TP 1: INITIATIONS AUX OUTILS D'ANALYSE D'UNE SERIE TEMPORELLE

IS 2A5

Introduction aux outils d'analyses

1. Chargement du jeu de données et création de l'objet TS

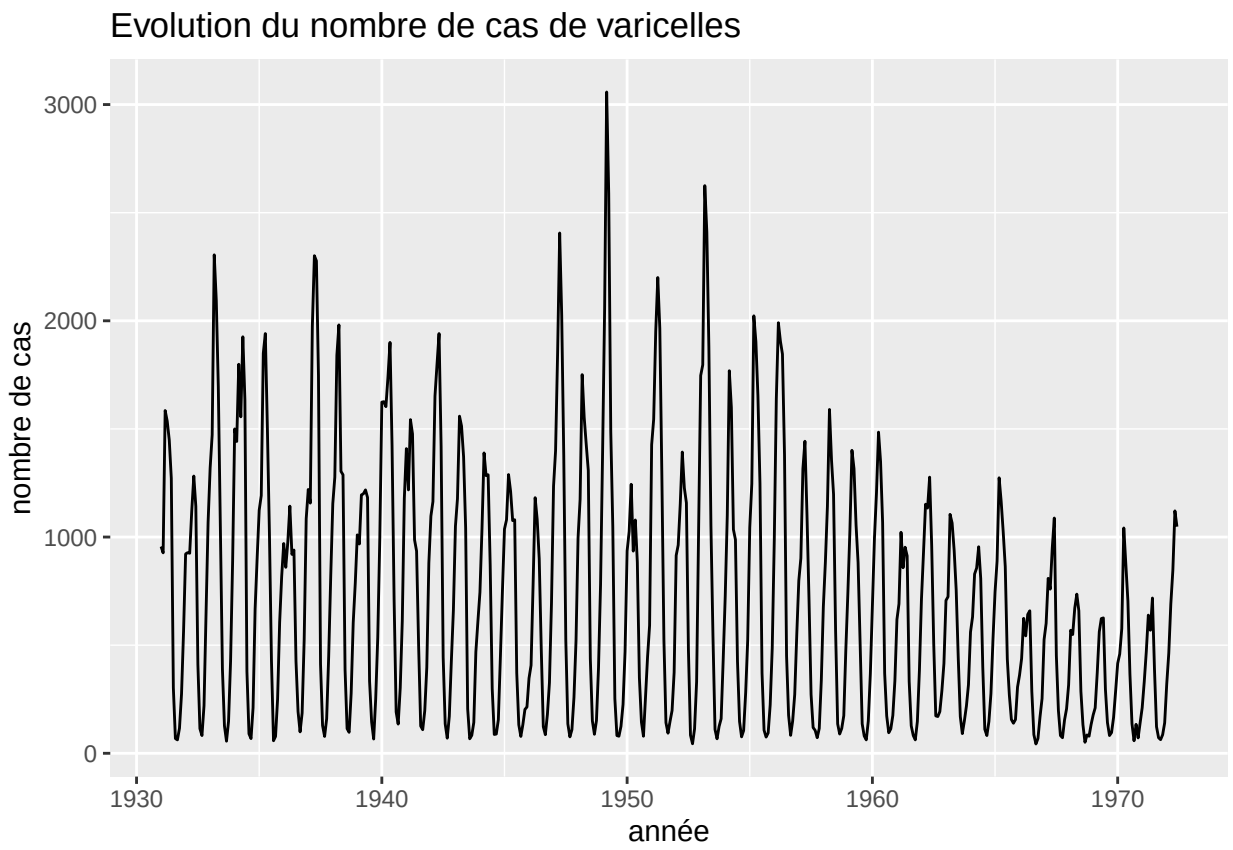
```
data <-scan("../TP1 - Intro et Lissages/varicelle.dat",skip=1)
ts_varicelle <- ts(data,frequency=12,start=c(1931,1),end=c(1972,6))
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1931	956	927	1585	1536	1448	1272	303	68	62	116	275	565
1932	922	928	925	1121	1282	1142	411	114	82	220	646	1069
1933	1320	1473	2305	2094	1694	1043	390	127	56	148	426	890
1934	1500	1442	1799	1556	1926	1635	379	90	68	210	667	905
1935	1124	1192	1850	1941	1505	1016	429	58	78	251	605	817
1936	970	860	977	1143	920	940	426	193	99	186	525	1085
1937	1220	1157	1974	2301	2277	1746	413	129	78	160	448	820
1938	1154	1277	1841	1981	1304	1288	387	114	97	278	604	787
1939	1010	968	1195	1200	1218	1183	334	145	66	252	536	996
1940	1624	1626	1603	1740	1900	1424	711	191	135	302	612	1178
1941	1409	1218	1543	1477	987	935	495	126	109	197	397	880
1942	1097	1164	1652	1800	1941	1419	444	136	70	171	424	660
1943	1050	1177	1559	1513	1371	1042	205	67	83	143	469	611
