

Grau en Estadística

Títol: Creació d'un sistema d'alerta epidemiològica basat en models temporals discrets

Autor: Júlia Corominas Del Hoyo

Director: David Moriña Soler

Departament: Departament d'Econometria, Estadística i Economia Aplicada

Convocatòria: Gener 2022



Agraïments

En primer lloc, m'agradaria expressar el meu agraïment al meu tutor, el Dr. David Moríña, per fer d'aquesta travessia una experiència tan agradable. Per tot l'empeny, confiança i dedicació que m'ha brindat durant l'últim quadrimestre, a més de tots els coneixements aportats. Sens dubte, ell ha sabut transmetre'm tota la seva passió per l'estadística i les sèries temporals i poder fer aquest treball amb ell ha estat tot un privilegi.

A la meva família i amigues, per ajudar-me en els moments més complicats i compartir els més feliços. De tots ells, he après moltes coses, entre elles, la constància, l'esforç i el plaer d'aprendre.

Resum

L'objectiu d'aquest treball és construir un sistema d'alerta epidemiològica que pugui anticipar l'aparició de brots i permeti implementar les mesures necessàries a temps per a poder frenar l'augment de casos. Per això, s'analitzarà l'evolució de casos de la grip per tal de detectar l'increment anòmal de casos diagnosticats d'aquesta malaltia relacionats amb l'aparició de la Covid-19 a finals de 2019 en les diferents regions sanitàries de Catalunya, fent ús de diferents models de sèries temporals discretes i clàssics. Moltes vegades existeixen processos on els seus elements d'anàlisi només poden prendre valors enters positius, a més, aquests tenen una dependència lineal al llarg del temps. Per aquestes ocasions quan es decideix desenvolupar una anàlisi de sèries temporals s'utilitza un dels processos anomenat Integer-valued Autoregressive (INAR), el model INAR(k) on k representa l'ordre d'autocorrelació.

Paraules Clau: Grip, Covid-19, Regions Sanitàries, Catalunya, Sèries temporals, Sèries temporals discretes, Model INAR.

Classificació AMS : 37M10 - Time series analysis, 62M10 - Time series, auto-correlation, regression, etc, 92C60 - Medical epidemiology, 92D30 - Epidemiology

Abstract

The objective of this work is to build an epidemiological alert system that can anticipate the appearance of outbreaks and allow the necessary measures to be implemented in time to be able to slow down the increase in cases. Therefore, the evolution of influenza cases will be analyzed in order to detect the abnormal increase in diagnosed cases of this disease related to the appearance of Covid-19 at the end of 2019 in the different health regions of Catalonia, making use of different models of discrete and classical time series. There are often processes where its analysis elements can only take positive integer values, in addition, these have a linear dependence over time. For these occasions when it is decided to develop a time series analysis one of the processes called Integer-valued Autoregressive (INAR) is used, the INAR model (k) where k represents the autocorrelation order.

Keywords: Influenza, Covid-19, Health Regions, Catalonia, Time series, Discrete time series, INAR model.

AMS Classification : 37M10 - Time series analysis, 62M10 - Time series, auto-correlation, regression, etc, 92C60 - Medical epidemiology, 92D30 - Epidemiology

Índex General

1	Introducció	1
1.1	Motivació	1
1.2	La Grip i la Covid-19	2
1.3	Sistema de vigilància epidemiològica	2
1.3.1	Vigilància basada en la xarxa de metges sentinella	3
1.3.2	Vigilància basada en la xarxa de farmàcies Sentinella	3
1.4	Objectiu del treball	4
2	Gestió de la base de dades	5
2.1	Origen de les dades	5
2.2	Variables d'estudi	5
3	Models de sèries temporals i conceptes bàsics	8
3.1	Introducció	8
3.2	Sèries temporals clàssiques	10
3.2.1	Models AR	10
3.2.2	Models MA	10
3.2.3	Models ARMA	10
3.3	Models integrats	11
3.3.1	Models ARIMA	11
3.3.2	Models SARIMA	12
3.3.3	Estimació dels paràmetres i validació del model	12
3.3.4	Regions de predicció	14
3.4	Sèries temporals discretes	14
3.4.1	Models INAR	14
3.4.2	Estimació dels paràmetres	15
3.4.3	Regions de predicció a curt termini	16
4	Resultats	17
4.1	Anàlisi descriptives	17
4.2	Identificació del model	20
4.2.1	Coeficients d'autocorrelació (ACF)	20

4.2.2	Coeficients d'autocorrelació parcials (PACF)	21
4.3	Estimació dels paràmetres i validació de la proposta	23
4.3.1	Validació de la proposta	25
4.4	Prediccions	29
4.4.1	Regions de predicció a curt termini	29
4.5	Barcelona	34
4.5.1	Definició del model	34
4.5.2	Validació	35
4.5.3	Prediccions	35
5	Discussió i conclusions	37
A	Codi R	41

Índex de figures

4.1	Representació gràfica del nombre de casos detectats en les Regions Sanitàries de Camp de Tarragona i Terres de l'Ebre	18
4.2	Representació gràfica del nombre de casos detectats en les Regions Sanitàries de l'Alt Pirineu i Aran i Lleida	18
4.3	Representació gràfica del nombre de casos detectats en les Regions Sanitàries de Girona i Catalunya Central	19
4.4	Representació gràfica del nombre de casos detectats a la Regió Sanitària de Barcelona	19
4.5	ACF de les diferents regions sanitàries de Catalunya	20
4.6	PACF de les diferents regions sanitàries de Catalunya	21
4.7	Regió de predicció per la temporada 2011-2012 de Tarragona	25
4.8	Regió de predicció per la temporada 2011-2012 de les Terres de l'Ebre	26
4.9	Regió de predicció per la temporada 2011-2012 de l'Alt Pirineu i Aran	26
4.10	Regió de predicció per la temporada 2011-2012 de Lleida	27
4.11	Regió de predicció per la temporada 2011-2012 de Girona	27
4.12	Regió de predicció per la temporada 2011-2012 de Catalunya Central	28
4.13	Regions de predicció per la temporada 2019-2020 a Tarragona	30
4.14	Regions de predicció per la temporada 2019-2020 a les Terres de l'Ebre	30
4.15	Regions de predicció per la temporada 2019-2020 a l'Alt Pirineu i Aran	31
4.16	Regions de predicció per la temporada 2019-2020 a Lleida	31
4.17	Regions de predicció per la temporada 2019-2020 a Girona	32
4.18	Regions de predicció per la temporada 2019-2020 a Catalunya Central	32
4.19	ACF i PACF de la Regió Sanitària de Barcelona	34
4.20	Residus de la Regió Sanitària de Barcelona	35
4.21	Regions de predicció per la temporada 2019-2020 a Barcelona	36

Índex de taules

2.1	Variables d'interès de l'estudi	7
4.1	Nombre de sanitaris de les Regions Sanitàries	24
4.2	Estimadors i Error Est. dels paràmetres de màxima versemblança	24

Capítol 1

Introducció

1.1. Motivació

La Covid-19 és una malaltia infecciosa causada a partir del coronavirus que va aparèixer per primera vegada a Wuhan (Xina), a finals del desembre del 2019. La seva ràpida propagació, la seva transmissió aèria i la necessitat del mínim contacte per la infecció, va provocar que el març del 2020, l'Organització Mundial de la Salut (OMS) ho classifiqués com una pandèmia [1].

La pandèmia del Coronavirus ha estat i és el repte més gran per la salut que hi ha hagut aquest últim segle. La seva dimensió ha obligat a una gran majoria de països a adoptar mesures excepcionals, provocant tensions en els sistemes de Salut i en els mercats de productes sanitaris a escala mundial generant una crisi sociosanitària amb un gran impacte econòmic. Aquesta pandèmia ha posat en manifest la importància d'un bon sistema de Salut per disposar d'un pla per reforçar les capacitats: en l'àmbit assistencial, de salut pública i preparar-se per respondre qualsevol escenari en cas d'augment de transmissió, assumint que no existeix un únic enfocament en la gestió dels casos i dels brots de la Covid-19 i que es deuen avaluar els riscos en cada territori de Catalunya segons les seves particularitats.

Per tant, la comunicació immediata de troballes d'interès epidemiològic dels serveis assistencials és fonamental per a combatre aquesta situació com són la Xarxa de Vigilància Epidemiològica de Catalunya (XVEC) [2] i el Pla d'informació de les infeccions respiratòries agudes a Catalunya (PIDIRAC) [3]. Ambdós serveis comparteixen un clar objectiu dins la salut pública que és reduir l'impacte de les situacions d'emergència en la societat, ja que per fer front a una pandèmia amb garanties d'èxit cal disposar dels recursos necessaris per atendre les necessitats sanitàries immediates de la població, a més de tenir la capacitat de gestionar els nous casos i donar servei a tots els pacients en el context de la pandèmia, en la qual la majoria d'aspectes més rellevants, com el moment de la irrupció o la ràtio d'infecció, seran desconeguts.

En el treball que es presenta a continuació s'intentarà construir un sistema d'alerta epidemiològica que permeti anticipar l'aparició de brots, facilitant la implementació de mesures necessàries per a poder avisar amb antelació a la societat d'aquest nou brot.

1.2. La Grip i la Covid-19

La grip és una malaltia infecciosa respiratòria contagiosa causada pel virus de la influença, que principalment infecten el nas, la gola, els bronquis i, en alguns casos, els pulmons. Aquesta es pot transmetre de persona malalta a persona afectada amb unes simples gotes de saliva expulsades mentre parles, tocant superfícies contaminades, etc. Acostuma a aparèixer en forma epidèmica un cop l'any, generalment l'hivern. És per això que, durant els mesos de tardor i hivern, el PIDIRAC fa un seguiment de les persones afectades per a poder agilitzar la detecció i detenció de l'epidèmia gripal.

El coronavirus SARS-CoV-2, apareixent a finals de desembre del 2019, és un nou tipus de virus que causa una malaltia infecciosa anomenada Covid-19. Aquesta, també es transmet de la mateixa manera que la grip, és a dir, per contacte directe amb possibles secrecions respiratòries, però té una alta propagació de contagi de la malaltia.

Així doncs, ambdues són malalties respiratòries i tot i que els virus que les causen són diferents, els símptomes i la seva propagació són similars; és per això que no s'ha de confondre, ja que presenten diferències bàsiques. Mentre la grip dura una o dues setmanes, la Covid-19 es pot allargar molt més. A més, aquesta última també té símptomes característics com són la pèrdua de gust i olfacte en els pacients infectats.

El fet que les dues malalties comparteixen símptomes, a principis de la pandèmia va crear una gran problemàtica a causa de la confusió del diagnòstic, per la qual cosa aquest és un dels motius pels quals el desembre del 2019, no es van detectar els possibles casos de Covid-19 com a Covid-19, sinó com a grip, ja que no sabien diferenciar-les correctament.

1.3. Sistema de vigilància epidemiològica

Els sistemes de vigilància són una eina de salut pública bàsica i contemplen la detecció i la intervenció urgent en determinades situacions. Aquests han de ser àgils per a poder tenir un control de les malalties que es transmeten més ràpides i intentar facilitar la convivència entre els nous virus i els ja coneguts.

Tradicionalment, dins dels sistemes de salut, la vigilància ha tingut el propòsit d'identificar els aspectes més importants dels riscos que comporten diferents malalties per poder dissenyar estratègies que impedeixin la propagació d'aquestes i puguin malmetre l'estat de salut de la població. La combinació d'activitats de vigilància és específica per cada país i regió, i encara més per cada ciutat i localitat, és a dir, es considera un procés que va variant segons ho faci el perfil de salut de la població.

El sistema de vigilàncies epidemiològiques es considera una component essencial en els sistemes moderns de salut i una eina bàsica per a l'elaboració de les polítiques de salut. A Catalunya, l'estratègia de vigilància epidemiològica de la grip, es realitza des de la setmana 40 de l'any en curs fins a la setmana 20 de l'any següent i es recull en el Pla d'informació de les infeccions respiratòries agudes de Catalunya (PIDIRAC) [4].

Des del 2017, l'estratègia de vigilància epidemiològica de la grip no només agrupa la vigilància en el registre de visites mèdiques que realitzen els metges d'intenció primària i la vigilància basada en la xarxa de metges sentinella sinó també la xarxa de farmàcies sentinella de Catalunya.

Aquest nou sistema permet obtenir informació diària sobre la quantitat de persones que pateixen les infeccions respiratòries agudes ateses a les farmàcies sentinella del sistema de vigilància sindròmica de la grip, obtenir la informació sobre la medicació que reben els pacients i detectar de la manera més ràpida possible l'aparició de les epidèmies de la grip [5].

1.3.1. Vigilància basada en la xarxa de metges sentinella

La vigilància basada en la xarxa de metges sentinella és un sistema que ajuda a complementar el PIDIRAC. Aquesta xarxa recull informació durant la temporada gripal, és a dir, des de la setmana 40 d'un any fins a la setmana 20 de l'any següent, ja que s'intenta disposar d'informació actualitzada com els indicadors de morbiditat atesa recollits pels metges sentinella amb periodicitat diària per poder veure l'activitat representada pels virus gripals, com per altres virus respiratoris que, depenent de la seva freqüència, potencial i gravetat poden ocasionar augments destacables en la demanda d'assistència al CAP o domiciliàries [6].

Així doncs, els metges sentinella poden notificar algun cas o agrupació de casos, possibles sospitosos d'infecció per virus de la grip a partir de mostres que s'envien a laboratoris de referència per a la seva confirmació virològica [7].

1.3.2. Vigilància basada en la xarxa de farmàcies Sentinella

La xarxa de farmàcies sentinella a Catalunya té com a objectiu principal donar suport als sistemes de notificació, agilitzar l'eficiència de les xarxes de vigilància que existeixen i poder reconèixer els hàbits i comportaments de la societat, per poder tenir una visió global de la càrrega que té malaltia en la població.

Aquesta xarxa va estar creada, ja que és el centre d'atenció sanitària més pròxim al pacient i pot complir amb les necessitats d'aquest tenint en compte que majoritàriament els processos són de baixa complexitat. És per això, que s'ha impulsat la integració en els sistemes sanitaris moderns, perquè el farmacèutic és capaç d'actuar en diferents escenaris.

Per tant, les farmàcies sentinella poden treballar en diferents línies d'actuació, ja sigui la detecció d'errors de medicaments o el fet d'identificar un cas de sospita, accelerant el ritme de notificació de casos de grip [8].

1.4. Objectiu del treball

L'objectiu d'aquest treball que es presenta a continuació és construir un sistema d'alerta epidemiològica que pugui anticipar l'aparició de brots i així permeti implementar les mesures necessàries a temps per poder frenar l'augment de casos. És per això, que s'analitzarà l'evolució dels casos de la grip per tal de poder detectar l'increment anòmal de casos diagnosticats d'aquesta malaltia relacionats amb l'aparició de la Covid-19 a finals de 2019 en les diferents regions sanitàries de Catalunya.

Per portar a terme aquest estudi se seguirà el següent pla de treball:

Primer de tot, es realitzarà una revisió bibliogràfica sobre els models proposats per analitzar sèries temporals discretes, amb especial atenció als models basats en l'operador *thinning*.

Seguidament, es treballarà en la identificació del model que s'ajusti millor a l'evolució històrica de casos de la grip a Catalunya l'any 2019 i en el desenvolupament de les eines que permetin identificar un comportament anòmal en la sèrie històrica; en concret els models AR i INAR. Discutir les diferències, decidir quin model s'ajusta millor a la sèrie i és capaç d'explicar-la mitjançant les prediccions. També, es prestarà atenció a les sèries temporals clàssiques, ja que no totes les Regions sanitàries tenen el mateix nombre de població i es vol comprovar per quines regions sanitàries és efectiu un model o un altre.

L'anàlisi temporal dels casos de la grip a Catalunya, en funció de les diferents regions sanitàries, es pretén dur a terme mitjançant la parametrització dels models capaços d'explicar la sèrie, ajustant-se correctament a la volatilitat dels possibles valors i predir amb èxit la seva evolució. Caldrà tenir en compte que s'està considerant que cada regió sanitària és diferent, a causa de la gran diferència de densitat de població entre aquestes; és per això, que s'observarà diferents models per les diferents regions.

Finalment, es crearà i posarà en marxa el sistema d'alerta epidemiològica en una pàgina web que permeti fer el seguiment del sistema d'alerta epidemiològica desenvolupat segons les set regions sanitàries de Catalunya.

El present Treball Final de Grau s'estructura de la següent forma:

1. **Gestió de la base de dades dels casos diagnosticats de la grip l'any 2019:** Lectura de la base de dades i anàlisi estadística d'aquesta.
2. **Models estadístics i conceptes bàsics:** Introducció als models de sèries temporals discretes i explicació sobre els conceptes bàsics que s'utilitzen en el treball.
3. **Construcció, validació i capacitat predictiva dels models:** Construcció i descripció dels models de sèries temporals discretes finals descrits anteriorment, validació i anàlisi de les estimacions.
4. **Resultats:** Resultats obtinguts a partir de les prediccions dels models i anàlisi d'aquests.
5. **Discussió i conclusions:** Conclusions a les quals s'han arribat després de realitzar l'estudi.

Capítol 2

Gestió de la base de dades

2.1. Origen de les dades

La base de dades que s'ha utilitzat per a la realització d'aquest Treball de Final de Grau prové del Sistema d'Informació dels Serveis d'Atenció Primària, *SISAP*, una plataforma amb el propòsit de proporcionar informació, principalment per a la gestió clínica, als diferents professionals i a les estructures de gestió. És a dir, obtenir dades i processar-les per construir indicadors. A partir d'aquest sistema, l'any 2011 es va crear una web anomenada *Diagnosticat* amb l'objectiu principal de facilitar informació de forma oberta sobre els diagnòstics de les malalties transmissibles més rellevants de l'atenció primària de Catalunya. Aquesta mostra corbes epidèmiques, piràmides d'edat i una taula resum que es pot descarregar fàcilment de diferents malalties transmissibles, entre els quals es troba la varicel·la o la grip, sent aquestes les dades que s'han utilitzat per a aquest treball i que es pot trobar en el següent enllaç:

<https://www.ics.gencat.cat/sisap/diagnosticat/principal>

La font de dades de *Diagnosticat* són els registres dels problemes de salut recollits mitjançant l'estació clínica d'atenció primària, *eCAP*, que és el programa d'història clínica informatitzada fet servir pels professionals de totes les unitats d'atenció primària de l'Institut Català de la Salut, *ICS*, i per un nombre creixent d'altres proveïdors d'atenció primària de Catalunya.

2.2. Variables d'estudi

La base de dades que s'ha utilitzat per a la realització d'aquest treball consta de 13 variables i de 598 observacions. En aquesta font es troba la taxa setmanal de la malaltia de les set regions sanitàries de Catalunya i també del total de Catalunya. Les dades que es presenten són dades agregades sense informació identificativa de cap pacient. No només apareixen les dades per totes les regions sanitàries de Catalunya que són: *Lleida, Camp de Tarragona, Barcelona, Girona, Catalunya Central, Alt Pirineu i Aran, Terres de l'Ebre* i també el conjunt total de Catalunya, sinó també la temporada d'any i la setmana epidemiològica en què es van comptabilitzar els

casos. Les taxes que fan referència a les 8 variables anomenades anteriorment es calculen dividint els casos setmanals per la població total assignada i multiplicada per 100.000 per poder donar una taxa per 100 mil habitants. Es pren aquest valor com a taxa, ja que facilita que la mesura es pugui comparar amb dades de diferents territoris de forma independent al nombre d'habitants.

La variable Setmana representa la setmana epidemiològica, sent aquesta una agrupació de set dies d'ençà que comença el diumenge i acaba el dissabte següent. La primera setmana d'un any acaba el primer dissabte de gener, sempre que els 4 primers dies d'aquella setmana siguin també del mes de gener. Es fa ús d'aquestes perquè l'any epidemiològic sempre té cinquanta-tres setmanes, mentre que si es representés amb setmanes naturals, no facilitaria l'estudi. Des de DiagnostiCat es va decidir estudiar amb les dades actualitzades cada setmana, ja que diàries serien massa freqüents per ser gestionades i analitzades i podrien portar afectacions per moltes circumstàncies, com el fet que no es fan gaires diagnòstics per motius organitzatius o laborals els caps de setmana o els dies festius. D'altra banda, un mes seria un període massa llarg, perquè les dades es necessiten amb més rapidesa i això ja s'obté amb els talls setmanals.

La variable Temporada apareix, ja que la grip és l'única malaltia que no es mostra en anys. Aquesta es presenta en temporades a causa de les característiques pròpies de la mateixa grip. Una epidèmia gripal acostuma a iniciar-se a finals d'any i s'allarga fins a mitjans de l'any següent amb pics màxims entre el desembre i gener; per això, una temporada gripal comença a la setmana 23 d'un any i finalitza a la setmana 22 de l'any següent. En la font de dades que es treballa, apareixen dades des de la temporada 2010-2011 fins a la temporada 2021-2022, tot i que aquesta última encara no ha acabat.

Per facilitar la lectura i interpretació de les dades, es va decidir crear tres noves variables que farien referència a l'any, data exacta de l'any i mes. La primera variable creada Any mostra exactament en quin any es troba la setmana epidemiològica de cada regió, independentment de la temporada. La segona variable *Date* pren el dia, mes i any exacte d'aquesta setmana epidemiològica, facilitant així la representació dels pics on hi ha més casos de grip. Finalment, la variable *Month*, dada extreta de la variable *Date* representa el mes en el qual es troba aquella setmana. Per crear aquestes variables s'ha fet ús de la funció *MMWRweek2Date* del paquet *MMWRweek* (Jarad Niemi, 22-04-2020) [9].

Les variables que s'han decidit estudiar finalment són les que es presenten a la Taula 2.1.

