_text(angle = 90, hjust = 1)) +

theme(legend.position = "none")+

scale\_color\_manual(values="#C77CFF")

## ----------------------------------------------------------------------------------------------------------------------

pred\_tick <- pred %>%

select(date, id, IDsite, IDstation, site, station, day, day\_radian, ln\_nNymph)

pred\_tick <- left\_join(pred\_tick,

ticks %>%

select(IDsite, date, log\_nymph))

## ----------------------------------------------------------------------------------------------------------------------

ggplot(pred\_tick, aes(x = date, y = ln\_nNymph)) +

geom\_line() +

geom\_point(aes(y = log\_nymph, col = "red")) +

facet\_wrap(~ site) +

theme(legend.position = "none") +

xlab("Time") +

ylab("ln(number of nymphs + 1)")

ggsave("figures/ln\_nymphs\_observed\_predicted.png")

## ----------------------------------------------------------------------------------------------------------------------

meteo <- as\_tibble(

read.csv("data/imputedMeteo\_New.csv"))

## ----------------------------------------------------------------------------------------------------------------------

meteo <- meteo %>%

mutate(date = as.Date(Date))

## ----------------------------------------------------------------------------------------------------------------------

meteo <- meteo %>%

mutate(year = format(date, "%Y"),

day = as.numeric(format(date, "%j")),

day\_radian = day / 365 \* 2 \* pi)

meteo <- meteo[-which(meteo$Site %in% c("S9")),]

## ---- message = FALSE--------------------------------------------------------------------------------------------------

ls\_site\_station\_meteo <- meteo%>%

group\_by(Site) %>%

summarise(

Site = unique(Site)) %>%

mutate(station = c("La Tour-de-Salvagny", "Saint-Genès-Champanelle", "Etiolles", "Carquefou", "Gardouch", "Velaine-en-Haye")) %>%

select(Site, station)

meteo <- left\_join(meteo, ls\_site\_station\_meteo)

## ----------------------------------------------------------------------------------------------------------------------

ggplot(meteo, aes(x = date, y = TM\_m1)) +

geom\_point() +

facet\_wrap(~ station)

## ----------------------------------------------------------------------------------------------------------------------

meteo\_day <- meteo %>%

group\_by(Site, day) %>%

summarise(station = unique(station),

date = max(date),

temp\_mean = mean(TM\_m1),

U\_min = mean(UN\_m1),

U\_mean = mean(UM\_m1))

meteo\_day <- meteo\_day %>%

filter(!is.na(temp\_mean)) %>%

mutate(

temp\_tmp = ifelse(temp\_mean >= 5,temp\_mean, 0),

temp\_cumsum = cumsum(temp\_mean),

temp\_cumsum1 = cumsum(temp\_tmp),

U\_cumsum = cumsum(U\_mean))

## ----------------------------------------------------------------------------------------------------------------------

ggplot(data = meteo\_day, aes(x = day, y = temp\_mean, col = station)) +

geom\_line()+

scale\_color\_manual(values=c("#F8766D", "#CD9600", "#00BE67", "#00BFC4", "#C77CFF", "#FF61CC"))+

ylab("Daily mean temperature")

ggsave("figures/temperature\_data.png")

## ----------------------------------------------------------------------------------------------------------------------

tick\_peak <- events %>%

filter(type\_event == "max") %>%

ungroup() %>%

select(day, station) %>%

group\_by(station) %>%

summarise(day = unique(day), station = unique(station))

meteo\_day$New\_Date=as.Date(meteo\_day$day, origin = "2021-01-01")

tick\_peak <- left\_join(

tick\_peak,

meteo\_day %>%

select(day, New\_Date, station, Site, temp\_cumsum)

)

## ----------------------------------------------------------------------------------------------------------------------

meteo\_day %>%

filter(day < 220 & temp\_cumsum < 2000) %>%

ggplot(., aes(x = New\_Date, y = temp\_cumsum, col=station)) +

geom\_line(size = 1) +

scale\_color\_manual(values=c("#F8766D", "#CD9600", "#00BE67", "#00BFC4", "#C77CFF", "#FF61CC")) +

geom\_point(tick\_peak,

mapping = aes(x = New\_Date, y = temp\_cumsum), size=3) +

geom\_text\_repel(data = tick\_peak,

aes(x = New\_Date, y = temp\_cumsum, label = station),

force = 50,

segment.linetype = 5) +

ylim(0, 2000) +

xlab("") +

ylab("Cumulative temperature (°C.d)") +

theme(legend.position = "none")

ggsave("figures/cumulative\_temp\_peak.png")

## ----------------------------------------------------------------------------------------------------------------------

date\_peak <- events %>%

filter(type\_event == "max") %>%

ungroup() %>%

select(day, station) %>%

group\_by(station)

# %>%

# summarise(day, station)

date\_peak <- left\_join(

date\_peak,

meteo\_day %>%

select(day, New\_Date, station, Site, temp\_mean, temp\_cumsum)

)

## ----------------------------------------------------------------------------------------------------------------------

hiv <- tibble(

site = unique(ls\_site\_station$site),

ln\_nNymph\_MoyHiv = rep(NA),

ln\_nNymph\_Max = rep(NA),

ln\_nNymph\_Mean = rep(NA),

ln\_nNymph\_MoyHiv\_sur\_Max = rep(NA),

ln\_nNymph\_MoyHiv\_sur\_Mean = rep(NA),

teta\_MoyHiv = rep(NA))

for(i in 1:nrow(ls\_site\_station)){

## Moyenne du ln(nombre de nymphes) pour janvier 2019 pour IDsite i

hiv$ln\_nNymph\_MoyHiv[i] <- mean(pred$ln\_nNymph[pred$id == i & pred$date > "2020-12-31" & pred$date < "2021-02-01"])

## Max du ln(nombre de nymphes) pour toute l'année pour IDsite i

hiv$ln\_nNymph\_Max[i] <- max(pred$ln\_nNymph[pred$id == i])

## Moyenne du ln(nombre de nymphes) pour toute l'année pour IDsite i

hiv$ln\_nNymph\_Mean[i] <- mean(pred$ln\_nNymph[pred$id == i])

## Moyenne du ln(nombre de nymphes) pour janvier divisé par le max pour toute l'année pour IDsite i

hiv$ln\_nNymph\_MoyHiv\_sur\_Max[i] <- hiv$ln\_nNymph\_MoyHiv[i]/hiv$ln\_nNymph\_Max[i]

## Moy / max / Moy ?

hiv$ln\_nNymph\_MoyHiv\_sur\_Mean[i] <- hiv$ln\_nNymph\_MoyHiv[i]/hiv$ln\_nNymph\_Mean[i]

hiv$teta\_MoyHiv[i] <- mean(meteo\_day$temp\_mean[meteo\_day$station == ls\_site\_station$station[i] & meteo\_day$date > "2020-12-31" & meteo\_day$date < "2021-02-01"])

}

## ----------------------------------------------------------------------------------------------------------------------

ggplot(hiv, aes(x = teta\_MoyHiv, y = ln\_nNymph\_MoyHiv\_sur\_Max, col = site)) +

xlab("Average temperature (°C)") +

ylab("Relative nymph density") +

geom\_point(size=3)

ggsave("figures/winter\_density\_temperature.png")