1944	745	1039	1389	1284	1288	871	299	87	89	155	446	749
1945	1037	1080	1289	1211	1076	1080	372	132	78	133	203	214
1946	347	407	780	1182	1082	899	479	123	86	180	326	695
1947	1235	1399	1854	2406	2026	1378	522	136	76	109	259	521
1948	996	1174	1751	1554	1428	1308	438	150	88	151	395	781
1949	1389	2059	3058	2589	1488	1048	253	82	79	125	226	470
1950	936	1026	1244	935	1079	884	349	144	79	260	445	592
1951	1427	1545	1951	2200	1964	1284	523	142	93	148	198	374
1952	915	963	1154	1393	1227	1158	478	84	44	113	331	1052
1953	1747	1796	2625	2411	1877	1052	543	110	67	124	160	430
1954	726	1101	1769	1599	1035	988	424	147	76	105	281	524
1955	1044	1247	2023	1903	1653	1247	372	107	75	94	224	487
1956	989	1639	1991	1905	1846	1381	451	176	83	150	272	550
1957	798	902	1316	1443	1102	705	272	119	106	72	115	337
1958	677	885	1142	1590	1355	1198	565	136	89	115	174	477
1959	741	1034	1401	1316	1056	882	506	136	80	62	149	368
1960	683	993	1205	1485	1349	1067	369	173	95	113	175	335
1961	619	691	1022	858	953	913	332	127	82	62	147	384
1962	711	928	1152	1134	1277	961	509	173	170	193	290	415
1963	707	724	1105	1065	938	755	442	170	91	150	219	317