Variable	Descripció
Temporada	Temporada gripal
Setmana	Setmana epidemiològica
Lleida	Nombre de casos setmanals per la regió sanitària de Lleida
Camp.de.Tarragona	Nombre de casos setmanals per la regió sanitària del Camp de Tarragona
Barcelona	Nombre de casos setmanals per la regió sanitària de Barcelona
Girona	Nombre de casos setmanals per la regió sanitària de Girona
Catalunya.Central	Nombre de casos setmanals per la regió sanitària de Catalunya Central
Alt.Pirineu.i.Aran	Nombre de casos setmanals per la regió sanitària de l'Alt Pirineu i Aran
Terres.de.l.Ebre	Nombre de casos setmanals per la regió sanitària de les Terres de l'Ebre
Catalunya	Nombre de casos setmanals de Catalunya
Any	Any de la setmana epidemiològica
Date	Data de la setmana epidemiològica
MONTH	Mes de la setmana epidemiològica

Taula 2.1: Variables d'interès de l'estudi

Capítol 3

Models de sèries temporals i conceptes bàsics

En aquest capítol s'explica els models de sèries temporals que s'utilitzaran en aquest treball per analitzar la sèrie de casos de la grip l'any 2019 a les diferents regions sanitàries de Catalunya. Per realitzar l'explicació d'aquests models se seguirà la informació de l'article *A statistical model for hospital admissions caused by seasonal diseases* [10].

3.1. Introducció

Una sèrie temporal és un conjunt d'observacions X_t , cadascuna recollida en un moment del temps concret t . Aquesta es defineix com una successió de valors d'una variable aleatòria ordenats cronològicament que s'expressa com:

$$x_1, x_2, x_3, \dots, x_t,$$

on t representa la posició temporal que ocupa el valor dins la sèrie.

L'anàlisi clàssica de les sèries temporals suposa que una sèrie es pot descompondre en totes o algunes de les següents components:

1. **Tendència:** Component que representa l'evolució a llarg termini de la sèrie. Aquesta de comportament estable en llargs períodes de temps pot ser creixent o decreixent.
2. **Cicle:** Component que recull moviments oscil·latoris per sobre i per sota de la tendència que es mantenen en períodes superiors a un any.
3. **Component Estacional:** Component que recull aquelles oscil·lacions d'una sèrie temporal que es completen dins un any i que es repeteixen en anys successius. En conseqüència, el període d'aquesta és igual o inferior a un any.

4. **Component irregular:** Fa referència a variacions esporàdiques de la sèrie que no estan recollides en la tendència, el cicle ni la component estacional, tenint, per tant, un caràcter residual.

Una de les estructures més habituals en el context de l'anàlisi de les sèries temporals és l'estructura dels models autoregressius, que segueixen la definició

$$X_t = \sum_{i=1}^p \alpha_i X_{t-i} + \varepsilon_t, \quad (3.1)$$

on $\alpha_1, \dots, \alpha_p$ són els paràmetres del model i ε_t és un soroll blanc. Pel que fa a l'estacionarietat del procés, cal que totes les arrels de z_1, \dots, z_p del polinomi característic $P(z) = z^p - \sum_{i=1}^p \alpha_i z^{p-i}$ satisfacin $|z_i| < 1$.

Els paràmetres d'un model autoregressiu es poden estimar mitjançant diverses tècniques, ja siguin basant-se en les equacions de Yule-Waler, el mètode dels moments o el mètode de màxima versemblança.

Una altra part important dins les sèries temporals són les mesures d'associació entre els valors de les sèries actuals i passades i indiquen quins són els valors de les sèries passades més útils per poder predir els futurs valors, aquests són els coeficients d'autocorrelació (ACF) i els coeficients d'autocorrelació parcials (PACF).

L'autocorrelació mesura la correlació entre dues variables separades per k períodes,

$$\rho_j = \text{corr}(X_j, X_{j-k}) = \frac{\text{cov}(X_j, X_{j-k})}{\sqrt{V(X_j)}\sqrt{V(X_{j-k})}}.$$

La funció d'autocorrelació simple té les següents propietats:

- $\rho_0 = 1$
- $-1 \leq \rho_j \leq 1$
- $\rho_j = \rho_{-j}$

L'autocorrelació parcial mesura la correlació entre dues variables separades per k períodes quan no es considera la dependència creada pels retards intermedis existents entre ambdues.

$$\pi_j = \text{corr}(X_j, X_{j-k}/X_{j-1}X_{j-2}\dots X_{j-k+1})$$

$$\pi_j = \frac{\text{cov}(X_j - \hat{X}_j, X_{j-k} - \hat{X}_{j-k})}{\sqrt{V(X_j - \hat{X}_j)}\sqrt{V(X_{j-k} - \hat{X}_{j-k})}}$$

3.2. Sèries temporals clàssiques

El pròxim capítol se centrarà a explicar els models de sèries temporals clàssiques que conformen una petita part de la metodologia. Seguidament, es detallarà les sèries temporals discretes, que conformen la part restant del gruix de la metodologia basada en [10].

3.2.1. Models AR

El model autoregressiu descriu el procés on les observacions en un instant són esperades a partir de les observacions antigues del procés més un terme d'error. Suposant que ε_t és un soroll blanc. Llavors un procés X_t s'anomena un procés autoregressiu d'ordre p , AR(p), si existeixen coeficients $\delta, \phi_1, \dots, \phi_p$ tal que:

$$X_t = \delta + \phi_1 X_{t-1} + \phi_2 X_{t-2} + \dots + \phi_p X_{t-p} + \varepsilon_t, \quad (3.2)$$

on $\varepsilon_t \sim N(0, \theta^2)$ i $\phi_1, \dots, \phi_p \in \mathfrak{R}$. A més, X_t es pot considerar com a una regressió lineal múltiple, ja que les variables explicatives són les p observacions passades del mateix procés: X_{t-1}, \dots, X_{t-p} . També es pot representar com $\phi(B)X_t = \delta + \varepsilon_t$. És per això, que pren el nom d'autoregressiu. Aquest procés d'ordre finit també es pot veure escrit com un procés MA d'ordre infinit, i per tant, també es pot escriure amb l'operador de retard ϕ .

S'han desenvolupat diferents models tenint en compte dos objectius principals: que assumeixin valors discrets i que tinguin un comportament equiparable als AR. En particular, una de les famílies de models més emprades és el model INAR (processos autoregressius enters) 3.4.1. Cal tenir present que amb l'estructura del model AR, la distribució marginal de X_t també segueix una normal i sabent que els recomptes que s'analitzen en aquest treball són relativament baixos, es justifica que no té sentit treballar amb el model AR.

3.2.2. Models MA

En els models de mitjanes mòbils, el valor actual es pot predir a partir de la conducta aleatòria del mateix instant. Un procés X_t es pot anomenar com a un procés de mitjanes mòbils d'ordre q , MA(q), si existeixen coeficients $\delta, \theta_1, \dots, \theta_q$ tal que:

$$X_t = \delta + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \dots - \theta_q \varepsilon_{t-q} = \delta + \theta(B)\varepsilon_t, \quad (3.3)$$

on ε_t és un soroll blanc amb mitjana zero i variància σ_z^2 i θ_i són constants. Un model MA(q) sempre és estacionari; els moments de primer i segon ordre són sempre finits i constants al llarg del temps.

3.2.3. Models ARMA

Un model autoregressiu de mitjana mòbil ARMA(p, q) segueix la següent forma:

$$X_t - \phi_1 X_{t-1} - \phi_2 X_{t-2} - \dots - \phi_p X_{t-p} = \delta + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \dots - \theta_q \varepsilon_{t-q}, \quad (3.4)$$

sent així un model amb una part autoregressiva i una part de mitjana mòbil. Per a saber si s'està treballant amb un model ARMA, cal comprovar si la part autoregressiva és estacionària i la part de mitjana mòbil és invertible. En el cas que la part AR sigui estacionària, es pot escriure com un model MA() i si la part MA és invertible es pot expressar com un model AR().

3.3. Models integrats

Per poder descriure el concepte de sèrie integrada, és necessari introduir el d'estacionarietat. Una sèrie estacionària es caracteritza per una mitjana constant que no varia en el temps i una variància constant i finita. Els processos no estacionaris comuns són els processos integrats, que es converteixen en estacionaris simplement prenent diferències. Pel que si Y_t no és estacionari però la sèrie Z_t de les primeres diferències:

$$Z_t = Y_t - Y_{t-1},$$

sí ho és, llavors Y_t és un procés integrat d'ordre 1, I(1). El terme integrat indica que el procés és la suma dels elements fins t d'un procés estacionari.

3.3.1. Models ARIMA

Els models de sèries temporals que s'han descrit anteriorment es basen en el supòsit d'estacionarietat, és a dir, la mitjana i la variància són constants en el temps i la covariància és invariant en el temps. Tot i això, moltes sèries no són estacionàries, algunes degudes al fet que la seva variància no és constant en el temps. Quan un procés no és estacionari, les seves diferències d'ordre d sí que ho són, el procés s'anomena procés integrat d'ordre d , i per això a la sèrie temporal cal aplicar-li diferències d vegades per fer-la estacionària.

Es defineix un model ARIMA(p,d,q) com una sèrie temporal autoregressiva integrada de mitjanes mòbils on p indica el nombre de termes autoregressius, d el nombre de vegades que a la sèrie cal aplicar les diferències per fer-la estacionària i q el nombre de termes de mitjana mòbil. Definit per la següent expressió:

$$\phi(B)(1 - B)^d X_t = \delta + \theta(B)\varepsilon_t, \quad (3.5)$$

on:

$$\phi_p(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p$$

$$\theta_p(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_p B^p,$$

on la primera fórmula és el polinomi AR(p) i la segona el polinomi MA(q).

Per tant, un model ARIMA representa la influència conjunta sobre el valor pres per la variable en cada moment, dels seus valors anteriors i els sorolls anteriors, ja siguin recents si la sèrie té estacionalitat. Si això últim passa, s'acostuma a identificar el model com a SARIMA.

3.3.2. Models SARIMA

L'estacionalitat d'una sèrie temporal és un patró regular de canvis que es repeteix durant S períodes de temps, on S és el nombre de períodes de temps fins que el patró es repeteixi novament.

Per exemple, existeix estacionalitat en les dades mensuals pels quals els valors alts tendeixen a passar sempre en alguns mesos en particular i els valors baixos tendeixen a passar sempre en altres mesos en particulars. En aquest cas, $S = 12$, mesos per any és el valor del comportament estacional periòdic.

Per tant, en el cas que en el model ARIMA es pugui considerar l'efecte de l'estacionalitat, es parla del model SARIMA amb ordres $(p, d, q)(P, D, Q)_S$ on p, d, q , són les ordres no estacionàries del model i P, D, Q són les ordres estacionàries, que fan referència als retards que s'afegeixen a l'ordre estacional S explicat anteriorment. Per captar de forma simultània la component regular i la component estacional, s'utilitza el model SARIMA $(p, d, q)(P, D, Q)_S$ definit com:

$$\phi(B)\Phi_s(B)(1-B)^d(1-B^S)^D X_t = \delta + \theta(B)\Theta_S(B)\varepsilon_t, \quad (3.6)$$

que inclou tres elements nous:

$$\Phi_P(B^S) = 1 - \Phi_1 B^S - \Phi_2 B^{2S} - \dots - \Phi_P B^{PS},$$

$$\Theta_Q(B^S) = 1 - \Theta_1 B^S - \Theta_2 B^{2S} - \dots - \Theta_Q B^{QS},$$

$$\nabla_S = 1 - B^S,$$

on la primera fórmula fa referència al polinomi $AR(P)_S$, la segona al polinomi $MA(Q)_S$ i finalment a l'operador de diferència estacional. Per poder determinar els valors de p, d, q del model ARIMA i P, D i Q del model ARIMA estacional, es realitza l'estudi de la funció d'autocorrelació i la funció d'autocorrelació parcial.

3.3.3. Estimació dels paràmetres i validació del model

Una vegada identificats els termes que conté el model s'estimen els paràmetres que el construeixen. Usualment, es realitza a través de paquets informàtics, com s'ha fet en la secció 4.5.1.

El principal criteri pel càlcul dels paràmetres és l'Estimació de màxima versemblança o *Maximum Likelihood*. S'ha demostrat que els paràmetres estimats a través d'aquest criteri reflecteixen amb gran exactitud les característiques presents en les dades de la sèrie temporal. Tot i això, també sorgeix l'ús del criteri de mínims quadrats. L'estimació dels paràmetres del model SARIMA es realitza a través de minimitzar la suma dels quadrats dels residus (SSR), aquests residus es defineixen com:

$$\hat{a}_t = Y_t - \hat{Y}_t,$$

on \hat{a}_t és el residual de la sèrie, Y_t és la sèrie original i \hat{Y}_t és la sèrie calculada amb els paràmetres estimats. Per tant, la suma dels quadrats dels residus que és la que s'ha de minimitzar, es defineix com:

$$SSR = \sum \hat{a}_t^2.$$

Cal tenir present que aplicar la modelització SARIMA a una sèrie temporal no és garantia que s'obtingrà un model que s'ajusti a la perfecció, en especial en aquelles sèries on existeix un gran nombre d'observacions. A més, acostuma a passar que per a una mateixa sèrie temporal, existeixin diversos models que compleixin amb les característiques del model adequat i proporcionin resultats similars.

Una vegada estimats els coeficients del model proposta, la següent etapa és l'avaluació o validació d'aquests. En aquest pas, es comprova l'eficiència del model i es decideix si és estadísticament adequat o no. Un model estadísticament adequat és aquell que els residus són independents entre si, és a dir, els residus són completament aleatoris. Pel que, si els residus mostren estar correlacionats entre ells, significa que existeix un patró que encara no s'ha tingut en compte pels termes autoregressius o de mitjanes mòbils del model proposta. Per tant, es conclou que si els residus estan correlacionats d'alguna manera, aquests no són soroll blanc i cal buscar un altre model on aquests siguin completament aleatoris.

La funció d'autocorrelació simple dels residus és l'instrument que s'utilitza per determinar si el model és estadísticament adequats, on el càlcul dels coeficients d'autocorrelació és:

$$r_k(\hat{a}) = \frac{\sum_{t=1}^{n-k} (\hat{a}_t - \bar{a})(\hat{a}_{t+k} - \bar{a})}{\sum_{t=1}^n (\hat{a}_t - \bar{a})^2}, \quad (3.7)$$

on $r_k(\hat{a})$ és el coeficient d'autocorrelació simple residual per un retard de k períodes, \bar{a} és la mitjana dels residus, \hat{a}_t és el residu en el període t , \hat{a}_{t+k} és el residu en el període amb k retassos i n és el nombre total de residus.

A través de l'anàlisi d'aquesta funció, es busca que els coeficients d'autocorrelació siguin zero o molt pròxims a ell. Específicament, es pretén que els coeficients no siguin significatius; això es determina a través de l'estadístic *Ljung-Box*, el qual enfoca l'anàlisi dels coeficients d'autocorrelació dels residus de manera grupal. És una prova que considera un conjunt de K coeficients d'autocorrelació. L'estadístic *Ljung-Box* es defineix com:

$$Q^* = n'(n' + 2) \sum_{k=1}^K (n' - k)^{-1} r_k(\hat{a}_t)^2, \quad (3.8)$$

on $n' = n - (d + LD)$, sent n el nombre d'observacions de la sèrie, d el grau de diferenciació regular, D el grau de diferenciació estacional, L el període estacional, $r_k(\hat{a}_t)^2$ és el quadrat de $r_k(\hat{a}_t)$ i K el nombre d'autocorrelacions.

El valor de Q^* ha de ser petit, tot i que, jutjar que el valor de Q^* és un valor petit, és subjectiu. És per això que cal utilitzar certes condicions per establir un punt de rebuig de la suficiència del model. Per això, s'empra una escala de la distribució *chi-quadrada*, la qual se li associa un valor p que representa la probabilitat de jutjar de manera correcta el valor de Q^* .

Es pot demostrar que si el valor de p és petit, el valor de Q^* és gran i el model no és adequat i viceversa.

De forma pràctica, si el valor de p és major que 0,01, però menor que 0,05, es considera que el model és inacceptable. Si el valor de p és major que 0,05 llavors es diu que és raonable concloure que el model és suficient.

3.3.4. Regions de predicció

El pronòstic d'una sèrie temporal es fa mitjançant les prediccions o estimacions puntuals. Així que, una predicció puntual \hat{Y}_t es defineix com el valor de la variable Y en el temps t , calculada del model SARIMA ajustat a la sèrie.

Cada pronòstic té un interval de confiança associat a la probabilitat que el valor futur observat es trobi dins d'aquest interval. Normalment, s'estableix que la probabilitat sigui del 95%. Aquest es calcula mitjançant l'equació:

$$[\hat{Y}_{n+t} \pm 1,96SE_{n+t}],$$

on \hat{Y}_{n+t} és el pronòstic en el període $(n+t)$ a partir de l'origen n i SE_{n+t} és l'error estàndard de l'error del pronòstic del període $(n+t)$.

3.4. Sèries temporals discretes

En aquest apartat, es presenten diversos models de sèries temporals discretes, un d'ells basat en l'operador *binomial thinning* que es definirà a l'apartat 3.4.1 i permetran assolir els objectius del treball.

3.4.1. Models INAR

Aquest capítol següent se centrarà en els models enters autoregressius (INAR); aquests són una extensió natural al cas discret dels models autoregressius (AR). La proposta més interessant sobre l'aplicació d'aquest model és en l'anàlisi de la incidència de certes malalties, pel fet que s'està modelant amb recomptes. El model INAR(p) ve definit per la següent fórmula:

$$X_t = p_1 \circ X_{t-1} + \dots + p_k \circ X_{t-k} + W_t, \quad (3.9)$$

on els paràmetres p_1, \dots, p_k són paràmetres fixos, $0 < p_1, \dots, p_k < 1$, i les innovacions W_t segueixen una distribució de Poisson amb una mitjana fixa λ_t . L'operador \circ de l'expressió 3.9, anomenat *binomial thinning* que ve definit com:

$$p_j \circ X_{t-j} = \sum_{i=1}^{X_{t-j}} Y_i, \quad (3.10)$$

on Y_i són variables aleatòries independents i idènticament distribuïdes que segueixen una Bernoulli amb una probabilitat d'èxit equivalent a p_j . Per tant, si $X_{t-j} = x_{t-j}$, llavors $p_j \circ x_{t-j}$ segueix una distribució Binomial, amb el nombre d'èxits com x_{t-j} [10].

3.4.2. Estimació dels paràmetres

Per a l'estimació dels paràmetres del model 3.9, $\theta = (\lambda_1, \dots, \lambda_t, p_1, \dots, p_k)$, s'ha utilitzat el mètode de màxima versemblança condicionada [10]. La funció de versemblança condicionada als valors de x_1, \dots, x_n es pot escriure com:

$$L(X; \theta) \sim \prod_{i=1}^n P(x_n | x_{n-1}, \dots, x_{n-k}; \theta). \quad (3.11)$$

Aquí, $P(x_i | x_{i-1}, \dots, x_{i-k}; \theta)$ indica la funció de probabilitat de X_i condicionada a les k observacions anteriors, avaluades en x_i .

Veient l'expressió del model 3.9, es percep que la distribució de probabilitat correspon a la suma de k variables aleatòries binomials independents amb paràmetres $(x_{i-1}, p_1), \dots, (x_{i-k}, p_k)$ i una variable aleatòria de Poisson amb mitjana $\lambda_1, \dots, \lambda_t$ dependent del mes al qual correspon l'observació [10].

La funció de probabilitat general pren la següent expressió, basada en [11]:

$$\begin{aligned} P(x_i | x_{i-1}, \dots, x_{i-k}) = & \sum_{k=0}^{\min(x_{i-1}, x_i)} \binom{x_{i-1}}{k} \cdot p_1^k \cdot (1-p_1)^{x_{i-1}-k} \\ & \dots \cdot \sum_{n=0}^{\min(x_{i-k}, x_i-k)} \binom{x_{i-k}}{n} \cdot p_k^n \\ & \cdot (1-p_k)^{x_{i-k}-n} \cdot \frac{e^{-\lambda_i} \cdot \lambda_i^{x_i-k-n}}{(x_i-k-n)!}. \end{aligned} \quad (3.12)$$

L'expressió 3.12 es generalitza a models d'ordre superior en el mateix article [11]. Els errors estàndard de les estimacions dels paràmetres s'han calculat a partir de la inversa de la matriu Hessiana corresponent.

La variància d'un estimador de màxima versemblança es calcula mitjançant la inversa de la matriu d'informació:

$$\text{var}(\theta) = [I_n(\theta)]^{-1},$$

on la matriu d'informació és el negatiu del valor esperat de la matriu Hessiana:

$$[I_n(\theta)] = -E(H(\theta)).$$

L'Hessiana és la matriu de segones derivades de la log-versemblança respecte als paràmetres:

$$H(\theta) = \frac{d^2 l(\theta)}{d\theta d\theta'},$$

pel que la matriu de variàncies i covariàncies de $\hat{\theta}$ és:

$$\text{var}(\theta) = [I_n(\theta)]^{-1} = (-E[H(\theta)])^{-1} = (-E[\frac{d^2 l(\theta)}{d\theta d\theta'}])^{-1}.$$

3.4.3. Regions de predicció a curt termini

Donades n observacions x_1, \dots, x_n , es vol conèixer la distribució de X_{n+k} , sent aquesta la distribució de les observacions k en el futur condicionada a valors previs. És a dir, si se suposés que $k = 1$, si es tingués en compte que X_{n+1} , condicionat a $X_n = x_n$ i $X_{n-1} = x_{n-1}$, és la suma de dues variables binomials independents amb paràmetres (x_n, p_1) i (x_{n-1}, p_2) , i una variable independent amb distribució de Poisson amb mitjana λ_{n+1} es pot obtenir la distribució de X_{n+1} fent servir la Funció de probabilitat 3.12.

Pel que fa a el càlcul de la distribució de X_{n+2} ja es complica més la cosa. Primerament, s'expressa X_{n+2} en termes de valors ja coneguts com són X_n i X_{n-1} , a partir de l'operador *binomial thinning* i àlgebra [10]:

$$\begin{aligned} X_{n+2} &= p_1 \circ X_{n+1} + p_2 \circ X_n + W_{n+2} \\ &= p_1 \circ (p_1 \circ X_n + p_2 \circ X_{n-1} + W_{n+1}) + p_2 \circ X_n + W_{n+2} \\ &= (p_1^2 + p_2) \circ X_n + p_1 + p_2 \circ X_{n-1} + p_1 \circ W_{n+1} + W_{n+2}. \end{aligned} \quad (3.13)$$

És important observar que $p_1 \circ W_{n+1} + W_{n+2}$ segueix una distribució de Poisson amb mitjana $p_1 \lambda_{n+1} + \lambda_{n+2}$, que a partir d'ara s'escriurà com $W(\lambda_2^*)$. També, condicionant que $X_n = x_n$ i $X_{n-1} = x_{n-1}$, que és el que es troba en l'equació 3.13 és la suma de dues variables binomials independents amb paràmetres $(x_n, p_1^2 + p_2)$ i $(x_{n-1}, p_1 p_2)$, i una variable de Poisson de mitjana $p_1 \lambda_{n+1} + \lambda_{n+2}$. Per tant, una altra vegada es pot utilitzar la funció de probabilitat 3.12 per obtenir la distribució de X_{n+2} , amb els nous paràmetres $p_1^2 + p_2, p_1 p_2$ i $p_1 \lambda_{n+1} + \lambda_{n+2}$.

La distribució de X_{n+k} per qualsevol $k > 2$, es pot obtenir de la mateixa manera, després d'expressar X_{n+k} de la següent manera [10]:

$$X_{n+k} = f_k \circ X_n + g_k \circ X_{n-1} + W(\lambda_k^*), \quad (3.14)$$

on $f_1 = p_1, g_1 = p_2, \lambda_1^* = \lambda_{n+1}, f_2 = p_1^2 + p_2, g_2 = p_1 p_2$ i finalment $\lambda_2^* = p_1 \lambda_{n+1} + \lambda_{n+2}$. A més, es pot veure immediatament que les següents relacions recursives es compleixen per $k > 1$:

$$\begin{aligned} f_{k+1} &= p_1 f_k + p_2 f_{k-1}, \\ g_{k+1} &= p_1 g_k + p_2 g_{k-1}, \\ \lambda_{k+1}^* &= p_1 \lambda_k^2 + p_2 \lambda_{k-1}^* + \lambda_{n+k+1}. \end{aligned} \quad (3.15)$$

Gràcies a les relacions recursives i utilitzant de nou l'expressió 4.3, es pot calcular fàcilment la distribució de $X_{n+k}, k > 1$. Si se substitueix els paràmetres de les relacions recursives 3.15 per les seves estimacions, es podrà calcular les regions de predicció aproximades de mida $1 - \alpha$, trobant els límits inferior i superior l_1 i l_2 que compleixen les següents expressions:

$$\sum_{j=0}^{l_1} P(X_{n+k} = j) \approx \alpha/2, \quad \sum_{j=l_1}^{l_2} P(X_{n+k} = j) \approx 1 - \alpha. \quad (3.16)$$

Capítol 4

Resultats

En aquest capítol es presenten els resultats que s’han obtingut en l’anàlisi descriptiva de les variables, en aquest cas són les regions sanitàries descrites en l’Apartat 2.2 d’aquest treball. A més, també es presentaran els diferents models de sèries temporals adaptats a cada regió sanitària, els resultats que s’han assolit, així com les interpretacions dels paràmetres d’aquests i les prediccions.

4.1. Anàlisi descriptives

Abans de l’elaboració dels diferents models de sèries temporals que s’han decidit estudiar en aquest treball, s’ha fet una anàlisi descriptiva amb la finalitat de poder veure i analitzar com es comporten les diferents regions sanitàries durant els anys a partir de la visualització de les sèries temporals.

Cal tenir en compte que, tot i les dades que s’analitzen estan recollides entre el juny del 2010 i juny del 2021, en aquesta anàlisi descriptiva que es mostra a continuació, s’ha representat el comportament de la sèrie temporal a partir del 2014 fins al 2021, sense tenir en compte els anys anteriors al 2014, ja que representar-los només haurien reafirmat les corbes que mostren la resta d’anys, pel que s’ha decidit no afegir-les. Aquesta gràfica té en compte el nombre de casos de contagi de la grip per cada mes i per cada any, tot i que realment el recompte de casos en la base de dades és per cada setmana, pel fet que aquesta és la unitat de temps utilitzada habitualment pels sistemes de vigilància epidemiològica de la grip.

Les primeres sèries que es mostren a la Figura 4.1 fan referència a les dues regions sanitàries de la província de Tarragona, d'esquerra a dreta, Camp de Tarragona i Terres de l'Ebre. Pel que fa a la Regió Sanitària de Tarragona s'observa que el pic més alt de contagis que han registrat és de 4499 al mes de gener i les Terres de l'Ebre és de 2084 contagis, coincidint també al gener. Durant la resta de mesos, ambdues regions mostren una clara disminució de casos que a mesura que se'n van els mesos de tardor i s'aproximen al novembre, desembre els casos tornen a augmentar, tot i no arribar als valors de gener. Cal tenir en compte que la densitat de població no és la mateixa, ja que Camp de Tarragona agrupa sis comarques de la província (Alt Camp, Baix Camp, Baix Penedès, Conca de Barbera, Priorat i Tarragonès) i Terres de l'Ebre n'agrupa quatre (Baix Ebre, Montsià, Ribera d'Ebre i Terra Alta). És per això, que s'estudia cada regió sanitària per separat per culpa de les seves densitats de població diferents que presenten.

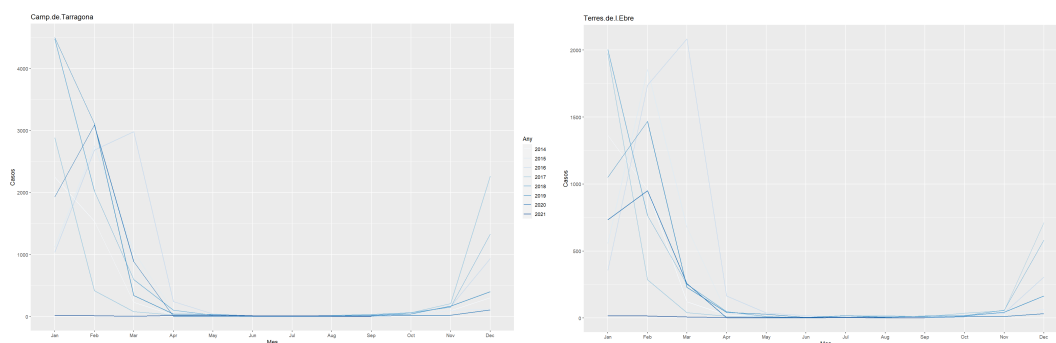


Figura 4.1: Representació gràfica del nombre de casos detectats en les Regions Sanitàries de Camp de Tarragona i Terres de l'Ebre

Les següents dues sèries de la Figura 4.2 representen d'esquerra a dreta, les Regions sanitàries de l'Alt Pirineu i Aran i Lleida. La primera regió comentada agrupa sis comarques Pirinenques: Alt Urgell, Alta Ribagorça, Cerdanya, Pallars Jussà, Pallars Sobirà i Vall d'Aran i, tot i que la regió abasta el 18% del territori de Catalunya, els seus residents habituals tot just representen el 0,9% de la població catalana [12]. La Regió Sanitària de Lleida inclou sis comarques de la mateixa província: Garrigues, Noguera, Pla de l'Urgell, Segarra, Segrià i Urgell.

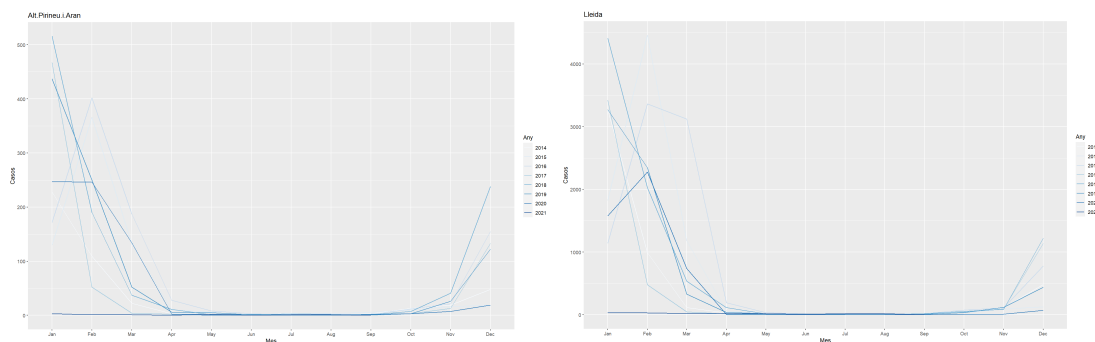


Figura 4.2: Representació gràfica del nombre de casos detectats en les Regions Sanitàries de l'Alt Pirineu i Aran i Lleida

Pel que fa a l'Alt Pirineu i Aran el valor de contagis més alt de tots els anys representats és de 516 al gener i a Lleida de 4461 i ambdues mostren gràfiques similars a les vistes anteriorment.

Seguidament, en la Figura 4.3 es veuen representades d'esquerra a dreta, les Regions Sanitàries de Girona i Catalunya Central. La Regió Sanitària de Girona agrupa set comarques de la província de Girona: Alt Empordà, Baix Empordà, Garrotxa, Gironès, Pla de l'Estany, Ripollès i Selva. També agrupa alguns municipis del Maresme: Calella, Canet de Mar, Malgrat de Mar, Palafolls, Pineda de Mar, Sant Cebrià de Vallalta, Sant Iscle de Vallalta, Sant Pol de Mar, Santa Susanna i Tordera. Pel que fa a la Regió Sanitària de Catalunya Central agrupa sis de les comarques centrals de Catalunya: Anoia, Bages, Berguedà, Moianès, Osona i Solsonès.

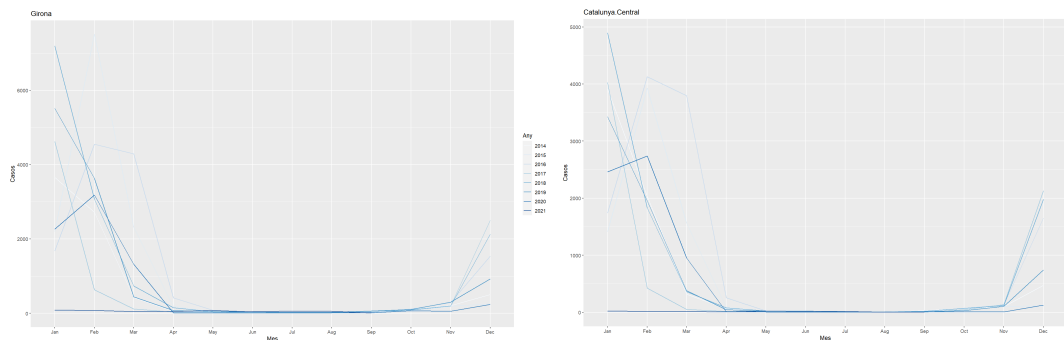


Figura 4.3: Representació gràfica del nombre de casos detectats en les Regions Sanitàries de Girona i Catalunya Central

Totes dues regions mostren valors molt elevats en els contagis de la grip en els mesos de gener: 7507 i 4896, respectivament i com era d'esperar, també dibuixen sèries temporals molt similars a les anteriors regions.

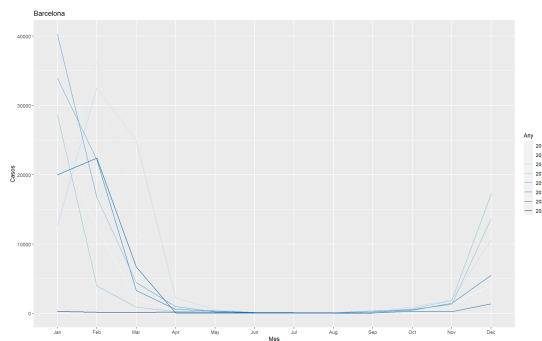


Figura 4.4: Representació gràfica del nombre de casos detectats a la Regió Sanitària de Barcelona

Finalment, la Figura 4.4 representa la Regió Sanitària amb més habitants, ja que agrupa tres àmbits: Àmbit Metropolità Nord, Sud i Barcelona Ciutat, i per això el nombre de casos de la grip és molt més elevats que en les altres regions. El màxim nombre de casos de grip declarats al gener és de 40306.

En definitiva, es pot observar que tot i haver-hi diferències en el nombre d'habitats de les diferents regions sanitàries, la grip segueix un comportament bastant similar durant els anys; sabent que és una malaltia que acostuma a aparèixer en forma epidèmica un l'any, generalment a l'hivern, té una major incidència en determinants moments de l'any.

4.2. Identificació del model

Per poder determinar el tipus de model i ordre, s'ha analitzat els coeficients d'autocorrelació i els coeficients d'autocorrelació parcials de cada Regió Sanitària excepte Barcelona, aquesta es tractarà a part degut al seu elevat nombre de contagis. Aquests gràfics indiquen el grau de correlació que tenen les observacions de la sèrie. Per veure si existeix correlació, cal veure quants coeficients són superiors a 0, pel fet que aquests seran els que ens diran el grau de correlació entre les observacions. En el cas d'aquest estudi, serà el nombre de setmanes anteriors que afecten el nombre de contagis de la setmana analitzats.

Sabent que els contagis de la grip d'una setmana depenen dels casos detectats en les setmanes anteriors, s'ha considerat com a més indicat els models autoregressius. Aquests es caracteritzen per tenir un ACF on el nombre de coeficients estadísticament diferents de 0, decreixen de manera exponencial cap al 0, i uns PACF amb p coeficients superiors a 0, els quals indiquen l'ordre del model.

4.2.1. Coeficients d'autocorrelació (ACF)

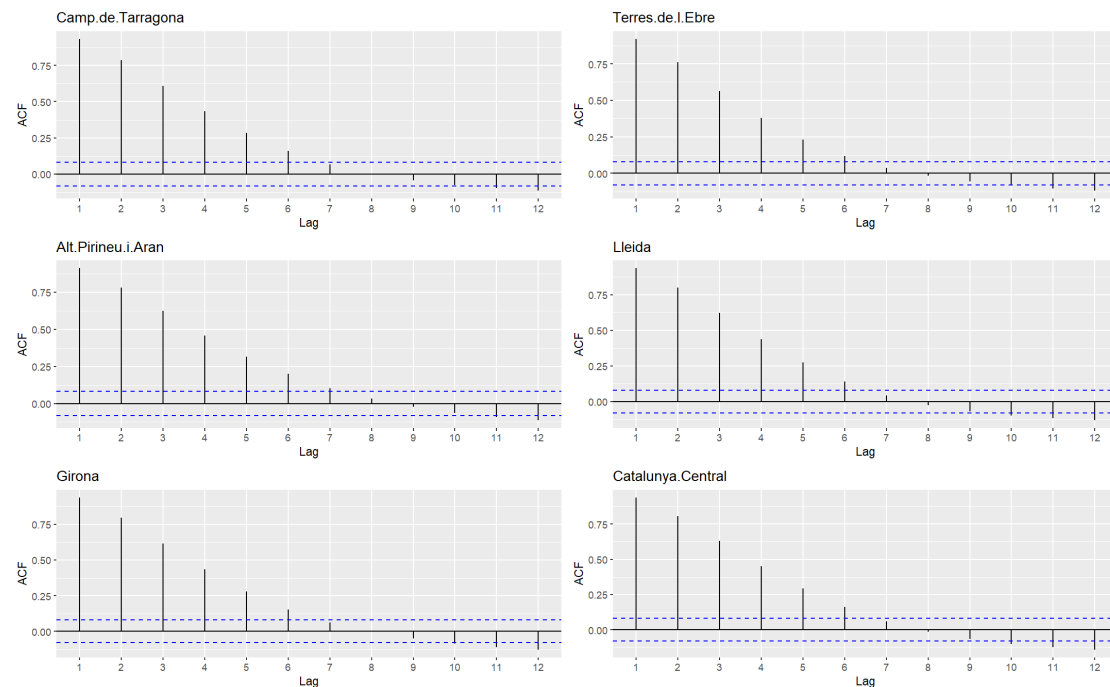


Figura 4.5: ACF de les diferents regions sanitàries de Catalunya

Si s'observa la Figura 4.5 es pot veure com en totes les Regions Sanitàries, els coeficients d'autocorrelació tendeixen exponencialment al zero al llarg del temps. És a dir, es pot veure com hi ha una correlació significativa i que va disminuint al llarg del temps, per la qual cosa aquest patró està indicant un terme autoregressiu. Per a poder-ho determinar es realitza la funció d'autocorrelació parcial per determinar l'ordre del terme autoregressiu.

4.2.2. Coeficients d'autocorrelació parcials (PACF)

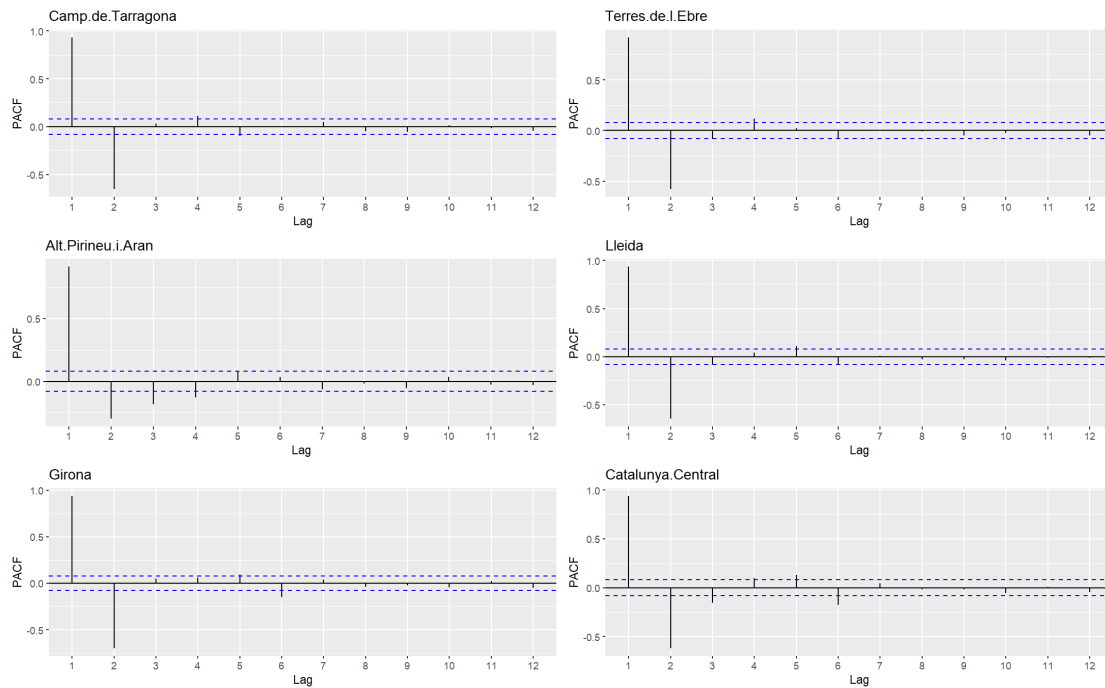


Figura 4.6: PACF de les diferents regions sanitàries de Catalunya

Pel que fa als PACF de les diferents regions sanitàries de Catalunya, els dos primers coeficients són estadísticament diferents de zero. Aquests dos conjunts de gràfics suggereixen que hi ha correlació d'ordre més alt que 1, exactament 2, i per això es provarà d'ajustar les diferents dades de les regions sanitàries per models INAR d'ordre 2: INAR(2), excepte per la regió de Barcelona que s'estimarà per un model SARIMA.

Per tal d'incloure l'estacionalitat en el model, s'explorarà la variació dels models definits a l'apartat 3.4.1, que diu que sent X_t una sèrie temporal discreta, segueix un model INAR(2) amb estacionalitat d'ordre 2 si satisfà

$$X_t = p_1 \circ X_{t-1} + p_2 \circ X_{t-2} + W_t(\lambda_t), \quad (4.1)$$

on els paràmetres p_1 i p_2 són com a l'equació 3.9, però assumint que les innovacions W_t segueixen distribucions diferents de Poisson amb mitjanes diferents de manera que $\lambda_t = \lambda_{t+\tau}$, on τ representa la periodicitat observada.

Cal recordar que la definició de *binomial thinning*,

$$p_j \circ X_{t-j} = \sum_{i=1}^{X_{t-j}} Y_i,$$

on Y_i són variables aleatòries de Bernoulli independents i idènticament distribuïdes amb probabilitat d'èxit igual a p_j , en el context d'aquest Treball Final de Grau, X_t representa el nombre de contagis de la grip a temps t . La variable aleatòria $Y_i, i = 1, \dots, x_{t-j}$ és igual a 1 si l'individu i , que es contagia a la setmana $t-j$, continua estant contagiada a la setmana j , o 0 si no es compleix aquesta condició. Pel que fa a els recomptes W_t es podrien interpretar com les innovacions o els nous casos de gripes en temps t .

Tot i que la unitat de temps de les dades és a la setmana, si s'utilitzés un model complet, s'hauria de considerar tantes mitjanes λ_t diferents per a les innovacions com setmanes hi hagués a l'any. Com que això implicaria estimar un nombre excessivament gran de paràmetres, s'ha assumit una estacionalitat o periodicitat de dotze mesos ($\tau = 12$) a les dades del treball, fent que les mitjanes λ_t siguin iguals per a qualsevol setmana dins del mateix mes. Pel que hi haurà només 12 λ_t diferents, el que significa un total de 14 paràmetres en el cas del model INAR(2), que això és, $\lambda_1, \dots, \lambda_{12}, p_1, p_2$.

Per tant, les dades d'aquest Treball Final de Grau s'han ajustat fent servir el model definit a 4.1 que aquest ha estat satisfactori, fet que es comentarà amb més profunditat a l'apartat 4.3.1.

4.3. Estimació dels paràmetres i validació de la proposta

Una vegada es té coneixença del model amb el qual es treballarà, s'estima els paràmetres del model 4.1, INAR(2), $\theta = (\lambda_1, \dots, \lambda_{12}, p_1, p_2)$, que s'ha utilitzat el mètode de màxima versemblança condicionada [10]. Cal recordar que la funció de versemblança condicionada als primers dos valors de x_1 i x_2 es pot escriure com

$$L(X; \theta) \sim P(x_3|x_1, x_2; \theta) \cdot P(x_4|x_2, x_3; \theta) \dots P(x_n|x_{n-1}, x_{n-2}; \theta), \quad (4.2)$$

perquè s'està considerant només dependències d'ordres 1 i 2. Aquí, $P(x_i|x_{i-1}, x_{i-2}; \theta)$ indica la funció de probabilitat de X_i condicionada a les dues observacions anteriors, avaluades en x_i .