1964	561	631	829	857	955	808	398	111	82	147	276	528

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1965	746	889	1274	1164	1024	863	436	270	156	139	156	306
1966	362	438	624	543	642	659	286	86	43	68	168	253
1967	526	601	809	759	950	1088	452	198	82	72	154	206
1968	316	569	549	671	736	659	287	132	51	85	79	133
1969	177	210	372	562	623	626	296	142	82	96	166	288
1970	416	459	576	1042	873	704	366	137	58	134	71	142
1971	211	331	471	639	569	718	391	123	72	63	86	141
1972	320	463	690	847	1121	1048	NA	NA	NA	NA	NA	NA

Représentation graphique

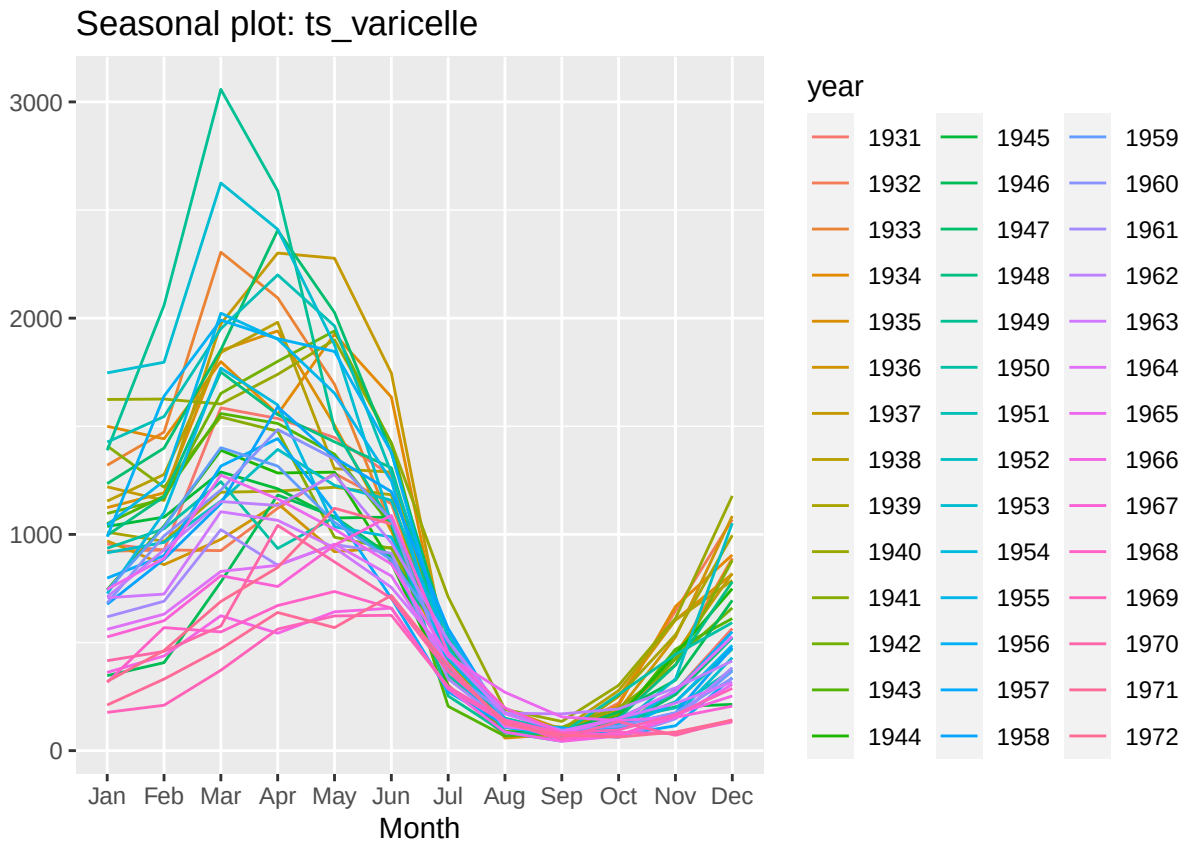
```
autoplot(ts_varicelle) +
  ggtitle("Evolution du nombre de cas de varicelles") +
  xlab("année") +
  ylab("nombre de cas")
```



La série présente des fluctuations pouvant supposer une saisonnalité, de plus on observe une tendance baissière prononcée à partir de 1950.

2. Evolution mensuelle du nombre de cas

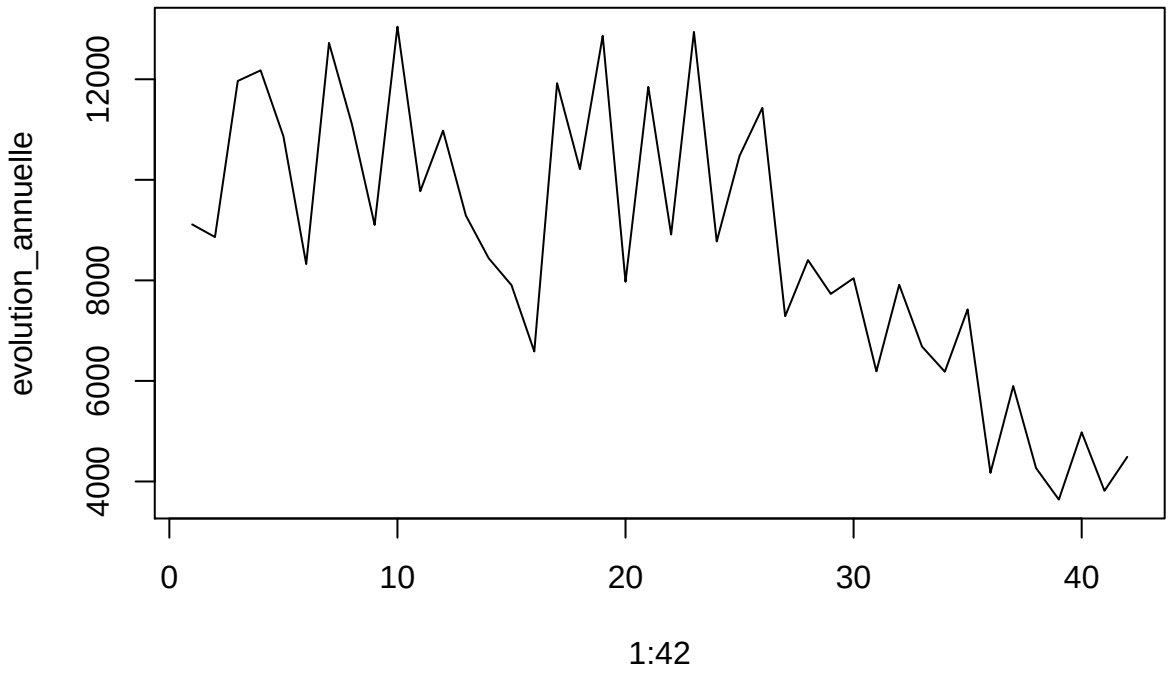
```
ggseasonplot(ts_varicelle)
```



La superposition des courbes mensuelles permet de mettre en évidence le comportement saisonnier des occurence de varicelle.

3. Evolution annuelle

```
evolution_annuelle=c();
for (i in 1931:1972)
{
    x<-window(ts_varicelle,start=c(i,1),end=c(i,12),extend=TRUE)
    evolution_annuelle=c(evolution_annuelle,sum(x,na.rm=TRUE))
}
plot(1:42,evolution_annuelle,type='l')
```



L'évolution annuelle confirme le comportement globalement décroissant du nombre de cas de varicelle dans le temps.

5. Les représentations graphiques des données mensuelles confirme l'hypothèse de saisonnalité, tandis que la représentation annuelle confirme l'existence d'une tendance dans notre jeu de données.

6. Données moyennes par an

```
year <-  c(rep(1931:1971,each=12),rep(1972,6))

stats <-  as.data.frame(cbind(year,data))  %>%
          group_by(year) %>%
          summarise(min = min(data), max= max(data),sd = sd(data),mean = mean(data),sum = sum(data))
```

year	min	max	sd	mean	sum
1931	62	1585	599.0517	759.4167	9113
1932	82	1282	428.9260	738.5000	8862
1933	56	2305	786.3739	997.1667	11966
1934	68	1926	705.6506	1014.7500	12177
1935	58	1941	648.2384	905.5000	10866
1936	99	1143	382.9750	693.6667	8324
1937	78	2301	844.9857	1060.2500	12723
1938	97	1981	645.2186	926.0000	11112
1939	66	1218	455.5248	758.5833	9103
1940	135	1900	656.9018	1087.1667	13046
1941	109	1543	536.3075	814.4167	9773
1942	70	1941	682.9455	914.8333	10978
1943	67	1559	575.8795	774.1667	9290
1944	87	1389	484.4304	703.4167	8441
1945	78	1289	500.2867	658.7500	7905
1946	86	1182	372.8950	548.8333	6586
1947	76	2406	826.6458	993.4167	11921
1948	88	1751	596.8337	851.1667	10214
1949	79	3058	1049.0521	1072.1667	12866

year	min	max	sd	mean	sum
1950	79	1244	400.4206	664.4167	7973
1951	93	2200	819.0423	987.4167	11849
1952	44	1393	498.2804	742.6667	8912
1953	67	2625	962.4954	1078.5000	12942
1954	76	1769	574.2831	731.2500	8775
1955	75	2023	735.6130	873.0000	10476
1956	83	1991	757.9808	952.7500	11433
1957	72	1443	502.9251	607.2500	7287
1958	89	1590	528.9370	700.2500	8403
1959	62	1401	492.4575	644.2500	7731
1960	95	1485	523.0446	670.1667	8042
1961	62	1022	368.7874	515.8333	6190
1962	170	1277	418.6054	659.4167	7913
1963	91	1105	370.5972	556.9167	6683
1964	82	955	311.5487	515.2500	6183
1965	139	1274	421.1494	618.5833	7423
1966	43	659	231.7896	347.6667	4172
1967	72	1088	353.9860	491.4167	5897
1968	51	736	264.1472	355.5833	4267
1969	82	626	199.9429	303.3333	3640
1970	58	1042	329.3564	414.8333	4978
1971	63	718	236.4735	317.9167	3815
1972	320	1121	318.3403	748.1667	4489

```
stats_2 <-  stats %>%
           summarise_all(list(min=min, max=max,mean=mean))

dtf <- round(sapply(stats,plyr::each(min, max, mean, sd, var, median, IQR)),3)
```

	year	min	max	sd	mean	sum
min	1931.000	43.000	626.000	199.943	303.333	3640.000
max	1972.000	320.000	3058.000	1049.052	1087.167	13046.000
mean	1951.500	85.929	1535.405	533.317	732.595	8684.262
sd	12.268	44.247	548.786	202.233	219.054	2710.841
var	150.500	1957.824	301166.296	40898.005	47984.625	7348657.369
median	1951.500	78.000	1422.000	501.606	734.875	8608.000
IQR	20.500	22.500	784.500	279.321	302.417	4116.000

7. Choix de modèle
   L'intercept n'étant pas significatif, on rejoue le modèle en l'excluant.
   Le modèle obtenu permet de confirmer liaison entre l'écart-type et la moyenne des observations. On en déduit que le modèle est multiplicatif. En cas d'ambiguité lors l'utilisation de cette méthode, il est d'usage recommandé d'utilisé un modèle multiplicatif complet.

```
modele = lm(stats$sd ~ stats$mean)
summary(modele)
```

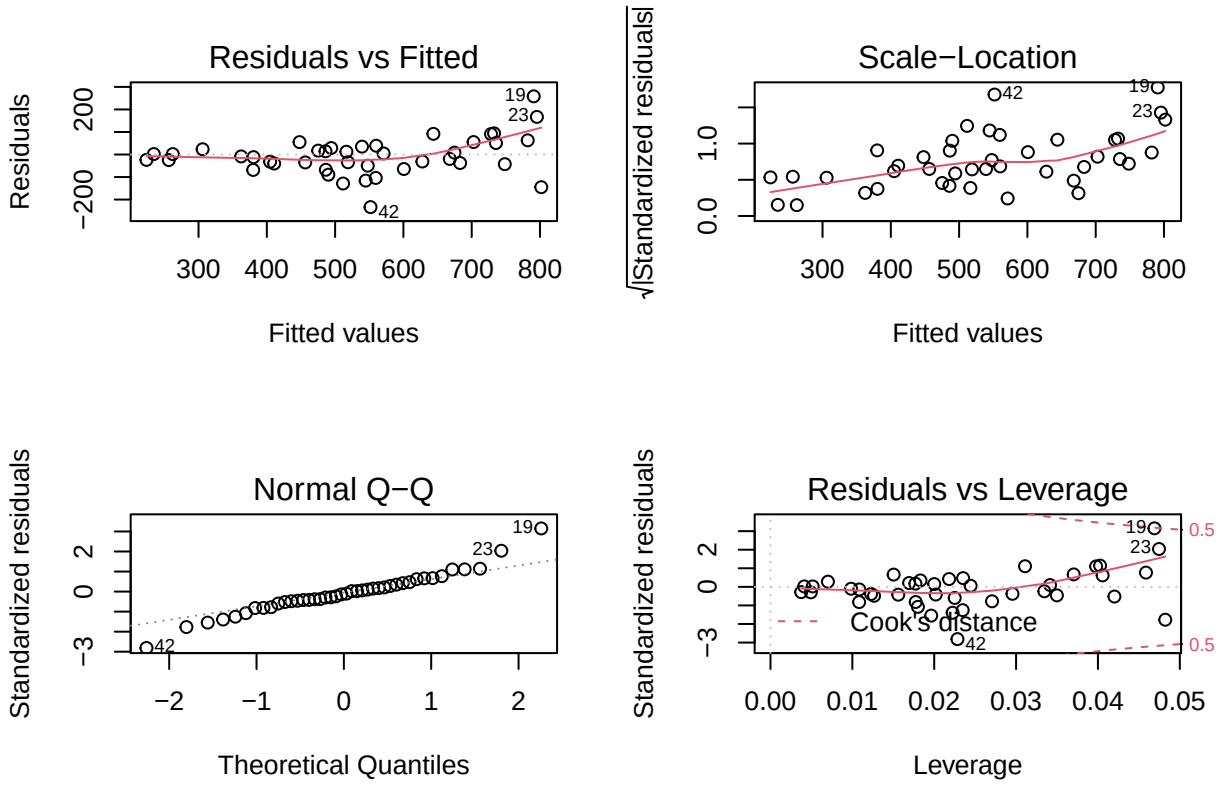
```
##
## Call:
## lm(formula = stats$sd ~ stats$mean)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -228.18  -42.06   21.14   42.52  227.86
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -87.75476   44.14788  -1.988   0.0537 .
## stats$mean    0.84777    0.05779  14.669   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 81.06 on 40 degrees of freedom
## Multiple R-squared:  0.8432, Adjusted R-squared:  0.8393
## F-statistic: 215.2 on 1 and 40 DF,  p-value: < 2.2e-16
```