Veient l'expressió del model 4.1, es percep que la distribució de probabilitat correspon a la suma de dues variables aleatòries binomials independents amb paràmetres (x_{i-1}, p_1) i (x_{i-2}, p_2) i una variable aleatòria de Poisson amb mitjana $\lambda_1, \dots, \lambda_{12}$ dependent del mes al qual correspon l'observació [10].

Per poder calcular la funció de probabilitat s'ha utilitzat la següent expressió, basada en [11]:

$$P(x_i|x_{i-1}, x_{i-2}) = \sum_{k=0}^{\min(x_{i-1}, x_i)} \binom{x_{i-1}}{k} \cdot p_1^k \cdot (1-p_1)^{x_{i-1}-k} \cdot \sum_{n=0}^{\min(x_{i-2}, x_i-k)} \binom{x_{i-2}}{n} \cdot p_2^n \cdot (1-p_2)^{x_{i-2}-n} \cdot \frac{e^{-\lambda_i} \cdot \lambda_i^{x_i-k-n}}{(x_i-k-n)!}. \quad (4.3)$$

L'expressió 4.3 es generalitza a models d'ordre superior en el mateix article [11]. La maximització de la funció de probabilitat 4.2 s'ha dut a terme amb un programa desenvolupat en *R*. Aquest programa utilitza el procediment de minimització no lineal anomenat *nlm*. Els errors estàndard de les estimacions dels paràmetres s'han calculat a partir de la inversa de la matriu Hessiana corresponent.

Cal tenir present que s'ha decidit no aplicar la fórmula 4.3 sobre el nombre de casos de la grip, sinó aplicar-la sobre la incidència de casos per habitants (dependent de la regió sanitària un nombre d'habitants diferents) i d'aquesta manera, evitar la influència que pot tenir la població de cada Regió Sanitària. El fet de treballar amb la incidència provoca que en la funció 4.3 caldrà multiplicar totes les λ_t per $\frac{Pop.annual}{N}$, tenint en compte que *Pop.Annual* que és la població anual serà 10 valors, el valor de la població des del 2010 fins l'any 2019 de cada Regió Sanitària i *N* que és el nombre d'habitants dependrà de la regió sanitària que s'estudia.

A continuació, a la següent Taula 4.1 es mostren els valors de les poblacions assegurades des de l'any 2010 fins a l'any 2019 de totes les regions sanitàries excepte per Barcelona. Les persones assegurades són aquelles que estan registrades en el Servei Català de la Salut que formen part d'un fitxer automatitzat denominat Registre central d'assegurats del CatSalut (RCA).

Regió Sanitària	N	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
Tarragona	60.000	593.241	593.997	596.328	595.836	596.508	593.391	592.010	591.625	594.861	600.603
Terres de l'Ebre	25.000	190.241	188.573	187.359	186.010	184.503	182.070	179.873	178.159	177.057	176.488
Alt Pirineu i Aran	60.000	69.977	69.457	68.688	68.652	68.417	67.632	67.084	66.833	66.867	66.813
Lleida	45.000	365.679	367.748	368.086	366.469	364.568	361.085	358.998	358.062	358.070	359.558
Girona	61.000	844.942	845.943	845.142	843.135	844.771	839.960	837.042	838.103	843.268	850.699
Catalunya Central	50.000	516.857	517.831	517.477	516.557	515.213	511.707	510.619	512.050	514.646	520.104

Taula 4.1: Nombre de sanitaris de les Regions Sanitàries

La Taula 4.2 mostra les estimacions obtingudes per totes les regions sanitàries excepte per la Regió Sanitària de Barcelona. Recordar que cada λ_t equival al mes corresponent del seu valor de t , és a dir, gener és λ_1 i així amb la resta de mesos. També, mostra els errors estimats de les estimacions aconseguits per totes les regions sanitàries excepte per a la Regió Sanitària de Barcelona.

Cada Regió Sanitària té dues files: la primera representa les estimacions i la segona mostra els errors estimats dels paràmetres de màxima versemblança.

Paràmetre	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8	λ_9	λ_{10}	λ_{11}	λ_{12}	p_1	p_2
Tarragona	13,25	16,29	1,97	0,34	0,07	0,10	0,05	0,08	0,31	0,39	1,11	4,76	0,62	0,27
	0,82	1,46	0,45	0,17	0,07	0,06	0,03	0,05	0,10	0,15	0,24	0,39	0,01	0,01
Terres de l'Ebre	6,24	12,44	1,56	0,20	0,09	0,09	0,06	0,06	0,14	0,20	0,54	2,53	0,60	0,30
	0,56	1,02	0,39	0,13	0,06	0,05	0,04	0,04	0,06	0,08	0,16	0,29	0,01	0,01
Alt Pirineu i Aran	11,69	12,94	2,78	0,74	0,50	0,29	0,17	0,28	0,52	1,25	1,87	7,10	0,57	0,29
	0,88	1,14	0,47	0,21	0,15	0,10	0,07	0,09	0,12	0,20	0,28	0,47	0,01	0,01
Lleida	14,71	15,50	1,72	0,21	0,13	0,03	0,02	0,03	0,26	0,62	0,86	5,58	0,62	0,28
	0,91	1,44	0,44	0,18	0,07	0,03	0,02	0,03	0,09	0,17	0,25	0,42	0,01	0,01
Girona	14,04	14,49	2,49	0,53	0,17	0,11	0,11	0,04	0,33	0,64	1,04	4,82	0,63	0,29
	0,88	1,55	0,52	0,22	0,10	0,06	0,06	0,04	0,10	0,19	0,27	0,41	0,01	0,01
Catalunya Central	13,17	13,13	2,29	0,11	0,06	0,11	0,05	0,05	0,32	0,53	0,85	6,68	0,63	0,28
	0,93	1,53	0,49	0,18	0,06	0,06	0,04	0,04	0,10	0,17	0,23	0,46	0,01	0,01

Taula 4.2: Estimadors i Error Est. dels paràmetres de màxima versemblança

Primerament, en la Taula 4.2 es pot observar que les innovacions de mitjana més alta en totes les regions sanitàries corresponen als mesos de gener, febrer i desembre sent (λ_1 , λ_2 i λ_{12}). De fet, en totes les regions, $\hat{p}_1 > \hat{p}_2$, indicant l'impacte de valor corresponent a la setmana anterior és més gran que l'impacte del valor corresponent a dues setmanes anteriors. És important destacar que tots els paràmetres són significativament diferents de zero excepte els mesos de juny, juliol i agost λ_6 , λ_7 i λ_8 que en totes les regions mostren valors molt pròxims a zero.

Gràcies a aquestes estimacions, reforcen la hipòtesi del fet que els casos de contagi de la grip apareixen en temporada d'hivern, però cal tenir present que el que es pretén veure és com la Covid-19 va influenciar i si va modificar el comportament de la grip en l'any 2019.

4.3.1. Validació de la proposta

Una vegada s'ha trobat el possible model, cal validar-lo, comprovar que realment compleix amb les condicions necessàries per poder afirmar que aquest model explica la sèrie i, per tant, poder aplicar-lo per intentar predir els valors futurs de la sèrie. Per a dur a terme aquesta validació de la proposta, s'ha realitzat prediccions per l'any 2011, ja que es pretén comprovar que en un període on s'espera observar el comportament clàssic de la grip, no es detecta cap alerta. Per això, es duen a terme prediccions per la temporada gripal 2011-2012.

Totes les gràfiques de les diferents regions sanitàries excepte Barcelona, ja que aquesta es tractarà més endavant, presenten la mateixa llegenda: la línia contínua de color negre dibuixa la mediana de casos setmanes de grip, les línies negres discontinues, mostren els límits superiors i inferiors i finalment la línia de color vermell representa els casos registrats.

Les primeres figures mostrades, la figura 4.7 fa referència a la Regió de predicció de Tarragona i la figura 4.8 fa referència a les Terres de l'Ebre. Ambdues mostren com els casos registrats estàn a dins dels intervals de confiança, ja que en els mesos de febrer i març, que és quan hi va haver el pic més alt de contagis de grip l'any 2012, el model prediu correctament.

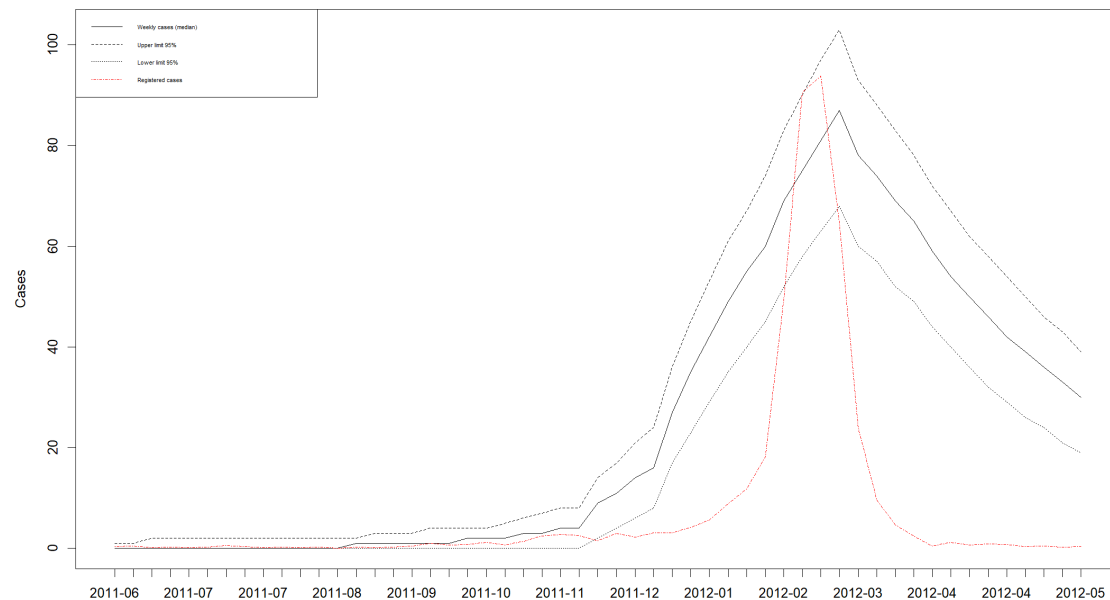


Figura 4.7: Regió de predicció per la temporada 2011-2012 de Tarragona

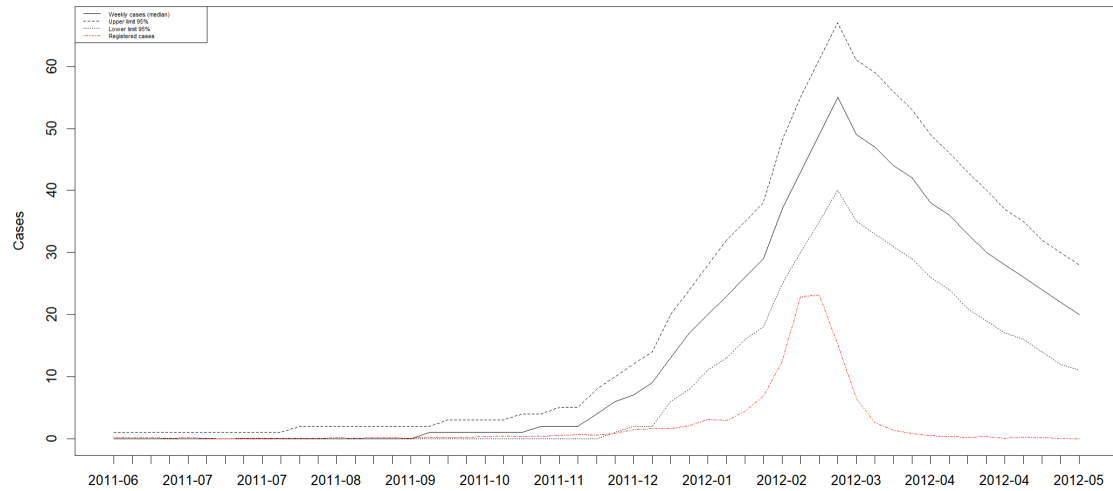


Figura 4.8: Regió de predicció per la temporada 2011-2012 de les Terres de l'Ebre

La figura 4.9 mostra la Regió Sanitària de l'Alt Pirineu i Aran. En aquest cas, el model hauria predit el brot de grip, de la mateixa manera que mostren les dades, i per això es pot donar per bona la validació en aquesta Regió Sanitària.

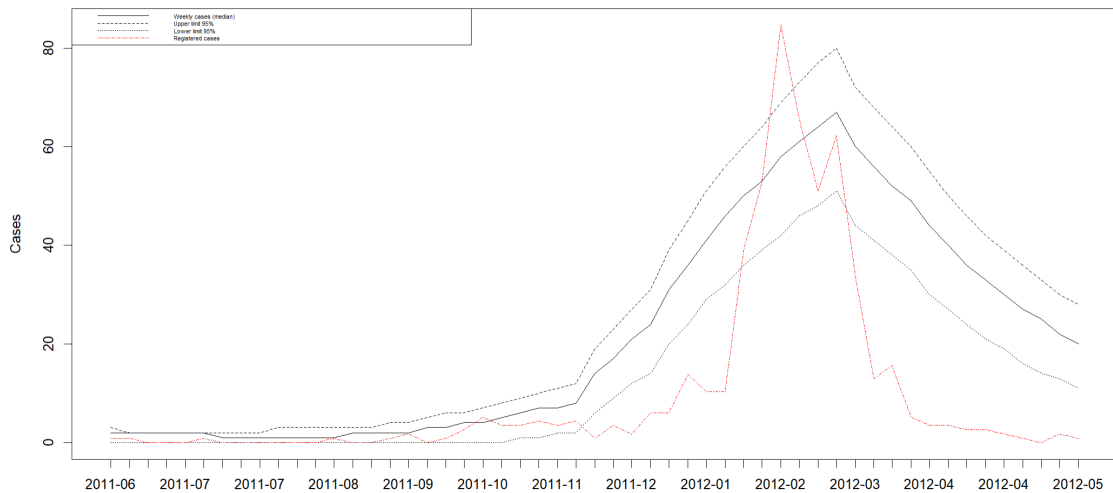


Figura 4.9: Regió de predicció per la temporada 2011-2012 de l'Alt Pirineu i Aran

En el cas de la figura 4.10 que mostra la Regió Sanitària de Lleida, la línia vermella que és la que representa els casos registrats, sobresurt lleugerament dels intervals de confiança en el pic més alt de contagis, en la temporada d'hivern especialment entre el mes de febrer i març.

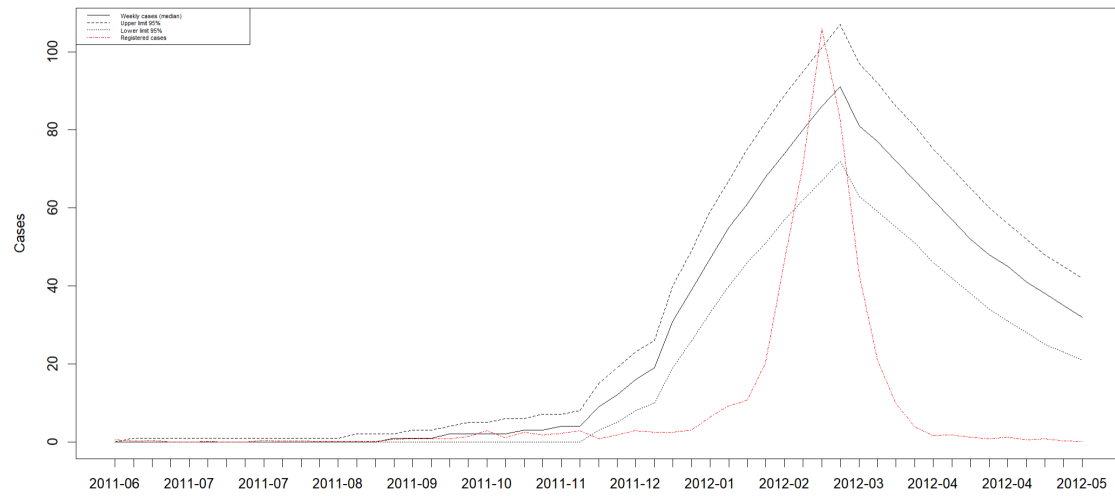


Figura 4.10: Regió de predicció per la temporada 2011-2012 de Lleida

La figura 4.11 que mostra clarament com el model s'adapta perfectament als valors del 2011-2012, any en el qual no va haver-hi cap anomalia pel que fa a malalties infeccioses.

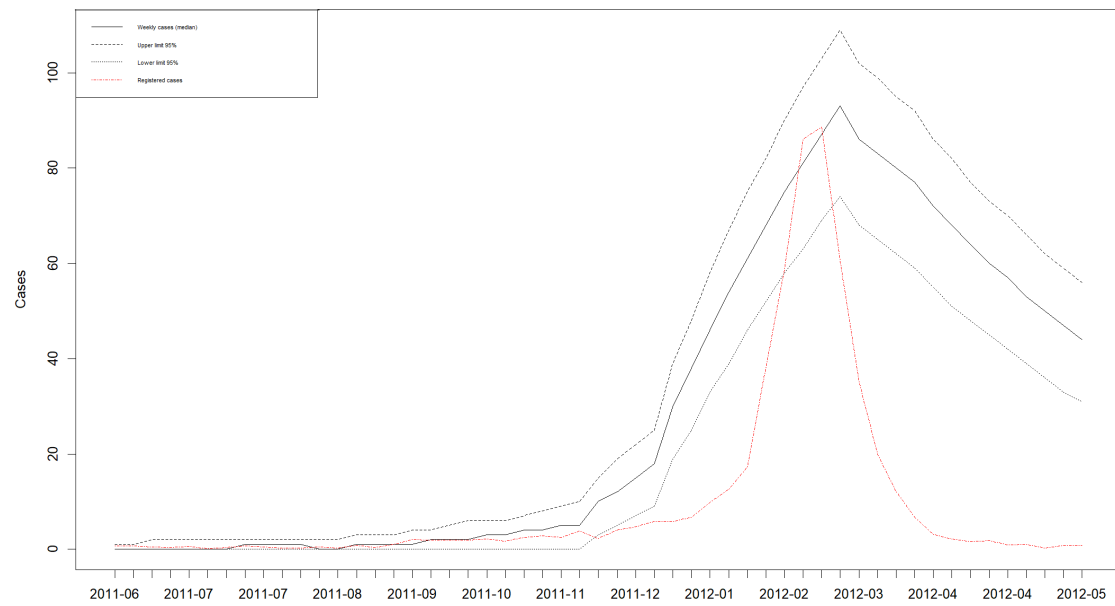


Figura 4.11: Regió de predicció per la temporada 2011-2012 de Girona

Finalment, la següent figura 4.12 representa la regió de predicció per a la temporada 2011-2012 de la Regió Sanitària de Catalunya Central. Una vegada més, el nombre de casos registrats està dins dels intervals de confiança, i per això queda demostrat que el model s'adapta perfectament a les dades independentment de si aquell any hi ha hagut afectacions sanitàries; es pot donar per vàlid el model proposat.

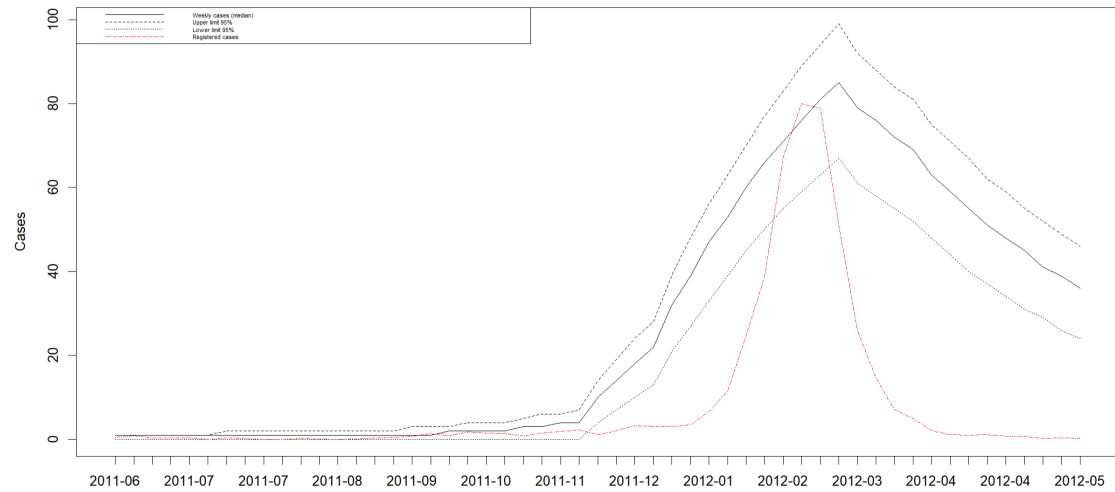


Figura 4.12: Regió de predicció per la temporada 2011-2012 de Catalunya Central

4.4. Prediccions

En aquesta secció es realitzarà les prediccions basades en el model 4.1, considerant aquí un any amb el nombre de 53 setmanes, com que s'estudiarà des de la setmana 23 de l'any 2019 fins a la setmana 22 de l'any següent 2020. Es podria haver agafat un any amb un total de quaranta-vuit setmanes, però per ser més exactes era més adequat fer-ho segons els anys epidemiològics. A més, el mètode que s'aplica és vàlid per portar a cap prediccions, ja que té en compte l'estructura temporal, afegint setmanes amb els paràmetres λ_j apropiats en els mesos corresponents [10].

4.4.1. Regions de predicció a curt termini

Com s'ha explicat en la secció 3.4.3, tenint n observacions x_1, \dots, x_n de contagis de grip, es vol conèixer la distribució de X_{n+k} , això és, la distribució de contagis k setmanals en el futur condicionada a valors previs. Per tant, se sap que X_{n+1} és la suma de dues variables binomials independents amb paràmetres (x_n, p_1) i (x_{n-1}, p_2) , i una variable independent amb distribució de Poisson amb mitjana λ_{n+1} .