```
modele = lm(stats$sd ~ stats$mean -1)
summary(modele)
```

```
##
## Call:
## lm(formula = stats$sd ~ stats$mean - 1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -233.51  -42.17  -10.09   33.37  258.22
##
## Coefficients:
##            Estimate Std. Error t value Pr(>|t|)
## stats$mean  0.73760    0.01695   43.51   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 83.93 on 41 degrees of freedom
## Multiple R-squared:  0.9788, Adjusted R-squared:  0.9783
## F-statistic:  1893 on 1 and 41 DF,  p-value: < 2.2e-16
```

```
layout(matrix(1:4,2,2))
plot(modele)
```




```
layout(c(1,1))

# NORMALITE
anova(modele)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
stats$mean	1	13333960.1	13333960.127	1892.938	0
Residuals	41	288806.3	7044.056	NA	NA

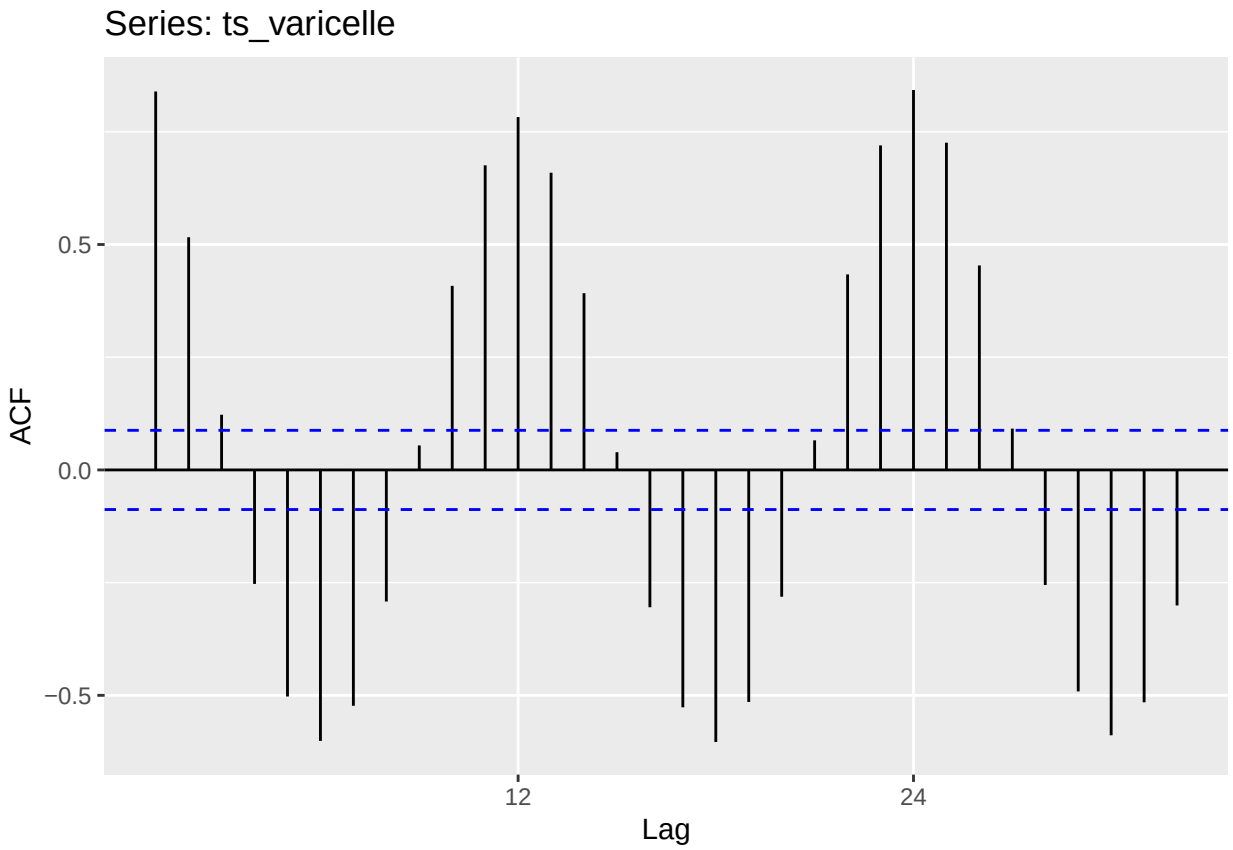
```
shapiro.test(residuals(modele))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(modele)
## W = 0.95913, p-value = 0.1375
```

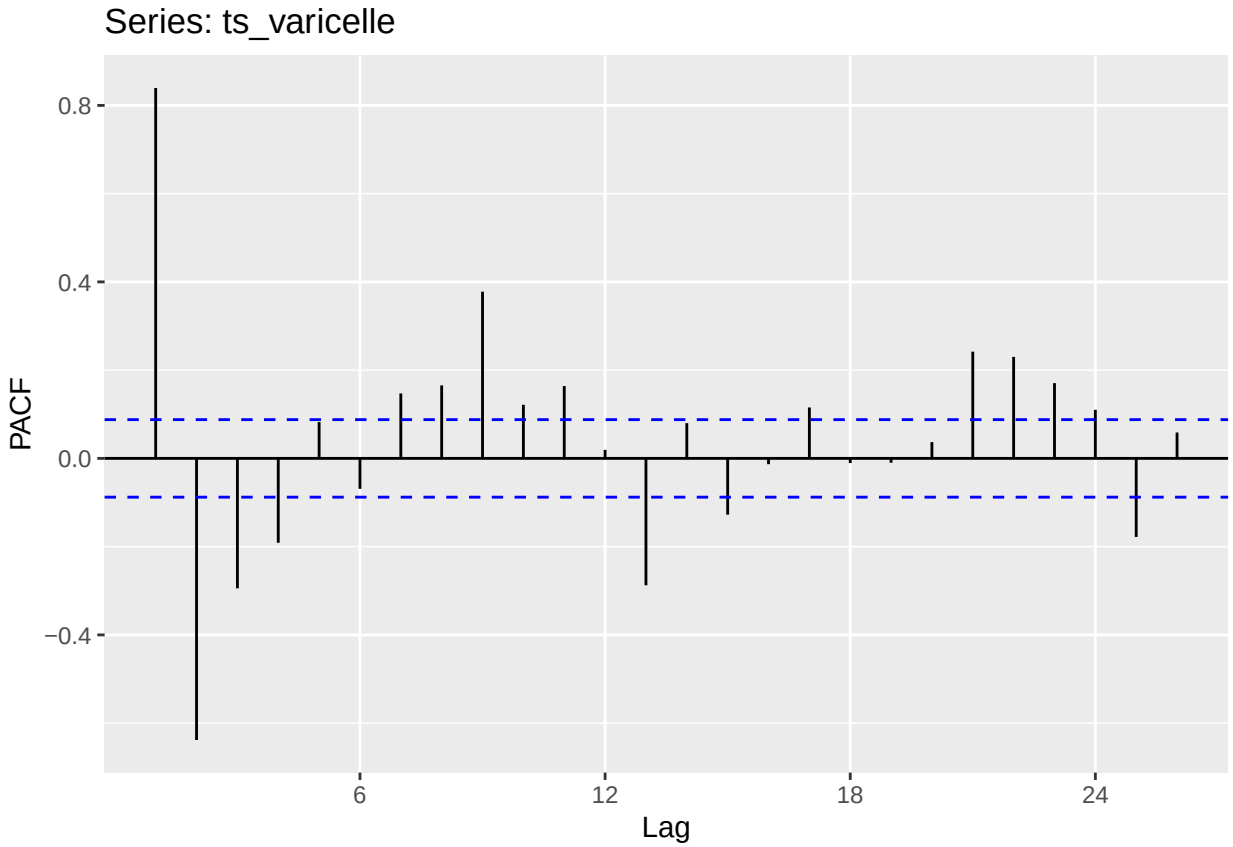
$$\sigma = 0.7375991 * \mu$$

ACF et PACF

```
ggAcf(ts_varicelle, lag.max = 32, type = "correlation", plot = TRUE)
```




```
ggPacf(ts_varicelle)
```



L'ACF permet de confirmer la saisonnalité de la série.