Respecte a la distribució de X_{n+2} , descrita també en la secció 3.4.3 l'equació 3.13. Recordar que $p_1 \circ W_{n+1} + W_{n+2}$ segueix una distribució de Poisson amb mitjana $p_1 \lambda_{n+1} + \lambda_{n+2}$, que a partir d'ara s'escriurà com $W(\lambda_2^*)$. També, condicionant que $X_n = x_n$ i $X_{n-1} = x_{n-1}$, que és el que es troba en l'equació 3.13 que és la suma de dues variables binomials independents amb paràmetres $(x_n, p_1^2 + p_2)$ i $(x_{n-1}, p_1 p_2)$, i una variable de Poisson de mitjana $p_1 \lambda_{n+1} + \lambda_{n+2}$. Per tant, una altra vegada es pot utilitzar la funció de probabilitat 4.3 per obtenir la distribució de X_{n+2} , amb els nous paràmetres $p_1^2 + p_2, p_1 p_2$ i $p_1 \lambda_{n+1} + \lambda_{n+2}$ com s'ha dit en el capítol anterior.

Per mostrar les prediccions de totes les regions sanitàries de Catalunya excepte de Barcelona, s'ha ajustat el model estimat a partir de les observacions de 06-06-2010, setmana 23 de l'any 2010 fins al 26-05-2019, setmana 22 de l'any 2019, ja que es pretén predir les dades des del 02-06-2019, setmana 23 de l'any 2019 fins a la setmana 22 del 2019 que és 24-05-2020, per poder veure si les dades amb aquest model haguessin predit una certa anomalia amb l'entrada de la Covid-19 en la societat catalana. Cal tenir en compte, que a finals del desembre del 2019, principis del 2020, es van confondre molts casos de grip amb Covid-19 i a la inversa, fins que a l'inici de la pandèmia declarada al març, gairebé tots els casos de grip també es diagnosticaven com a Covid-19 a causa de les seves semblances en els símptomes que presenten. Aquests càlculs de regions de predicció amb els límits inferiors i superiors, s'han calculat a partir de les expressions 3.16. Les gràfiques segueixen la llegenda anterior: la línia contínua de color negre dibuixa la mediana de casos setmanes de grip, les línies negres discontinues, mostren els límits superiors i inferiors i finalment la línia de color vermell representa els casos registrats.

La figura 4.13 mostra com la mitjana de casos registrats té el pic més alt en els mesos de febrer i març, mentre que el model dibuixa la corba de contagis un parell de mesos abans, per fora dels intervals de confiança, pel que fa a saltar les alarmes que aquí està passant alguna cosa. En aquest cas, el brot de grip se situa a principis de gener.

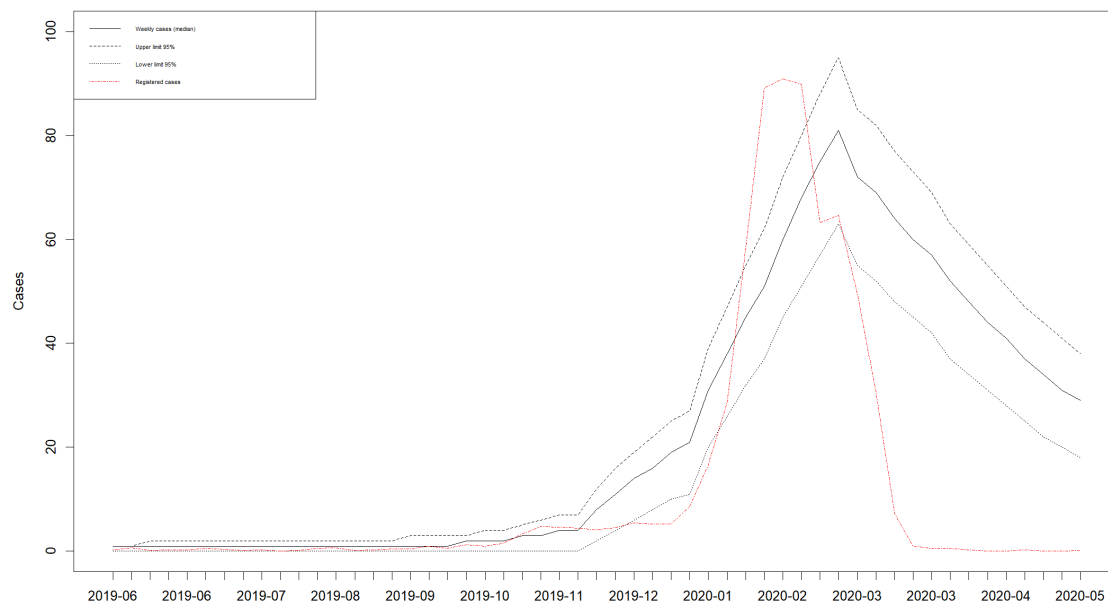


Figura 4.13: Regions de predicció per la temporada 2019-2020 a Tarragona

Pel que fa a la Regió Sanitària de les Terres de l'Ebre, 4.14, el model prediu que l'inici d'un gran nombre de contagis apareix a finals de 2019, amb un alt nombre de contagis, assolint el pic a mitjans de gener, mentre que les dades van detectar a finals de febrer. El pic no és tan dràstic com en altres regions sanitàries, però, tot i això, el model hauria alertat igualment d'un canvi en la sèrie temporal.

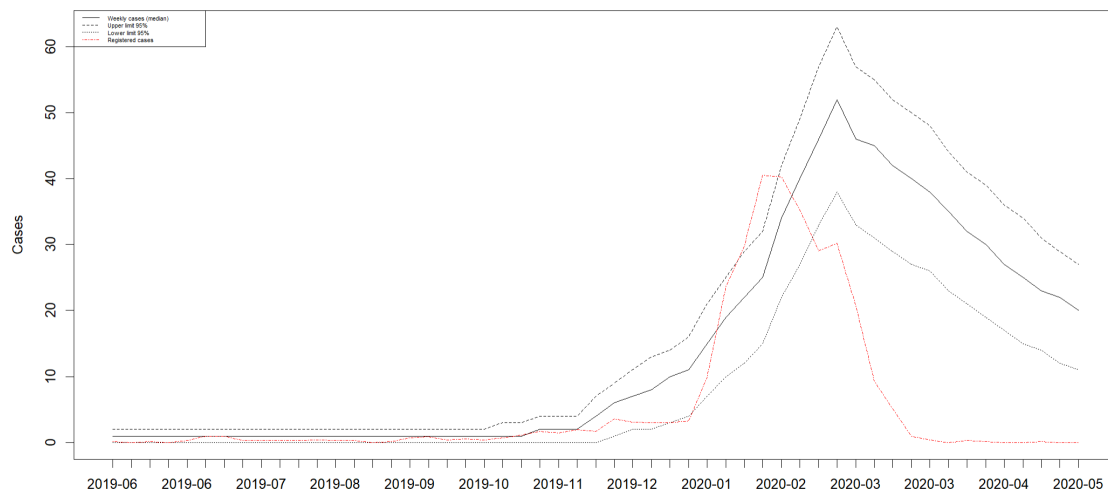


Figura 4.14: Regions de predicció per la temporada 2019-2020 a les Terres de l'Ebre

En el cas de la Regió Sanitària de l'Alt Pirineu i Aran, el model mostra com el nombre de contagis va elevant-se a inicis de desembre del 2019, tot i començar l'increment abans que les altres regions, no se sobresurt dels intervals de confiança fins a arribar al pic més alt de casos, a mitjans de gener, i per això realment el model hauria avisat al mateix temps que les altres regions.

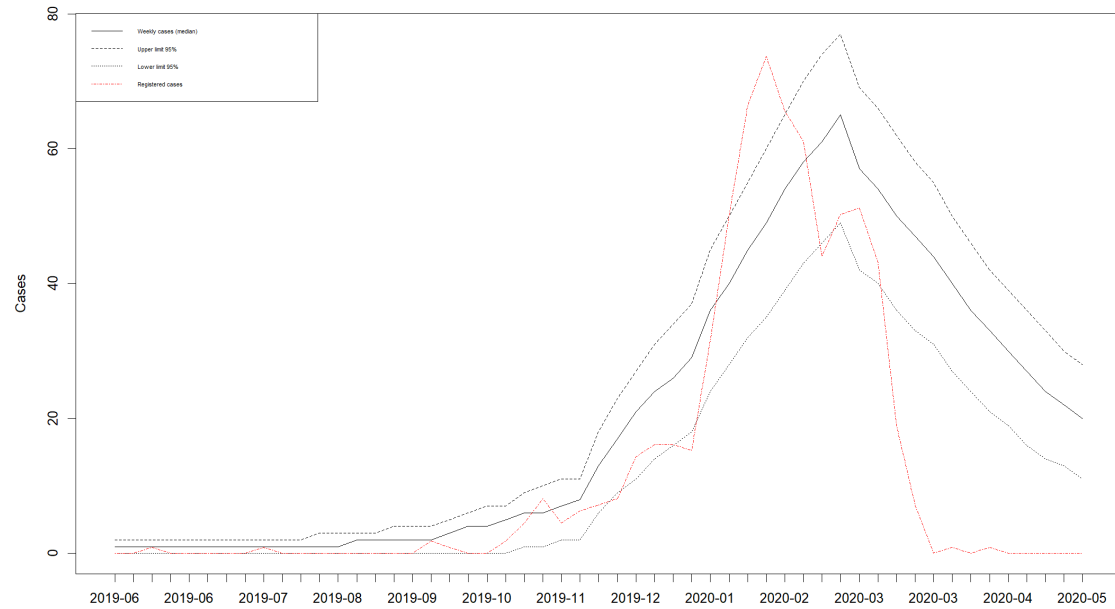


Figura 4.15: Regions de predicció per la temporada 2019-2020 a l'Alt Pirineu i Aran

La Regió Sanitària de Lleida, a diferència de l'última Regió Sanitària que s'ha vist, l'increment de casos de grip no es dispara fins a finals de gener del 2020, és a dir, que durant els mesos de desembre del 2019 que en les altres regions sí que s'ha detectat una petita anomalia, en aquesta regió no. Encara que el model continua detectant el pic més alt abans del previst.

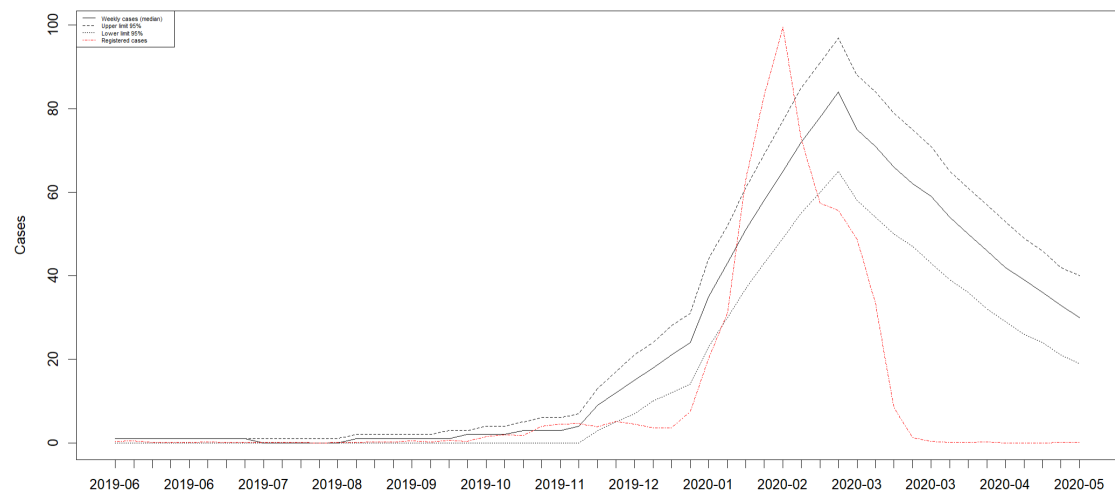


Figura 4.16: Regions de predicció per la temporada 2019-2020 a Lleida

Pel que fa a la Regió Sanitària de Girona, l'alerta epidemiològica no hauria saltat, ja que la línia de casos registrats no se sobresurt dels intervals. Una vegada s'ha arribat al valor més alt de contagis al mes de febrer, hi ha una gran davallada de contagis, que poden estar relacionats amb el fet que tots els casos de grip a partir del mes de març del 2020 es van començar a considerar com a possibles casos de la Covid-19 a causa de les seves similituds.

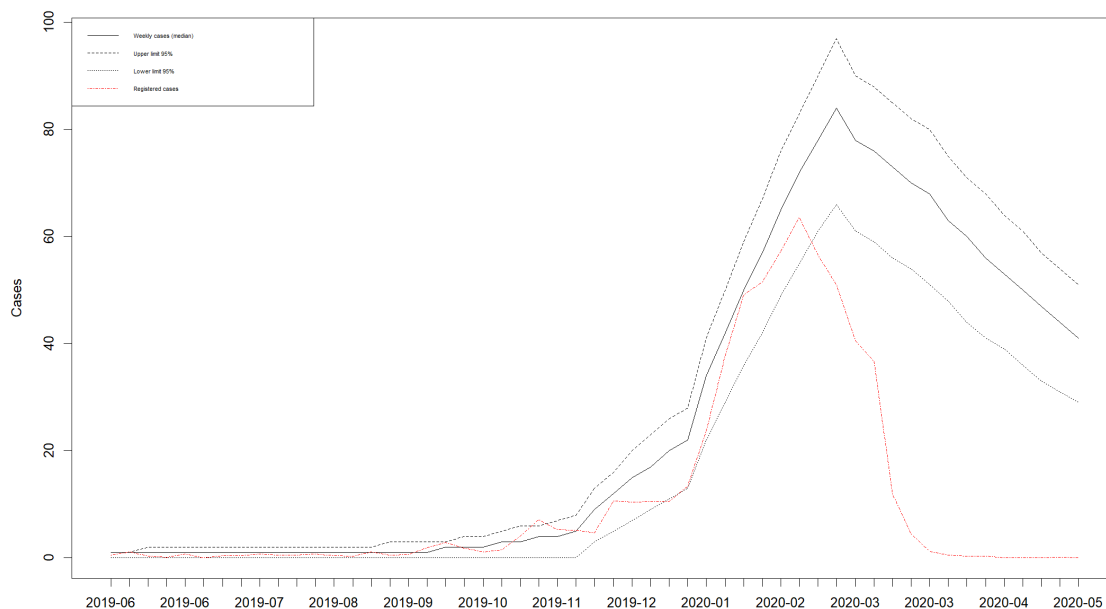


Figura 4.17: Regions de predicció per la temporada 2019-2020 a Girona

Finalment, per a l'última Regió Sanitària que fa referència a Catalunya Central, mostra un gràfic molt similar al vist en les altres regions. El nombre de contagis de la grip exponencialment a finals del desembre del 2019, arribant al punt més alt de contagis a mitjans de gener.

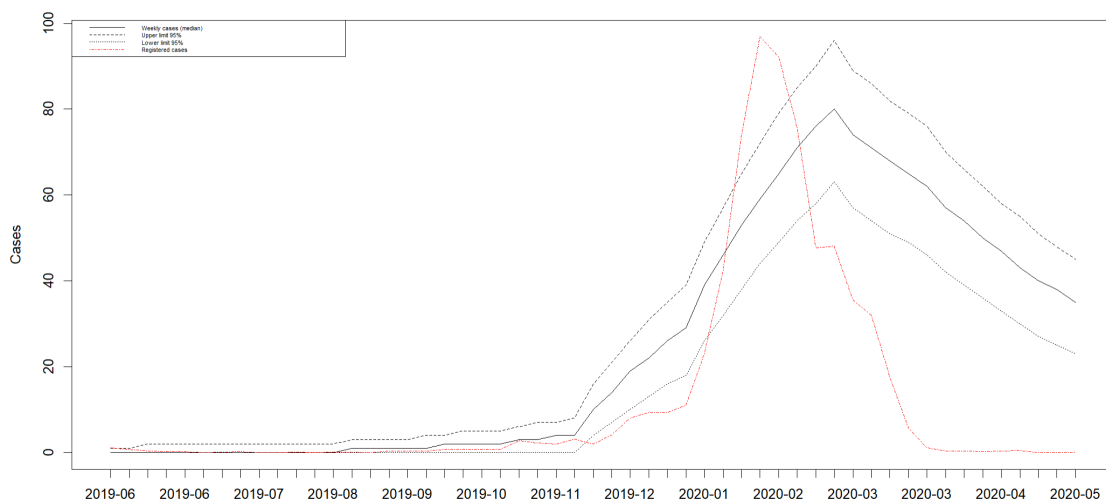


Figura 4.18: Regions de predicció per la temporada 2019-2020 a Catalunya Central

Així doncs, s'observa com en les Regions de predicció per la temporada 2019 - 2020 el model definit 4.1 és capaç d'adaptar-se i predir perfectament les tendències que segueix la sèrie. El llindar que marca l'inici del període epidèmic podria definir-se com el límit superior de la regió de predicció calculada a partir del model ajustat utilitzant les observacions del període no epidèmic previ. En totes les Regions Sanitàries coincideix que el brot de grip es troba en el seu pic més alt entre el febrer i març, mentre que realitzant-ho amb el model d'aquest estudi, a finals de l'any anterior, ja es comença a veure una progressió molt exponencial fins a arribar al pic més alt al gener. Aquesta representació tan visual recolza la clara idea que a partir del desembre la grip l'any 2019 dibuixa una corba diferent de les anteriors, per la qual cosa això podria haver fet saltar les alertes i detectar quelcom estava passant; en aquest cas, la Covid-19, tot i que la grip és una malaltia amb un comportament molt difícil de predir.

També és molt curiós observar com tots els casos detectats de grip a partir del gener del 2020 disminueixen de forma exponencial, fins a gairebé assolir els zero casos en mesos on en anys anteriors continuaven havent-hi casos. Això és degut al fet que segurament a principis del 2020, tots els casos de grip possiblement es consideraven casos sospitosos de la Covid-19, i per això a partir del març del 2020 només es registren de forma multitudinària casos de contagi de grip.

És important destacar que el canvi de sèrie temporal que va fer la grip a finals del 2019, per culpa de una entrada d'una nova malaltia, podria haver influenciat en com es van gestionar les coses dins del sector sanitari.

4.5. Barcelona

Com s'ha anat esmentant anteriorment, pel cas de la Regió Sanitària de Barcelona es va decidir realitzar un model de sèrie temporal diferent, perquè els nombres de contagis i de població en comparació amb les altres regions era molt elevat i el model INAR(2) no s'ajustaria tan bé com altres models.

4.5.1. Definició del model

La Regió Sanitària de Barcelona s'ha treballat d'una manera diferent, ja que com s'ha explicat anteriorment, amb un model INAR(2) no hauria pogut representar les regions de predicció correctament. És per això, que s'ha decidit identificar un tipus de sèrie temporal clàssica per a Barcelona a partir dels models autoregressius integrats de mitjana mòbil estacional SARIMA.

És habitual tractar amb sèries temporals amb períodes de mesura inferiors a un any, en les que és freqüent trobar-se amb patrons estacionals o cíclics. Pel cas de Barcelona, $S = 12$, ja que s'actua amb les dades de dotze mesos.

Tant si la sèrie temporal és estacionària o com si s'ha transformat en una sèrie estacionària diferenciant d vegades, que en aquest cas no ha estat així, el següent pas és seleccionar el model SARIMA apropiat, que vol dir aconseguir els valors p i q de SARIMA(p, d, q)(P, D, Q) $_S$ més adequats. Per això és necessari examinar l'ACF i la PACF de la sèrie temporal estacionària. Per representar aquestes funcions s'utilitzen les funcions de programa R , $acf()$ i $pacf()$, respectivament.

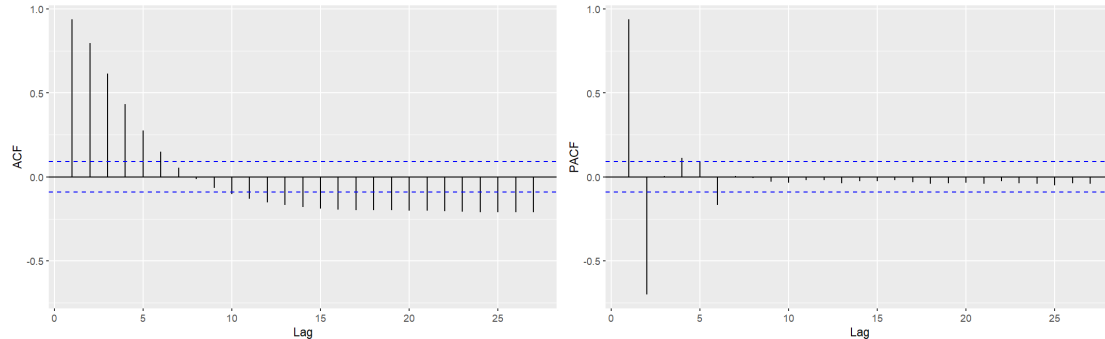


Figura 4.19: ACF i PACF de la Regió Sanitària de Barcelona

La gràfica de l'ACF, primerament no es té en compte la primera barra, ja que correspon a l'autocorrelació amb la mateixa observació, $retras/lag = 0$. Per tant, la segona i tercera barra són significatives. El cas de la gràfica de PACF, la primera barra correspon al primer retard, aquesta és molt significativa, la segona també però en l'altre sentit en ser negativa.

Una vegada s'han vist les gràfiques d'ACF i PACF, a partir de la funció $auto.arima()$ del programa R , que diu quin és el millor model per a la sèrie temporal que s'està tractant. Aquesta defineix com a millor model per a la sèrie temporal de la Regió Sanitària de Barcelona com un SARIMA(0,0,2)(0,1,1)[12].

4.5.2. Validació

Per a poder validar si el model és correcte, s'utilitza el test de Ljung-Box per veure si els residus són un soroll blanc, i, per tant, el model s'ajusta correctament a les dades de la sèrie temporal. Obtenint en el test un p-valor de 0,6911, s'accepta que sigui un soroll blanc. Per acabar de comprovar-ho, es realitzen els gràfics de l'ACF i dels residus.