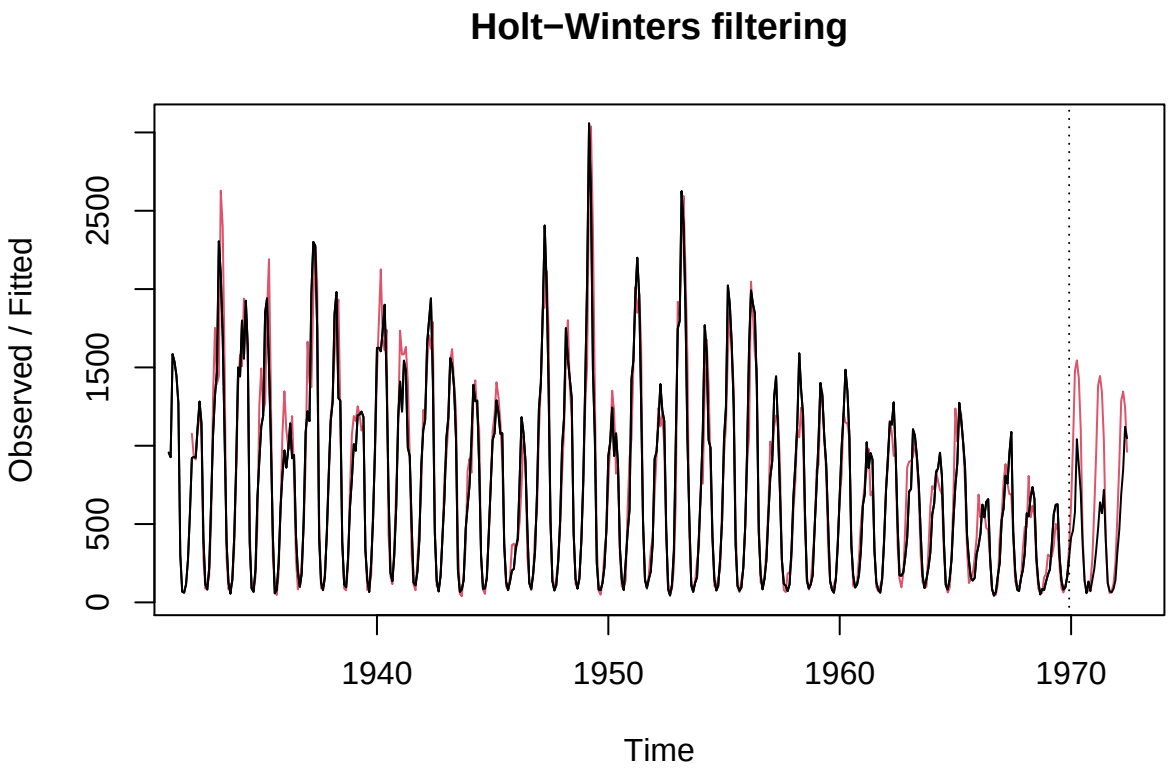
Lissages Exponentiels

1. Lissage de HW multiplicatif

```
xwr <- window(ts_varicelle,start=c(1931,1),end=c(1969,12))
Holt_winters <- HoltWinters(xwr, alpha = NULL, beta = NULL, gamma = NULL, seasonal = "multiplicative",)
prevision_HW = predict(Holt_winters,30)
```

Représentation prévision

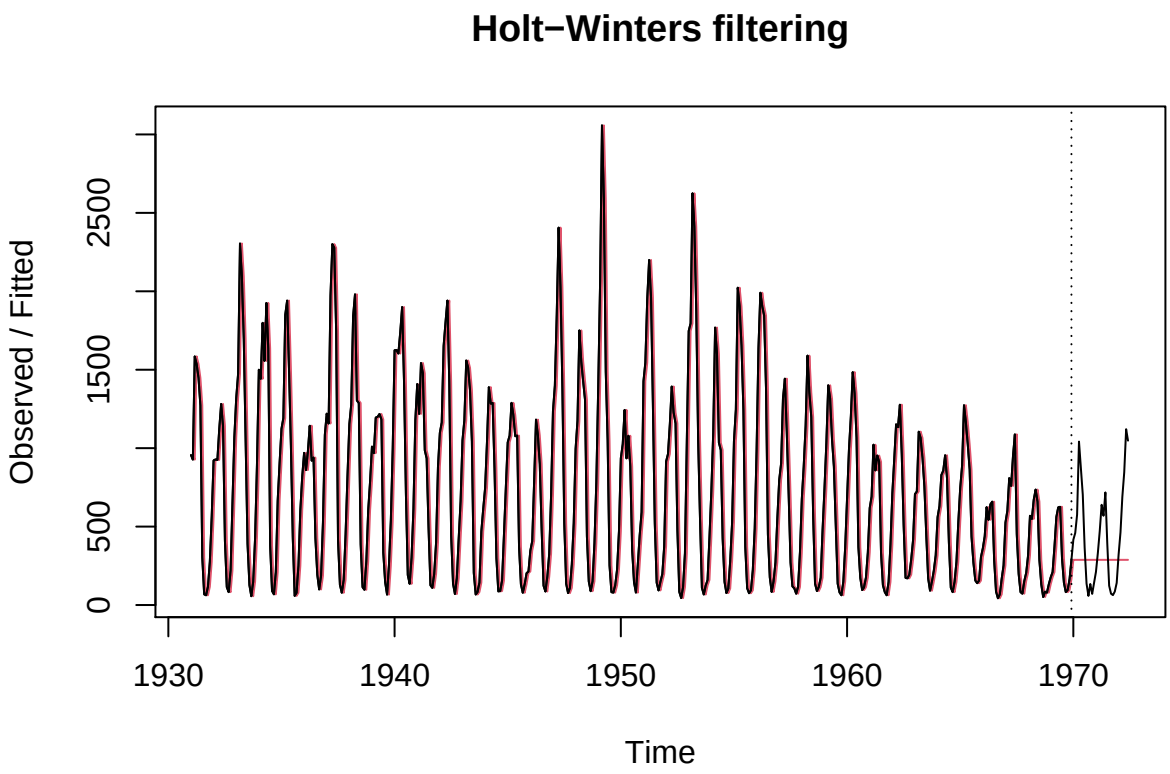
```
plot(Holt_winters,prevision_HW)
lines(ts_varicelle)
```



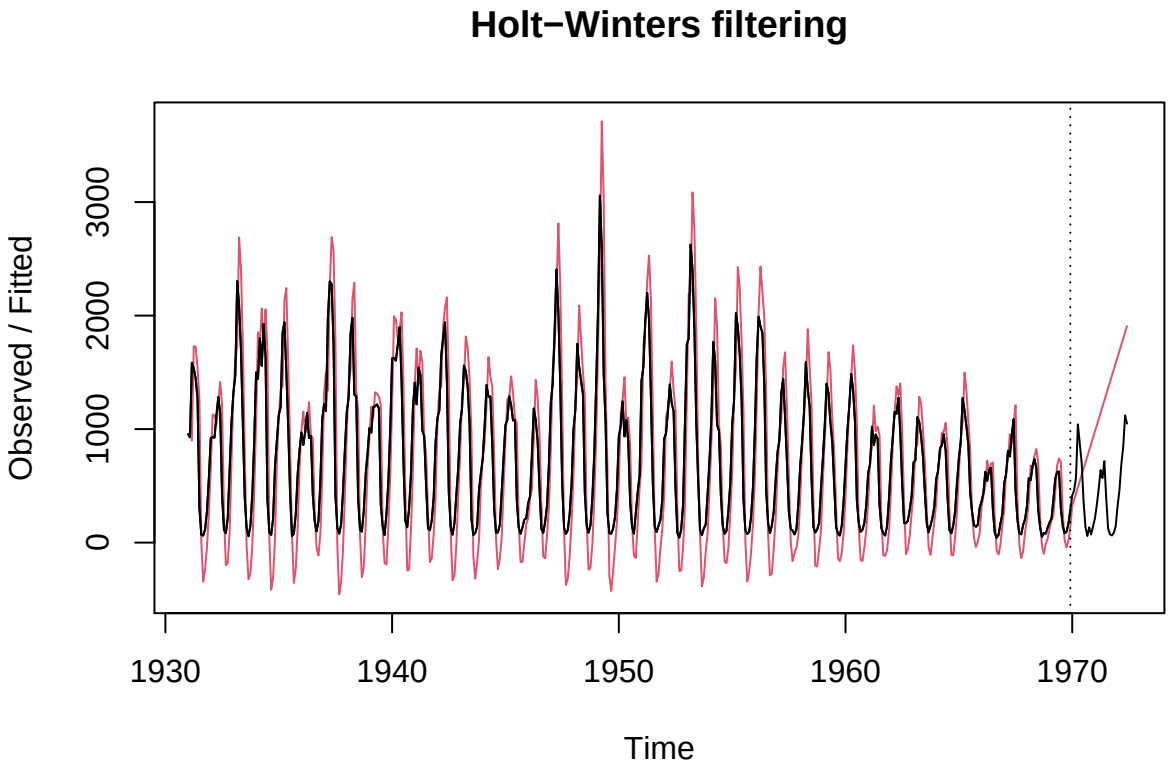
Le modèle obtenu semble adapté au jeu de données.

Lissage exponetielle simple et double

```
# LES
LES <- HoltWinters(xwr,alpha=NULL,beta=FALSE,gamma=FALSE)
p_les=predict(LES,n.ahead=30)
plot(LES,p_les)
lines(ts_varicelle)
```




```
# LED
alpha_1 <-sqrt(1-0.86)
LED <- HoltWinters(xwr, alpha = 0.86, beta = (1-alpha_1)/(1+alpha_1), gamma = FALSE)
p_led=predict(LED,n.ahead=30)
plot(LED,p_led)
lines(ts_varicelle)
```



4. Le lissage multiplicatif saisonnier semble plus approprié.Ce choix se confirme en comparant les erreurs quadratiques moyennes des trois modélisations, où on observe que c'est cette méthode qui minimise l'erreur quadratique moyenne.

	Erreur quadratique moyenne
Lissage exponentiel simple	116461.67
Lissage exponentiel double	142786.53
Holt winter multiplicatif	34707.53