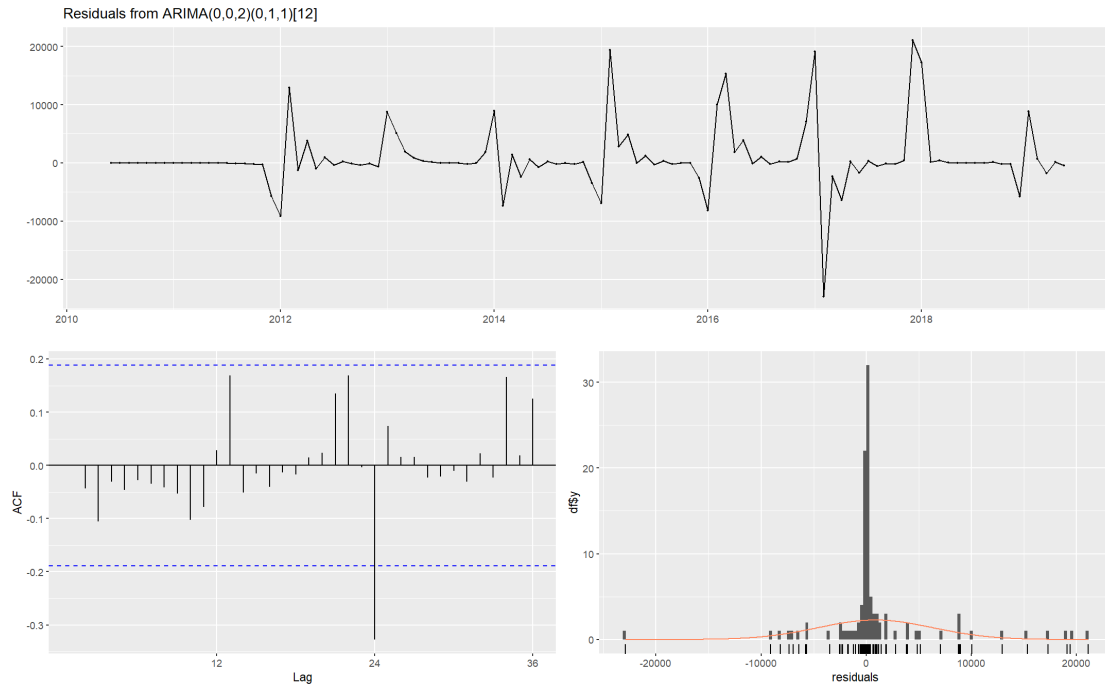


Figura 4.20: Residus de la Regió Sanitària de Barcelona

Es pot comprovar com els valors valen pràcticament zero i l'histograma dels residus sembla que segueixin una normal, fets que validen encara més el model proposat.

4.5.3. Prediccions

Una vegada que s'ha seleccionat el millor model SARIMA(0,0,2)(0,1,1)[12] per a les dades de la sèrie temporal, i s'han estimat els paràmetres per aquest model SARIMA, es mostren les prediccions dels valors futurs de Regió Sanitària de Barcelona. Per fer-ho es pot utilitzar la instrucció en R, `forecast()` del paquet "forecast".

Cal tenir present que aquesta gràfica segueix la mateixa llegenda que les anteriors, és a dir, la línia contínua de color negre dibuixa la mediana de casos setmanes de grip, les línies negres discontinues, mostren els límits superiors i inferiors i finalment, la línia de color vermell representa els casos registrats.

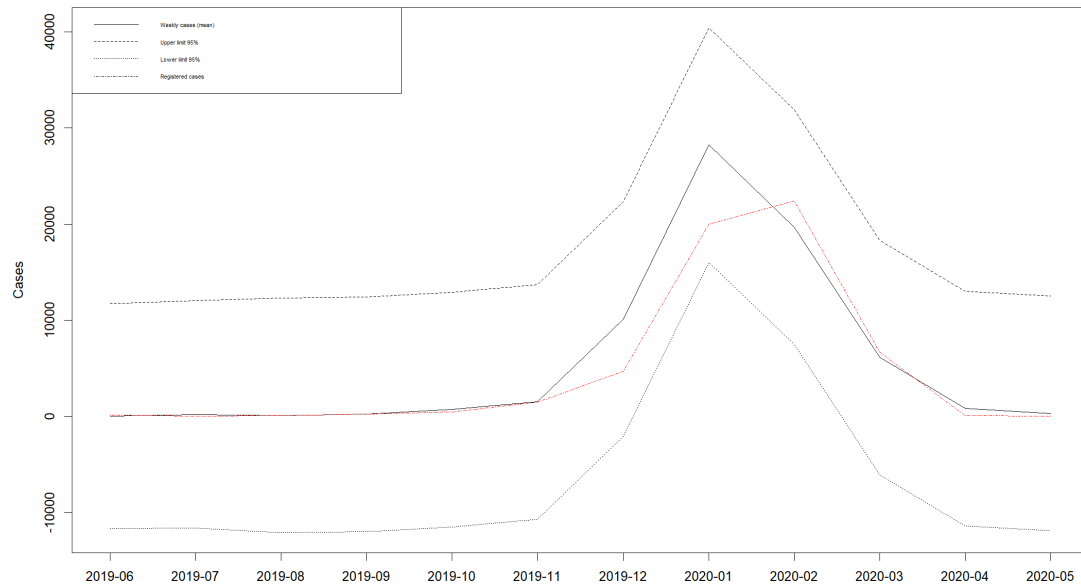


Figura 4.21: Regions de predicció per la temporada 2019-2020 a Barcelona

La figura 4.21 mostra les regions de predicció a la temporada 2019-2020 a la Regió Sanitària de Barcelona. Aquesta permet veure com el nombre de contagis va anar augmentant fins a arribar al punt més alt a principis de gener i, com la línia vermella no sobresurt dels intervals de confiança, i per això no hauria detectat cap anomalia i, per tant, no s'hauria declarat cap alerta. Barcelona és una regió sanitària amb molts valors com que agrupa moltes comarques i moltes d'aquestes amb molts habitants, per la qual cosa és molt difícil trobar un model que s'adapti i s'ajusti fàcilment a les dades. És important destacar que s'ha treballat amb el sistema de sèries temporals clàssiques a causa dels elevats nombres de recomptes que no permeten que s'ajusti bé a un model INAR com amb les altres regions sanitàries.

Capítol 5

Discussió i conclusions

Per últim, una vegada s'han obtingut els resultats necessaris per a aquesta anàlisi de sèries temporals, es fa un recull de les conclusions a les quals s'ha pogut arribar a terme.

L'objectiu principal d'aquest treball era construir una alerta epidemiològica que anticipés l'aparició de brots i permetés implementar les mesures necessàries a temps per a poder frenar l'augment de casos, posant com a exemple el desembre del 2019 que és quan va entrar la Covid-19 dins de la societat.

S'ha pogut veure que els models INAR que es presenten en aquest treball són una eina útil per poder detectar ràpidament quan està passant alguna cosa que s'escapa del comportament habitual, és a dir, detectar l'increment de casos de contagi de la grip, basats en els casos de contagis de setmanes anteriors. Malgrat això, quan s'està fent front a malalties que presenten una important component estacional és necessari incloure aquest impacte en els models. En aquest Treball Final de Grau s'ha comprovat que és suficient considerar correlacions de segon ordre, és a dir, un model INAR(2), tal com s'ha vist en les funcions d'autocorrelació i autocorrelació parcials vistes en les figures 4.5 i 4.6. Tots els mètodes presentats es poden estendre a models INAR(p) amb $p > 2$, tenint en compte que aquest tingui una estructura senzilla de treballar.

L'equilibri asimptòtic del model ha servit per tal d'obtenir una predicció completa, sense necessitat de fer servir les últimes observacions disponibles per poder veure amb agudesia com la Covid-19 va afectar els contagis de la grip i de la seva representació de sèrie temporal. Les regions de predicció proposades en les seccions 4.4 pretenen donar prediccions del nombre de contagis de grip per setmana amb un cert nivell de confiança per les diferents Regions Sanitàries. Es va voler posar a prova el model INAR(2) en diferents temporades epidemiològiques, una l'any 2011-2012, temporada que no tenia res a destacar, i la temporada de l'any 2019-2020, l'entrada d'una nova malaltia infecciosa. En ambdues, es va poder comprovar com el model s'hauria anticipat al brot, i per això es dona per vàlida l'alerta. A més, el mètode permet conèixer el nombre de contagis de grip en el moment $n + k$, coneixent el nombre de contagis en moments previs d'1 a n . Gràcies a la computació del programa, permet treballar i realitzar tots els càlculs necessaris per a aquestes prediccions en pocs minuts, pel que facilita i dona accés a poder efectuar prediccions amb ordinadors personals amb prestacions habituals.

Cal comentar que s'ha trobat amb diverses limitacions al llarg del desenvolupament del treball, com que la Regió Sanitària de Barcelona recull un nombre força elevat de recompte de contagis i els models INAR presenten problemes computacionals quan apareixen valors molt grans, per la qual cosa s'ha treballat finalment amb el sistema clàssic de sèries temporals amb el model SARIMA. Aquesta diferenciació de models permet comparar clarament una regió amb totes les altres, on les regions on s'ha aplicat el model INAR, declaren una alerta epidemiològica a finals de desembre del 2019, principis de gener del 2020, mentre que a Barcelona amb les regions de predicció que s'han aconseguit, no salta cap alerta, i per això en un model clàssic no s'hauria detectat l'entrada de la Covid-19 a la societat.

És cert que més que mai aquests últims anys, la societat ha hagut d'enfrontar-se a una epidèmia com és la Covid-19 i fer-ho de la millor manera possible significa l'ús d'un alt nombre de recursos que en circumstàncies normals no es troben disponibles de manera imminent. És per això, que el sistema de Salut ha de ser capaç de dissenyar estratègies de forma ràpida que solucionin els problemes que l'epidèmia provocarà, així com detectar l'aparició d'un nou virus per no saturar el sistema sanitari i hospitalari; tot això, en un poc temps, com que el moment en el qual apareix l'epidèmia o en aquest cas, la Covid-19, és desconegut. Per tant, és interessant el desenvolupament d'eines i el reforç d'aquestes que permetin preveure de manera aproximada el moment d'inici per tal d'assegurar l'efectivitat i eficiència del sistema. S'ha pogut comprovar que les prediccions basades en un termini curt de temps són de gran interès en les regions sanitàries de Catalunya, sense tenir en compte Barcelona, per tal de poder planificar millor l'ús dels recursos que ofereix els sistemes de Salut en la situació d'una epidèmia o pandèmia i no saturar-los.

Finalment, vist el potencial de l'alerta epidemiològica en les regions sanitàries on s'ha aplicat el model INAR(2), tenint les dades necessàries a l'abast, seria interessant implementar el sistema d'alerta epidemiològic en una pàgina web.

Bibliografia

- [1] Organización Mundial de Salud. *COVID-19: cronología de la actuación de la OMS*. URL: <https://www.who.int/es/news/item/27-04-2020-who-timeline---covid-19> (visitado 27-04-2020).
- [2] Agència de Salut Pública de Catalunya (ASPCAT). *Xarxa de Vigilància Epidemiològica de Catalunya (XVEC)*. URL: https://salutpublica.gencat.cat/ca/ambits/vigilancia_salut_publica/XVEC/ (visitado 08-10-2019).
- [3] Agència de Salut Pública de Catalunya (ASPCAT). *Pla d'informació de les infeccions respiratòries agudes a Catalunya (PIDIRAC)*. URL: <https://canalsalut.gencat.cat/ca/professionals/vigilancia-epidemiologica/pla-dinformacio-de-les-infeccions-respiratories-agudes-a-catalunya-pidirac/> (visitado 07-01-2022).
- [4] Butlletí Epidemiològic D E Catalunya. «BEC». En: (2019), págs. 139-161. URL: https://canalsalut.gencat.cat/web/.content/_Actualitat/Butlletins/Promocio_proteccio_salut/bec_butlleti_epidemiologic_de_catalunya/2019/bec-juliol-19.pdf.
- [5] Departament de Salut. «Sub-direcció General de Vigilància i Resposta a Emergències de Salut Pública». En: (2017). ISSN: 2020-2021. URL: https://canalsalut.gencat.cat/web/.content/_Professionals/Vigilancia_epidemiologica/documents/arxius/Informe-setmanal-farmacies.pdf.
- [6] Ana Martínez y col. *Redacció: Col·laboració: Grup de Treball de Vigilància de les Infeccions Respiratòries Agudes a Catalunya (PIDIRAC): Xarxa de metges sentinella de l'atenció Primària: E Martínez*. 2019. URL: http://canalsalut.gencat.cat/web/.content/_AZ/G/Grip/documents/arxius/grippostpan2010.pdf.
- [7] Agència de Salut Pública. «full_inter_temporada_pidirac». En: (). URL: https://canalsalut.gencat.cat/web/.content/_Professionals/Vigilancia_epidemiologica/documents/arxius/full_inter_temporada_pidirac.pdf.
- [8] *Butlletí de Prevenció d'Errors de Medicació de Catalunya. Xarxa de farmàcies sentinella de Catalunya. Una estratègia per a la detecció i el seguiment d'errors de medicació*. URL: https://scientiasalut.gencat.cat/bitstream/handle/11351/5413/xarxa_farmacies_sentinella_catalunya_estrategia_deteccio_seguiment_errors_medicacio_2020_ca.pdf?sequence=1&isAllowed=y.

- [9] *Package 'MMWRweek' Type Package Title Convert Dates to MMWR Day, Week, and Year*. 2020. URL: http://wwwn.cdc.gov/nndss/document/MMWR_Week_overview.pdf.
- [10] David Moriña y col. «A statistical model for hospital admissions caused by seasonal diseases». En: *Statistics in Medicine* 30 (26 2011), págs. 3125-3136. ISSN: 1097-0258. DOI: 10.1002/sim.4336. URL: <http://www.ncbi.nlm.nih.gov/pubmed/22025286>.
- [11] Ruijun Bu; Brendan McCabe; Kaddour Hadri (2008). «Maximum likelihood estimation of higher-order integer-valued autoregressive processes. », 29(6), 973-994.» En: (). URL: doi: 10.1111/j.1467-9892.2008.00590.x.
- [12] Departament de Salut. «pla-salut-territorial-alt-pirineu-aran». En: (). URL: `chrome-extension://efaidnbmnnnibpcajpcgclefindmkaj/viewer.html?pdfurl=https%3A%2F%2Fcatsalut.gencat.cat%2Fweb%2F.content%2Fminisite%2Fcatsalut%2Fterritori%2Falt-pirineu-aran%2Fpla-salut-territorial-alt-pirineu-aran-2016-2020.pdf&chunk=true`.

Annexos A

Codi R

A continuació, es presenta el codi R que s'ha utilitzat per a la realització d'aquest Treball Final de Grau.

Llibreries utilitzades

```
library(bayesforecast)
library(readxl)
library(MMWRweek)
library(forecast)
library(tidyverse)
library(ggplot2)
library(tidyverse)
library(ggplot2)
library(tseries)
library(fBasics)
library(dplyr)
library(stats)
```

Lectura de dades al R

```
#Base de Dades
grip <- read.csv2("diagnosticat.csv", sep = ";")
head(grip)
```

Gestió de la base de dades

```
grip$Any <- NA

grip$Any[grip$Temporada=="2010-2011" & grip$Setmana >=23] <- 2010
grip$Any[grip$Temporada=="2010-2011" & grip$Setmana <23] <- 2011
```

```

grip$Any[grip$Temporada=="2011-2012" & grip$Setmana >=23] <- 2011
grip$Any[grip$Temporada=="2011-2012" & grip$Setmana <23] <- 2012

grip$Any[grip$Temporada=="2012-2013" & grip$Setmana >=23] <- 2012
grip$Any[grip$Temporada=="2012-2013" & grip$Setmana <23] <- 2013

grip$Any[grip$Temporada=="2013-2014" & grip$Setmana >=23] <- 2013
grip$Any[grip$Temporada=="2013-2014" & grip$Setmana <23] <- 2014

grip$Any[grip$Temporada=="2014-2015" & grip$Setmana >=23] <- 2014
grip$Any[grip$Temporada=="2014-2015" & grip$Setmana <23] <- 2015

grip$Any[grip$Temporada=="2015-2016" & grip$Setmana >=23] <- 2015
grip$Any[grip$Temporada=="2015-2016" & grip$Setmana <23] <- 2016

grip$Any[grip$Temporada=="2016-2017" & grip$Setmana >=23] <- 2016
grip$Any[grip$Temporada=="2016-2017" & grip$Setmana <23] <- 2017

grip$Any[grip$Temporada=="2017-2018" & grip$Setmana >=23] <- 2017
grip$Any[grip$Temporada=="2017-2018" & grip$Setmana <23] <- 2018

grip$Any[grip$Temporada=="2018-2019" & grip$Setmana >=23] <- 2018
grip$Any[grip$Temporada=="2018-2019" & grip$Setmana <23] <- 2019

grip$Any[grip$Temporada=="2019-2020" & grip$Setmana >=23] <- 2019
grip$Any[grip$Temporada=="2019-2020" & grip$Setmana <23] <- 2020

grip$Any[grip$Temporada=="2020-2021" & grip$Setmana >=23] <- 2020
grip$Any[grip$Temporada=="2020-2021" & grip$Setmana <23] <- 2021

grip$Any[grip$Temporada=="2021-2022" & grip$Setmana >=23] <- 2021
grip$Any[grip$Temporada=="2021-2022" & grip$Setmana <23] <- 2022

#package lubridate: calcul mesos cada setmana
#install.packages("MMWRweek")
library("MMWRweek")
grip$DATE <-MMWRweek2Date(grip$Any, grip$Setmana)
grip$MONTH <-format(as.Date(grip$DATE), "%m")
grip$X <- NULL

```

Anàlisi descriptiu

```
for (i in 3:9){
  data<-data.frame(Any=c(grip$Any),
                  Month=c(grip$MONTH),Casos=c(grip[,i]))
  data <- sqldf("select Any,Month as Mes,sum(Casos) as Casos from data group by 1,2
  having Any > 2013")
  data$Mes<-factor(data$Mes)
  data$Any <- as.factor(data$Any)
  levels(data$Mes)<-month.abb
  zp<-ggplot(data,aes(x=Mes,y=Casos,group=Any,color=Any))
  zp<-zp+geom_line()+labs(title = paste (names(grip[i])),size=14)
  + scale_color_brewer(palette='Blues')
  print(zp)
}
```

```
data<-aggregate(grip$Camp.de.Tarragona, by=list(grip$MONTH,grip$Any), FUN=sum)
max(data$x)
```

```
data_ebr<-aggregate(grip$Terres.de.l.Ebre, by=list(grip$MONTH,grip$Any), FUN=sum)
```

```
data_piri<-aggregate(grip$Alt.Pirineu.i.Aran, by=list(grip$MONTH,grip$Any), FUN=sum)
max(data_piri$x)
```

```
data_llei<-aggregate(grip$Lleida, by=list(grip$MONTH,grip$Any), FUN=sum)
max(data_llei$x)
```

```
data_gir<-aggregate(grip$Girona, by=list(grip$MONTH,grip$Any), FUN=sum)
max(data_gir$x)
```

```
data_centr<-aggregate(grip$Catalunya.Central, by=list(grip$MONTH,grip$Any), FUN=sum)
max(data_centr$x)
```

```
data_bcn<-aggregate(grip$Barcelona, by=list(grip$MONTH,grip$Any), FUN=sum)
max(data_bcn$x)
```

Funcions d'Autocorrelació i Autocorrelació parcial

```
t<-ggAcf(grip$Camp.de.Tarragona, lag.max = 12,main='Camp.de.Tarragona')
e<-ggAcf(grip$Terres.de.l.Ebre, lag.max = 12,main='Terres.de.l.Ebre')
a<-ggAcf(grip$Alt.Pirineu.i.Aran, lag.max = 12,main='Alt.Pirineu.i.Aran')
ll<-ggAcf(grip$Lleida, lag.max = 12,main='Lleida')
g<-ggAcf(grip$Girona, lag.max = 12,main='Girona')
```

```

c<-ggAcf(grip$Catalunya.Central, lag.max = 12,main='Catalunya.Central')
b<-ggAcf(grip$Barcelona, lag.max = 12,main='Barcelona')

ggarrange(t,e,a,ll,g,c,b, nrow=2, ncol=4,common.legend = FALSE)

t1<-ggPacf(grip$Camp.de.Tarragona, lag.max = 12,main='Camp.de.Tarragona')
e1<-ggPacf(grip$Terres.de.l.Ebre, lag.max = 12,main='Terres.de.l.Ebre')
a1<-ggPacf(grip$Alt.Pirineu.i.Aran, lag.max = 12,main='Alt.Pirineu.i.Aran')
ll1<-ggPacf(grip$Lleida, lag.max = 12,main='Lleida')
g1<-ggPacf(grip$Girona, lag.max = 12,main='Girona')
c1<-ggPacf(grip$Catalunya.Central, lag.max = 12,main='Catalunya.Central')
b1<-ggPacf(grip$Barcelona, lag.max = 12,main='Barcelona')

ggarrange(t1,e1,a1,ll1,g1,c1,b1, nrow=2, ncol=4,common.legend = FALSE)

```

Gestió de les poblacions de les regions sanitàries

```

pob <- read.csv2("Registre_central_de_poblaci__del_CatSalut
__poblaci__per__rea_b_sica_de_salut.csv", sep = ",",header=TRUE)
pob$'codi.Region.Sanitaria' <- NULL
pob$codi.Rea.B.sica.de.Saut <- NULL
pob$g.nere <- NULL
pob$edat <- NULL
pob$Rea.B.sica.de.Salut <- NULL

names(pob)[names(pob) == "i.any"] <- "any"
names(pob)[names(pob) == "Region.Sanitaria"] <- "regio"
names(pob)[names(pob) == "poblacion.oficial"] <- "poblacio"
head(pob)

pob_agrupats <- aggregate(pob$poblacio,by=list(any=pob$any,regio=pob$regio),FUN=sum)

poblacions <- pob_agrupats[!pob_agrupats$regio=="SENSE ESPECIFICAR", ]
poblacions

any <- c(2010,2010,2010,2010,2010,2010,2010)
regio <- c("ALT PIRINEU i ARAN","BARCELONA","CAMP DE TARRAGONA","CATALUNYA CENTRAL",
"GIRONA","LLEIDA","TERRES DE L'EBRE")
x<-c(69977,5063118,593241,516857,844942,365679,191920)

data_1 <- data.frame(any,regio,x)

```

```

data_1

poblacions <- rbind(data_1,poblacions)
colnames(poblacions)
poblacions

any <- c(2011,2011,2011,2011,2011,2011,2011)
regio <- c("ALT PIRINEU i ARAN","BARCELONA","CAMP DE TARRAGONA","CATALUNYA CENTRAL",
"GIRONA","LLEIDA","TERRES DE L'EBRE")
x <- c(69457,5029329,593997,517831,845943,367748,191971)

data_2 <- data.frame(any,regio,x)
data_2

poblacions <- rbind(data_2,poblacions)
colnames(poblacions)
poblacions

write.xlsx2(poblacions,file="poblacions.xlsx", sheetName = "poblacions",
col.names = TRUE, row.names = TRUE, append = FALSE, showNA = TRUE, password = NULL)

```

Model INAR(2)

```

#Component combinatoria que s'utilitzarà en la funció loglik
coefcomb <- function(n,m)
{
  resultat <- factorial(n)/(factorial(m)*factorial(abs(n-m)))
  resultat
}

loglik <- function(b)
{
  loglik <- 0
  periode <- length(b) - 2
  for (i in 3:length(x))
  {
    if (y[i] == "01")
    {
      c <- 0
      d <- 0
      m <- min(x[i-1], x[i])

```

```

for (k in 0:m)
{
  l <- min(x[i-2], x[i]-k)
  for (j in 0:l)
  {
    d <- d+((coefcomb(x[i-2],x[i]-j)*(b[periode+2]^j)*
      (1-b[periode+2])^(x[i-2]-j))*(exp(-b[1])*b[1]^(x[i]-k-j))/factorial(x[i]-k-j))
    }#for
    c <- c + (b[periode+1]^k*(1-b[periode+1])^(x[i-1]-k)*coefcomb(x[i-1],k))*d
  }#for
  loglik <- loglik - log(c)
}#if
if (y[i] == "02")
{
  c <- 0
  d <- 0
  m <- min(x[i-1], x[i])
  for (k in 0:m)
  {
    l <- min(x[i-2],x[i]-k)
    for (j in 0:l)
    {
      d <- d+((coefcomb(x[i-2],x[i]-j)*(b[periode+2]^j)*
        (1-b[periode+2])^(x[i-2]-j))*(exp(-b[2])*b[2]^(x[i]-k-j))/factorial(x[i]-k-j))
      }#for
      c <- c + (b[periode+1]^k*(1-b[periode+1])^(x[i-1]-k)*coefcomb(x[i-1],k))*d
    }#for
    loglik <- loglik - log(c)
  }#if
if (y[i] == "03")
{
  c <- 0
  d <- 0
  m <- min(x[i-1], x[i])
  for (k in 0:m)
  {
    l <- min(x[i-2],x[i]-k)
    for (j in 0:l)
    {
      d <- d+((coefcomb(x[i-2],x[i]-j)*(b[periode+2]^j)*

```



```

        (1-b[periode+2])^(x[i-2]-j))*(exp(-b[3])*b[3]^(x[i]-k-j))/factorial(x[i]-k-j))
    }#for
    c <- c + (b[periode+1]^k*(1-b[periode+1])^(x[i-1]-k)*coefcomb(x[i-1],k))*d
  }#for
  loglik <- loglik - log(c)
}#if
if (y[i] == "04")
{
  c <- 0
  d <- 0
  m <- min(x[i-1], x[i])
  for (k in 0:m)
  {
    l <- min(x[i-2], x[i]-k)
    for (j in 0:l)
    {
      d <- d+((coefcomb(x[i-2],x[i]-j)*(b[periode+2]^j)*
        (1-b[periode+2])^(x[i-2]-j))*(exp(-b[4])*b[4]^(x[i]-k-j))/factorial(x[i]-k-j))
    }#for
    c <- c + (b[periode+1]^k*(1-b[periode+1])^(x[i-1]-k)*coefcomb(x[i-1],k))*d
  }#for
  loglik <- loglik - log(c)
}#if
if (y[i] == "05")
{
  c <- 0
  d <- 0
  m <- min(x[i-1], x[i])
  for (k in 0:m)
  {
    l <- min(x[i-2],x[i]-k)
    for (j in 0:l)
    {
      d <- d+((coefcomb(x[i-2],x[i]-j)*(b[periode+2]^j)*
        (1-b[periode+2])^(x[i-2]-j))*(exp(-b[5])*b[5]^(x[i]-k-j))/factorial(x[i]-k-j))
    }#for
    c <- c + (b[periode+1]^k*(1-b[periode+1])^(x[i-1]-k)*coefcomb(x[i-1],k))*d
  }#for
  loglik <- loglik - log(c)
}#if

```

```

if (y[i] == "06")
{
  c <- 0
  d <- 0
  m <- min(x[i-1], x[i])
  for (k in 0:m)
  {
    l <- min(x[i-2], x[i]-k)
    for (j in 0:l)
    {
      d <- d+((coefcomb(x[i-2],x[i]-j)*(b[periode+2]^j)*
        (1-b[periode+2])^(x[i-2]-j))*(exp(-b[6])*b[6]^(x[i]-k-j))/factorial(x[i]-k-j))
    }#for
    c <- c + (b[periode+1]^k*(1-b[periode+1])^(x[i-1]-k)*coefcomb(x[i-1],k))*d
  }#for
  loglik <- loglik - log(c)
}#if
if (y[i] == "07")
{
  c <- 0
  d <- 0
  m <- min(x[i-1], x[i])
  for (k in 0:m)
  {
    l <- min(x[i-2], x[i]-k)
    for (j in 0:l)
    {
      d <- d+((coefcomb(x[i-2],x[i]-j)*(b[periode+2]^j)*
        (1-b[periode+2])^(x[i-2]-j))*(exp(-b[7])*b[7]^(x[i]-k-j))/factorial(x[i]-k-j))
    }#for
    c <- c + (b[periode+1]^k*(1-b[periode+1])^(x[i-1]-k)*coefcomb(x[i-1],k))*d
  }#for
  loglik <- loglik - log(c)
}#if
if (y[i] == "08")
{
  c <- 0
  d <- 0
  m <- min(x[i-1], x[i])
  for (k in 0:m)

```

```

{
  l <- min(x[i-2], x[i]-k)
  for (j in 0:l)
  {
    d <- d+((coefcomb(x[i-2],x[i]-j)*(b[periode+2]^j)*
      (1-b[periode+2])^(x[i-2]-j))*(exp(-b[8])*b[8]^(x[i]-k-j))/factorial(x[i]-k-j))
    }#for
    c <- c + (b[periode+1]^k*(1-b[periode+1])^(x[i-1]-k)*coefcomb(x[i-1],k))*d
  }#for
  loglik <- loglik - log(c)
}#if
if (y[i] == "09")
{
  c <- 0
  d <- 0
  m <- min(x[i-1], x[i])
  for (k in 0:m)
  {
    l <- min(x[i-2], x[i]-k)
    for (j in 0:l)
    {
      d <- d+((coefcomb(x[i-2],x[i]-j)*(b[periode+2]^j)*
        (1-b[periode+2])^(x[i-2]-j))*(exp(-b[9])*b[9]^(x[i]-k-j))/factorial(x[i]-k-j))
      }#for
      c <- c + (b[periode+1]^k*(1-b[periode+1])^(x[i-1]-k)*coefcomb(x[i-1],k))*d
    }#for
    loglik <- loglik - log(c)
  }#if
if (y[i] == "10")
{
  c <- 0
  d <- 0
  m <- min(x[i-1], x[i])
  for (k in 0:m)
  {
    l <- min(x[i-2], x[i]-k)
    for (j in 0:l)
    {
      d <- d+((coefcomb(x[i-2],x[i]-j)*(b[periode+2]^j)*
        (1-b[periode+2])^(x[i-2]-j))*(exp(-b[10])*b[10]^(x[i]-k-j))/factorial(x[i]-k-j))
    }#for
  }#if
}

```

```

    }#for
    c <- c + (b[periode+1]^k*(1-b[periode+1])^(x[i-1]-k)*coefcomb(x[i-1],k))*d
  }#for
  loglik <- loglik - log(c)
}#if
if (y[i] == "11")
{
  c <- 0
  d <- 0
  m <- min(x[i-1], x[i])
  for (k in 0:m)
  {
    l <- min(x[i-2], x[i]-k)
    for (j in 0:l)
    {
      d <- d+((coefcomb(x[i-2],x[i]-j)*(b[periode+2]^j)*
        (1-b[periode+2])^(x[i-2]-j))*(exp(-b[11])*(b[11])^(x[i]-k-j))/factorial(x[i]-k-j))
    }#for
    c <- c + (b[periode+1]^k*(1-b[periode+1])^(x[i-1]-k)*coefcomb(x[i-1],k))*d
  }#for
  loglik <- loglik - log(c)
}#if
if (y[i] == "12")
{
  c <- 0
  d <- 0
  m <- min(x[i-1], x[i])
  for (k in 0:m)
  {
    l <- min(x[i-2], x[i]-k)
    for (j in 0:l)
    {
      d <- d+((coefcomb(x[i-2],x[i]-j)*(b[periode+2]^j)*
        (1-b[periode+2])^(x[i-2]-j))*(exp(-b[12])*(b[12])^(x[i]-k-j))/factorial(x[i]-k-j))
    }#for
    c <- c + (b[periode+1]^k*(1-b[periode+1])^(x[i-1]-k)*coefcomb(x[i-1],k))*d
  }#for
  loglik <- loglik - log(c)
}#if
}#for

```

```

    return(loglik)
}#loglik

#calcul de x
Calcul_X <- function(regio,data_frame_aux,n)
{x <- c()
for(i in 1:nrow(data_frame_aux))
{for(j in 1:nrow(regio))
{if(regio$Any[j]==data_frame_aux[i,1])
{x <-c(x,round(regio[j,1]/data_frame_aux[i,2]*n))}
}
}
return(x)}

grip <- grip[!grip$Temporada %in% c("2019-2020", "2020-2021", "2021-2022"), ]
any <- 2010:2019

#Pirineu
Pirineu <- grip[,c("Alt.Pirineu.i.Aran", "MONTH", "Any")]
N <- c(69977,69457,68688,68652,68417,67632,67084,66833,66867,66813)#,67277,68518)
pob_pirineu <- data.frame(any=any,N=N)
x <- Calcul_X(Pirineu,pob_pirineu,60000)
y <- Pirineu[,2]
# Maximum likelihood estimation of the parameters:
MLE.Pirineu <- nlm(f=loglik,p=c(9,6,3,1.5,1.5,2,4,2,3,4,5,6,0.1,0.1),hessian=TRUE,
iterlim=1000)
MLE.Pirineu$estimate
sqrt(diag(solve(MLE.Pirineu$hessian)))

#Terres de l'Ebre
Ebre <- grip[,c("Terres.de.l.Ebre", "MONTH", "Any")]
N <- c(190241,188573,187359,186010,184503,182070,179873,178159,177057,176488)#,176817,178112)
pob_ebre <- data.frame(any=any,N=N)
x <- Calcul_X(Ebre,pob_ebre,25000)
y <- Ebre[,2]
# Maximum likelihood estimation of the parameters:
MLE.Ebre <- nlm(f=loglik,p=c(9,6,3,1.5,1.5,2,4,2,3,4,5,6,0.1,0.1),hessian=TRUE,
iterlim=1000)
MLE.Ebre$code
MLE.Ebre$estimate

```

```

sqrt(diag(solve(MLE.Ebre$hessian)))

#Lleida
Lleida <- grip[,c("Lleida", "MONTH", "Any")]
N <- c(365679,367748,368086,366469,364568,361085,358998,358062,358070,359558)#,362850,365322
pob_lleida <- data.frame(any=any,N=N)
x <- Calcul_X(Lleida,pob_lleida,45000)
y <- Lleida[,2]
# Maximum likelihood estimation of the parameters:
MLE.Lleida <- nlm(f=loglik,p=c(9,6,3,1.5,1.5,2,4,2,3,4,5,6,0.1,0.1),hessian=TRUE,
iterlim=1000)
MLE.Lleida$code
MLE.Lleida$estimate
sqrt(diag(solve(MLE.Lleida$hessian)))

#Camp.de.Tarragona
Tarragona <- grip[,c("Camp.de.Tarragona", "MONTH", "Any")]
N <- c(593241,593997,596328,595836,596508,593391,592010,591625,594861,600603)#,607999,614830)
pob_tarragona <- data.frame(any=any,N=N)
x <- Calcul_X(Tarragona,pob_tarragona,60000)
y <- Tarragona[,2]
# Maximum likelihood estimation of the parameters:
MLE.Tarragona <- nlm(f=loglik,p=c(9,6,3,1.5,1.5,2,4,2,3,4,5,6,0.1,0.1),hessian=TRUE,
iterlim=1000)
MLE.Tarragona$code
MLE.Tarragona$estimate
sqrt(diag(solve(MLE.Tarragona$hessian)))

#Barcelona
Barcelona <- grip[,c("Barcelona", "MONTH", "Any")]
N <- c(5063118,5029329,5018722,4992322,4982346,4947837,4938121,4943443,4964136,4996165)#
,5050190,5068733)
pob_barcelona <- data.frame(any=any,N=N)
x <- Calcul_X(Barcelona,pob_barcelona,65000)
y <- Barcelona[,2]
# Maximum likelihood estimation of the parameters:
MLE.Barcelona<- nlm(f=loglik,p=c(9,6,3,1.5,1.5,2,4,2,3,4,5,6,0.1,0.1),hessian=TRUE,
iterlim=1000)
MLE.Barcelona$code
MLE.Barcelona$estimate

```

```

sqrt(diag(solve(MLE.Barcelona$hessian)))

#Girona
Girona <- grip[,c("Girona", "MONTH", "Any")]
N <- c(844942,845943,845142,843135,844771,839960,837042,838103,843268,850699)#,861753,870481)
pob_girona <- data.frame(any=any,N=N)
x <- Calcul_X(Girona,pob_girona,61000)
y <- Girona[,2]
# Maximum likelihood estimation of the parameters:
MLE.Girona<- nlm(f=loglik,p=c(9,6,3,1.5,1.5,2,4,2,3,4,5,6,0.1,0.1),hessian=TRUE,
iterlim=1000)
MLE.Girona$code
MLE.Girona$estimate
sqrt(diag(solve(MLE.Girona$hessian)))

#Catalunya.Central
Central <- grip[,c("Catalunya.Central", "MONTH", "Any")]
N <- c(516857,517831,517477,516557,515213,511707,510619,512050,514646,520104)#,526959,531073)
pob_central <- data.frame(any=any,N=N)
x <- Calcul_X(Central,pob_central,50000)
y <- Central[,2]
# Maximum likelihood estimation of the parameters:
MLE.Central<- nlm(f=loglik,p=c(9,6,3,1.5,1.5,2,4,2,3,4,5,6,0.1,0.1),hessian=TRUE,
iterlim=1000)
MLE.Central$code
MLE.Central$estimate
sqrt(diag(solve(MLE.Central$hessian)))

```

Regions de Predicció per la temporada 2019-2020

```

grip <- grip[!grip$Temporada %in% c("2020-2021", "2021-2022"), ]
any <- 2010:2020
grip_predit <- grip[grip$Temporada=="2019-2020", ]
grip <- grip[grip$Temporada!="2019-2020", ]

coefcomb <- function(n,m)
{
  resultat <- factorial(n)/(factorial(m)*factorial(abs(n-m)))
  return(resultat)
}

```

```

probfunc <- function(prob,p1,p2,lamb)
{
  c <- 0
  d <- 0
  m <- min(x[477],prob)
  for (k in 0:m)
  {
    l <- min(x[476],prob-k)
    for (j in 0:l)
    {
      d <- d+((coefcomb(x[476],j)*(p2^j)*(1-p2)^(x[476]-j))*(exp(-lamb)*lamb^(prob-k-j))/
              factorial(prob-k-j))
    }
    c <- c + (p1^k*(1-p1)^(x[477]-k)*coefcomb(x[477],k))*d
  }
  return(c)
}

#####
#Pirineu
#####
Pirineu <- grip[,c("Alt.Pirineu.i.Aran", "MONTH", "Any")]
N <- c(69977,69457,68688,68652,68417,67632,67084,66833,66867,66813,67277)#,68518)
pob_pirineu <- data.frame(any=any,N=N)
x <- Calcul_X(Pirineu,pob_pirineu,60000)
y <- Pirineu[,2]
n <- length(x)

# Calcul dels limits de la regio de prediccio al 95%:
p <- c(0.5681273,0.2914024,
      11.6912550, 12.9416043 , 2.7848853,
      0.7392861, 0.4960868, 0.2915468,
      0.1699720,0.2817136, 0.5153849,
      1.2500798, 1.8722109, 7.0974621)

lambda <- vector()
lambda <- p[as.numeric(grip_predit$MONTH)+2]

f <- vector()
g <- vector()
l <- vector()

```



```

f[1] <- p[1]
f[2] <- p[1]^2 + p[2]
g[1] <- p[2]
g[2] <- p[1]*p[2]
l[1] <- lambda[1]
l[2] <- p[1]*lambda[1] + lambda[2]

for (i in 3:length(lambda))
{
  f[i] <- p[1]*f[i-1] + p[2]*f[i-2]
  g[i] <- p[1]*g[i-1] + p[2]*g[i-2]
  l[i] <- p[1]*l[i-1] + p[2]*l[i-2] + lambda[i]
}
# Calcul dels limits de la regio de prediccio
a <- vector()
b <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    if (sumprob < .025)
    {
      a[i] <- k
    }
    k <- k + 1
  }
  k <- 0
  sumprob <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    b[i] <- k
    k <- k + 1
  }
}
}

```

```

for (i in 1:length(lambda))
{
  if (is.na(a[i]) == TRUE) a[i] <- 0
}
# Calcul de la mediana de cada  $X_{t+k}$ :
m <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .50)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    m[i] <- k
    k <- k + 1
  }
}

# Limit inferior de la regio de prediccio:
liminf2 <- a

# Limit superior de la regio de prediccio:
limsup2 <- b

# Limit superior de la regio de prediccio:
med2 <- m

dates <- substr(as.character(grip_predit$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regio de confian?a:
dibuix2 <- ts(matrix(c(med2, limsup2,
liminf2, grip_predit$Alt.Pirineu.i.Aran/66813*60000),length(lambda), 4))
colnames(dibuix2) <- c("Weekly cases (median)","Upper limit 95%",
"Lower limit 95%", "Registered cases")
#postscript("LaTeX/img/morina_met3_pred2.ps")
ts.plot(x=dibuix2,
        gpars=list(xlab="",ylab="Cases", lty=c(1:4),xaxt="n"),
        col=c(rep("black",3),rep("red",1)))
axis(side=1,at=c(1:53),labels=dates)

```

```

legend("topleft", legend=colnames(dibuix2),
col=c(rep("black",3),rep("red",1)), lty=c(1:4), cex=0.45)
dev.off()

#####
#Terres de l'Ebre
#####
Ebre <- grip[,c("Terres.de.l.Ebre", "MONTH", "Any")]
N <- c(593241,593997,187359,186010,184503,182070,179873,178159,177057,176488,176817)#,178112)
pob_ebre <- data.frame(any=any,N=N)
x <- Calcul_X(Ebre,pob_ebre,25000)
y <- Ebre[,2]
n <- length(x)

# Calcul dels limits de la regio de prediccio al 95%:
p <- c(0.59580661, 0.29878597,
      6.24059904, 12.44257577, 1.56283791,
      0.20256797, 0.09018772, 0.08760571,
      0.05577783, 0.05650421, 0.14213784,
      0.20204065, 0.53652750, 2.5317060)

lambda <- vector()
lambda <- p[as.numeric(grip_predit$MONTH)+2]

f <- vector()
g <- vector()
l <- vector()
f[1] <- p[1]
f[2] <- p[1]^2 + p[2]
g[1] <- p[2]
g[2] <- p[1]*p[2]
l[1] <- lambda[1]
l[2] <- p[1]*lambda[1] + lambda[2]

for (i in 3:length(lambda))
{
  f[i] <- p[1]*f[i-1] + p[2]*f[i-2]
  g[i] <- p[1]*g[i-1] + p[2]*g[i-2]
  l[i] <- p[1]*l[i-1] + p[2]*l[i-2] + lambda[i]
}

```

```

# Calcul dels limits de la regio de prediccio
a <- vector()
b <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k      <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    if (sumprob < .025)
    {
      a[i] <- k
    }
    k <- k + 1
  }
  k <- 0
  sumprob <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    b[i] <- k
    k <- k + 1
  }
}
for (i in 1:length(lambda))
{
  if (is.na(a[i]) == TRUE) a[i] <- 0
}
# Calcul de la mediana de cada  $X_{t+k}$ :
m <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k      <- 0
  while (sumprob < .50)
  {
    probk <- probfunc(k,f[i],g[i],l[i])

```

```

    sumprob <- probk + sumprob
    m[i] <- k
    k <- k + 1
  }
}

# Limit inferior de la regio de prediccio:
liminf2 <- a

# Limit superior de la regio de prediccio:
limsup2 <- b

# Limit superior de la regio de prediccio:
med2 <- m

dates <- substr(as.character(grip_predit$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regio de confianca:
dibuix2 <- ts(matrix(c(med2, limsup2, liminf2,
grip_predit$Terres.de.l.Ebre/176488*25000),length(lambda), 4))
colnames(dibuix2) <- c("Weekly cases (median)","Upper limit 95%",
"Lower limit 95%", "Registered cases")
#postscript("LaTeX/img/morina_met3_pred2.ps")
ts.plot(x=dibuix2,
        gpars=list(xlab="",ylab="Cases", lty=c(1:4),xaxt="n"),
        col=c(rep("black",3),rep("red",1)))
axis(side=1,at=c(1:53),labels=dates)
legend("topleft", legend=colnames(dibuix2),
col=c(rep("black",3),rep("red",1)), lty=c(1:4), cex=0.45)
dev.off()

#####
#Lleida
#####
Lleida <- grip[,c("Lleida", "MONTH", "Any")]
N <- c(365679,367748,368086,366469,364568,361085,358998,358062,358070,359558,362850)#,365322
pob_lleida <- data.frame(any=any,N=N)
x <- Calcul_X(Lleida,pob_lleida,45000)
y <- Lleida[,2]

```

```

n <- length(x)

# Calcul dels limits de la regio de prediccio al 95%:
p <- c(0.61898359, 0.27562354,
      14.70589387, 15.50211610, 1.71824267,
      0.21064635, 0.13458694, 0.03218233,
      0.02439116, 0.02564038, 0.25818842,
      0.61583795, 0.85696050, 5.58339886)

p <- c(0.61898359, 0.27562354, 14.70589387,15.50211610, 1.71824267, 0.21064635 ,
      0.13458694 , 0.03218233, 0.02439116, 0.02564038, 0.25818842 ,
      0.61583795 , 0.85696050 , 5.58339886)
lambda <- vector()
lambda <- p[as.numeric(grip_predit$MONTH)+2]

f <- vector()
g <- vector()
l <- vector()
f[1] <- p[1]
f[2] <- p[1]^2 + p[2]
g[1] <- p[2]
g[2] <- p[1]*p[2]
l[1] <- lambda[1]
l[2] <- p[1]*lambda[1] + lambda[2]

for (i in 3:length(lambda))
{
  f[i] <- p[1]*f[i-1] + p[2]*f[i-2]
  g[i] <- p[1]*g[i-1] + p[2]*g[i-2]
  l[i] <- p[1]*l[i-1] + p[2]*l[i-2] + lambda[i]
}

# Calcul dels limits de la regio de prediccio
a <- vector()
b <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .95)
  {

```

```

    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    if (sumprob < .025)
    {
        a[i] <- k
    }
    k <- k + 1
}
k <- 0
sumprob <- 0
while (sumprob < .95)
{
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    b[i] <- k
    k <- k + 1
}
}
for (i in 1:length(lambda))
{
    if (is.na(a[i]) == TRUE) a[i] <- 0
}
# Calcul de la mediana de cada  $X_{t+k}$ :
m <- vector()
for (i in 1:length(lambda))
{
    sumprob <- 0
    k <- 0
    while (sumprob < .50)
    {
        probk <- probfunc(k,f[i],g[i],l[i])
        sumprob <- probk + sumprob
        m[i] <- k
        k <- k + 1
    }
}
}

# Limit inferior de la regio de prediccio:
liminf2 <- a

```

```

# Limit superior de la regio de prediccio:
limsup2 <- b

# Limit superior de la regio de prediccio:
med2 <- m

dates <- substr(as.character(grip_predit$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regi? de confian?a:
dibuix2 <- ts(matrix(c(med2, limsup2, liminf2, grip_predit$Lleida/359558*45000),
length(lambda), 4))
colnames(dibuix2) <- c("Weekly cases (median)", "Upper limit 95%", "Lower limit 95%",
"Registered cases")
#postscript("LaTeX/img/morina_met3_pred2.ps")
ts.plot(x=dibuix2,
        gpars=list(xlab="", ylab="Cases", lty=c(1:4), xaxt="n"),
        col=c(rep("black",3), rep("red",1)))
axis(side=1, at=c(1:53), labels=dates)
legend("topleft", legend=colnames(dibuix2),
col=c(rep("black",3), rep("red",1)), lty=c(1:4), cex=0.45)
dev.off()

#####
#Camp.de.Tarragona
#####
Tarragona <- grip[,c("Camp.de.Tarragona", "MONTH", "Any")]
N <- c(593241,593997,596328,595836,596508,593391,592010,591625,594861,600603,607999)#,614830)
pob_tarragona <- data.frame(any=any,N=N)
x <- Calcul_X(Tarragona,pob_tarragona,60000)
y <- Tarragona[,2]
n <- length(x)

# Calcul dels limits de la regio de prediccio al 95%:
p <- c(0.61952046, 0.27132991,
        13.24526040, 16.29358851, 1.96926447,
        0.33596875, 0.07183926, 0.09732600,
        0.04878055, 0.08152714, 0.31404050,
        0.38737347, 1.11204728, 4.76460128)
lambda <- vector()

```



```

lambda <- p[as.numeric(grip_predit$MONTH)+2]

f <- vector()
g <- vector()
l <- vector()
f[1] <- p[1]
f[2] <- p[1]^2 + p[2]
g[1] <- p[2]
g[2] <- p[1]*p[2]
l[1] <- lambda[1]
l[2] <- p[1]*lambda[1] + lambda[2]

for (i in 3:length(lambda))
{
  f[i] <- p[1]*f[i-1] + p[2]*f[i-2]
  g[i] <- p[1]*g[i-1] + p[2]*g[i-2]
  l[i] <- p[1]*l[i-1] + p[2]*l[i-2] + lambda[i]
}
# Calcul dels limits de la regio de prediccio
a <- vector()
b <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    if (sumprob < .025)
    {
      a[i] <- k
    }
    k <- k + 1
  }
  k <- 0
  sumprob <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])

```

```

        sumprob <- probk + sumprob
        b[i] <- k
        k <- k + 1
    }
}
for (i in 1:length(lambda))
{
    if (is.na(a[i]) == TRUE) a[i] <- 0
}
# Calcul de la mediana de cada  $X_{t+k}$ :
m <- vector()
for (i in 1:length(lambda))
{
    sumprob <- 0
    k <- 0
    while (sumprob < .50)
    {
        probk <- probfunc(k,f[i],g[i],l[i])
        sumprob <- probk + sumprob
        m[i] <- k
        k <- k + 1
    }
}

# Limit inferior de la regio de prediccio:
liminf2 <- a

# Limit superior de la regio de prediccio:
limsup2 <- b

# Limit superior de la regio de prediccio:
med2 <- m

dates <- substr(as.character(grip_predit$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regio de confianca:
dibuix2 <- ts(matrix(c(med2, limsup2, liminf2, grip_predit$Camp.de.Tarragona/600603*60000),
length(lambda), 4))
colnames(dibuix2) <- c("Weekly cases (median)", "Upper limit 95%",
"Lower limit 95%", "Registered cases")

```

```

#postscript("LaTeX/img/morina_met3_pred2.ps")
ts.plot(x=dibuix2,
        gpars=list(xlab="",ylab="Cases", lty=c(1:4),xaxt="n"),
        col=c(rep("black",3),rep("red",1)))
axis(side=1,at=c(1:53),labels=dates)
legend("topleft", legend=colnames(dibuix2),
col=c(rep("black",3),rep("red",1)), lty=c(1:4), cex=0.45)
dev.off()

#####
#Girona
#####
Girona <- grip[,c("Girona", "MONTH", "Any")]
N <- c(844942,845943,845142,843135,844771,839960,837042,838103,843268,850699,861753)#,870481)
pob_girona <- data.frame(any=any,N=N)
x <- Calcul_X(Girona,pob_girona,61000)
y <- Girona[,2]
n <- length(x)

# Calcul dels limits de la regio de prediccio al 95%:
p <- c(0.62856906, 0.28873065,
      14.03755866, 14.49452117, 2.48880183,
      0.52558861, 0.17454352, 0.10925695,
      0.11177493, 0.04212126, 0.33274735,
      0.63706859, 1.03825325, 4.81504621)
lambda <- vector()
lambda <- p[as.numeric(grip_predit$MONTH)+2]

f <- vector()
g <- vector()
l <- vector()
f[1] <- p[1]
f[2] <- p[1]^2 + p[2]
g[1] <- p[2]
g[2] <- p[1]*p[2]
l[1] <- lambda[1]
l[2] <- p[1]*lambda[1] + lambda[2]

for (i in 3:length(lambda))
{

```

```

f[i] <- p[1]*f[i-1] + p[2]*f[i-2]
g[i] <- p[1]*g[i-1] + p[2]*g[i-2]
l[i] <- p[1]*l[i-1] + p[2]*l[i-2] + lambda[i]
}
# Calcul dels limits de la regio de prediccio
a <- vector()
b <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    if (sumprob < .025)
    {
      a[i] <- k
    }
    k <- k + 1
  }
  k <- 0
  sumprob <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    b[i] <- k
    k <- k + 1
  }
}
for (i in 1:length(lambda))
{
  if (is.na(a[i]) == TRUE) a[i] <- 0
}
# Calcul de la mediana de cada  $X_{t+k}$ :
m <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0

```

```

k      <- 0
while (sumprob < .50)
{
  probk <- probfunc(k,f[i],g[i],l[i])
  sumprob <- probk + sumprob
  m[i] <- k
  k <- k + 1
}
}

# Limit inferior de la regio de prediccio:
liminf2 <- a

# Limit superior de la regio de prediccio:
limsup2 <- b

# Limit superior de la regio de prediccio:
med2 <- m

dates <- substr(as.character(grip_predit$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regio de confianca:
dibuix2 <- ts(matrix(c(med2, limsup2, liminf2, grip_predit$Girona/850699*61000),
length(lambda), 4))
colnames(dibuix2) <- c("Weekly cases (median)", "Upper limit 95%",
"Lower limit 95%", "Registered cases")
#postscript("LaTeX/img/morina_met3_pred2.ps")
ts.plot(x=dibuix2,
        gpars=list(xlab="", ylab="Cases", lty=c(1:4), xaxt="n"),
        col=c(rep("black",3), rep("red",1)))
axis(side=1, at=c(1:53), labels=dates)
legend("topleft", legend=colnames(dibuix2),
col=c(rep("black",3), rep("red",1)), lty=c(1:4), cex=0.45)
dev.off()

#####
#Catalunya.Central
#####
Central <- grip[,c("Catalunya.Central", "MONTH", "Any")]

```

```

N <- c(516857,517831,517477,516557,515213,511707,510619,512050,514646,520104,526959)#,531073)
pob_central <- data.frame(any=any,N=N)
x <- Calcul_X(Central,pob_central,50000)
y <- Central[,2]
n <- length(x)

# Calcul dels limits de la regio de prediccio al 95%:
p <- c(0.62647079, 0.28249746,
      13.16716874, 13.12544692, 2.29455829,
      0.10973690, 0.05860777, 0.11053944,
      0.05060306, 0.05329175, 0.32420126,
      0.52502587, 0.84538663, 6.68168639)

lambda <- vector()
lambda <- p[as.numeric(grip_predit$MONTH)+2]

f <- vector()
g <- vector()
l <- vector()
f[1] <- p[1]
f[2] <- p[1]^2 + p[2]
g[1] <- p[2]
g[2] <- p[1]*p[2]
l[1] <- lambda[1]
l[2] <- p[1]*lambda[1] + lambda[2]

for (i in 3:length(lambda))
{
  f[i] <- p[1]*f[i-1] + p[2]*f[i-2]
  g[i] <- p[1]*g[i-1] + p[2]*g[i-2]
  l[i] <- p[1]*l[i-1] + p[2]*l[i-2] + lambda[i]
}
# Calcul dels limits de la regio de prediccio
a <- vector()
b <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .95)

```

```

{
  probk <- probfunc(k,f[i],g[i],l[i])
  sumprob <- probk + sumprob
  if (sumprob < .025)
  {
    a[i] <- k
  }
  k <- k + 1
}
k <- 0
sumprob <- 0
while (sumprob < .95)
{
  probk <- probfunc(k,f[i],g[i],l[i])
  sumprob <- probk + sumprob
  b[i] <- k
  k <- k + 1
}
}
for (i in 1:length(lambda))
{
  if (is.na(a[i]) == TRUE) a[i] <- 0
}
# Calcul de la mediana de cada  $X_{t+k}$ :
m <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .50)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    m[i] <- k
    k <- k + 1
  }
}
}

# Limit inferior de la regio de prediccio:
liminf2 <- a

```

```

# Limit superior de la regio de prediccio:
limsup2 <- b

# Limit superior de la regio de prediccio:
med2     <- m

dates <- substr(as.character(grip_predit$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regio de confiança:
dibuix2 <- ts(matrix(c(med2, limsup2, liminf2, grip_predit$Catalunya.Central/520104*50000),
length(lambda), 4))
colnames(dibuix2) <- c("Weekly cases (median)", "Upper limit 95%",
"Lower limit 95%", "Registered cases")
#postscript("LaTeX/img/morina_met3_pred2.ps")
ts.plot(x=dibuix2,
        gpars=list(xlab="", ylab="Cases", lty=c(1:4), xaxt="n"),
        col=c(rep("black",3),
rep("red",1)))
axis(side=1, at=c(1:53), labels=dates)
legend("topleft", legend=colnames(dibuix2),
col=c(rep("black",3),
rep("red",1)), lty=c(1:4), cex=0.45)
dev.off()

```

Regions de Predicció per la temporada 2011-2012

```

grip <- grip[!grip$Temporada %in% c("2019-2020", "2020-2021", "2021-2022"), ]
any <- 2010:2019
grip_predit <- grip[grip$Temporada=="2011-2012", ]

coefcomb <- function(n,m)
{
  resultat <- factorial(n)/(factorial(m)*factorial(abs(n-m)))
  return(resultat)
}

probfunc <- function(prob,p1,p2,lamb)
{
  c <- 0
  d <- 0
  m <- min(x[53],prob)

```



```

for (k in 0:m)
{
  l <- min(x[52],prob-k)
  for (j in 0:l)
  {
    d <- d+((coefcomb(x[52],j)*(p2^j)*(1-p2)^(x[52]-j))*(exp(-lamb)*lamb^(prob-k-j))/
            factorial(prob-k-j))
  }
  c <- c + (p1^k*(1-p1)^(x[53]-k)*coefcomb(x[53],k))*d
}
return(c)
}
#Pirineu
Pirineu <- grip[,c("Alt.Pirineu.i.Aran", "MONTH", "Any")]
N <- c(69977,69457,68688,68652,68417,67632,67084,66833,66867,66813)#,67277,68518)
pob_pirineu <- data.frame(any=any,N=N)
x <- Calcul_X(Pirineu,pob_pirineu,60000)
y <- Pirineu[,2]
n <- length(x)

# Calcul dels limits de la regio de prediccio al 95%:
p <- c(0.5681273,0.2914024,
       11.6912550, 12.9416043 , 2.7848853,
       0.7392861, 0.4960868, 0.2915468,
       0.1699720,0.2817136, 0.5153849,
       1.2500798, 1.8722109, 7.0974621)
lambda <- vector()
lambda <- p[as.numeric(grip_predit$MONTH)+2]

f <- vector()
g <- vector()
l <- vector()
f[1] <- p[1]
f[2] <- p[1]^2 + p[2]
g[1] <- p[2]
g[2] <- p[1]*p[2]
l[1] <- lambda[1]
l[2] <- p[1]*lambda[1] + lambda[2]

for (i in 3:length(lambda))

```

```

{
  f[i] <- p[1]*f[i-1] + p[2]*f[i-2]
  g[i] <- p[1]*g[i-1] + p[2]*g[i-2]
  l[i] <- p[1]*l[i-1] + p[2]*l[i-2] + lambda[i]
}
# Calcul dels limits de la regio de prediccio
a <- vector()
b <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    if (sumprob < .025)
    {
      a[i] <- k
    }
    k <- k + 1
  }
  k <- 0
  sumprob <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    b[i] <- k
    k <- k + 1
  }
}
for (i in 1:length(lambda))
{
  if (is.na(a[i]) == TRUE) a[i] <- 0
}
# Calcul de la mediana de cada  $X_{t+k}$ :
m <- vector()
for (i in 1:length(lambda))
{

```

```

sumprob <- 0
k      <- 0
while (sumprob < .50)
{
  probk <- probfunc(k,f[i],g[i],l[i])
  sumprob <- probk + sumprob
  m[i] <- k
  k <- k + 1
}
}

# Limit inferior de la regio de prediccio:
liminf2 <- a

# Limit superior de la regio de prediccio:
limsup2 <- b

# Limit superior de la regio de prediccio:
med2 <- m

dates <- substr(as.character(grip_predit$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regio de confiança:
dibuix2 <- ts(matrix(c(med2, limsup2, liminf2, grip_predit$Alt.Pirineu.i.Aran/69457*60000),
length(lambda), 4))
colnames(dibuix2) <- c("Weekly cases (median)", "Upper limit 95%",
"Lower limit 95%", "Registered cases")
#postscript("LaTeX/img/morina_met3_pred2.ps")
ts.plot(x=dibuix2,
        gpars=list(xlab="", ylab="Cases", lty=c(1:4), xaxt="n"),
        col=c(rep("black",3), rep("red",1)))
axis(side=1, at=c(1:53), labels=dates)
legend("topleft", legend=colnames(dibuix2),
col=c(rep("black",3), rep("red",1)), lty=c(1:4), cex=0.45)
dev.off()

#####
#Terres de l'Ebre
#####

```

```

Ebre <- grip[,c("Terres.de.l.Ebre", "MONTH", "Any")]
N <- c(593241,593997,187359,186010,184503,182070,179873,178159,177057,176488)#,176817)#,178112)
pob_ebre <- data.frame(any=any,N=N)
x <- Calcul_X(Ebre,pob_ebre,25000)
y <- Ebre[,2]
n <- length(x)

# Calcul dels limits de la regio de prediccio al 95%:
p <- c(0.59580661, 0.29878597,
      6.24059904, 12.44257577, 1.56283791,
      0.20256797, 0.09018772, 0.08760571,
      0.05577783, 0.05650421, 0.14213784,
      0.20204065, 0.53652750, 2.53170607)
lambda <- vector()
lambda <- p[as.numeric(grip_predit$MONTH)+2]

f <- vector()
g <- vector()
l <- vector()
f[1] <- p[1]
f[2] <- p[1]^2 + p[2]
g[1] <- p[2]
g[2] <- p[1]*p[2]
l[1] <- lambda[1]
l[2] <- p[1]*lambda[1] + lambda[2]

for (i in 3:length(lambda))
{
  f[i] <- p[1]*f[i-1] + p[2]*f[i-2]
  g[i] <- p[1]*g[i-1] + p[2]*g[i-2]
  l[i] <- p[1]*l[i-1] + p[2]*l[i-2] + lambda[i]
}
# Calcul dels limits de la regio de prediccio
a <- vector()
b <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .95)

```

```

{
  probk <- probfunc(k,f[i],g[i],l[i])
  sumprob <- probk + sumprob
  if (sumprob < .025)
  {
    a[i] <- k
  }
  k <- k + 1
}
k <- 0
sumprob <- 0
while (sumprob < .95)
{
  probk <- probfunc(k,f[i],g[i],l[i])
  sumprob <- probk + sumprob
  b[i] <- k
  k <- k + 1
}
}
for (i in 1:length(lambda))
{
  if (is.na(a[i]) == TRUE) a[i] <- 0
}
# Calcul de la mediana de cada  $X_{t+k}$ :
m <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .50)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    m[i] <- k
    k <- k + 1
  }
}
}

# Limit inferior de la regio de prediccio:
liminf2 <- a

```

```

# Limit superior de la regio de prediccio:
limsup2 <- b

# Limit superior de la regio de prediccio:
med2 <- m

dates <- substr(as.character(grip_predit$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regio de confianca:
dibuix2 <- ts(matrix(c(med2, limsup2, liminf2, grip_predit$Terres.de.l.Ebre/593997*25000),
length(lambda), 4))
colnames(dibuix2) <- c("Weekly cases (median)", "Upper limit 95%",
"Lower limit 95%", "Registered cases")
#postscript("LaTeX/img/morina_met3_pred2.ps")
ts.plot(x=dibuix2,
        gpars=list(xlab="", ylab="Cases", lty=c(1:4), xaxt="n"),
        col=c(rep("black",3), rep("red",1)))
axis(side=1, at=c(1:53), labels=dates)
legend("topleft", legend=colnames(dibuix2),
col=c(rep("black",3), rep("red",1)), lty=c(1:4), cex=0.45)
dev.off()

#####
#Lleida
#####
Lleida <- grip[,c("Lleida", "MONTH", "Any")]
N <- c(365679,367748,368086,366469,364568,361085,358998,358062,358070,359558)#362850)#,365322
pob_lleida <- data.frame(any=any,N=N)
x <- Calcul_X(Lleida,pob_lleida,45000)
y <- Lleida[,2]
n <- length(x)

# Calcul dels limits de la regio de prediccio al 95%:
p<- c(0.61898359, 0.27562354, 14.70589387,15.50211610, 1.71824267, 0.21064635 ,
0.13458694 , 0.03218233,0.02439116,
0.02564038, 0.25818842 ,0.61583795 , 0.85696050 , 5.58339886)

lambda <- vector()
lambda <- p[as.numeric(grip_predit$MONTH)+2]

```

```

f <- vector()
g <- vector()
l <- vector()
f[1] <- p[1]
f[2] <- p[1]^2 + p[2]
g[1] <- p[2]
g[2] <- p[1]*p[2]
l[1] <- lambda[1]
l[2] <- p[1]*lambda[1] + lambda[2]

for (i in 3:length(lambda))
{
  f[i] <- p[1]*f[i-1] + p[2]*f[i-2]
  g[i] <- p[1]*g[i-1] + p[2]*g[i-2]
  l[i] <- p[1]*l[i-1] + p[2]*l[i-2] + lambda[i]
}
# Calcul dels limits de la regio de prediccio
a <- vector()
b <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    if (sumprob < .025)
    {
      a[i] <- k
    }
    k <- k + 1
  }
  k <- 0
  sumprob <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
  }
}

```

```

    b[i] <- k
    k <- k + 1
  }
}
for (i in 1:length(lambda))
{
  if (is.na(a[i]) == TRUE) a[i] <- 0
}
# Calcul de la mediana de cada  $X_{t+k}$ :
m <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .50)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    m[i] <- k
    k <- k + 1
  }
}

# Limit inferior de la regio de prediccio:
liminf2 <- a

# Limit superior de la regio de prediccio:
limsup2 <- b

# Limit superior de la regio de prediccio:
med2 <- m

dates <- substr(as.character(grip_predit$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regio de confianca:
dibuix2 <- ts(matrix(c(med2, limsup2, liminf2, grip_predit$Lleida/367748*45000),
length(lambda), 4))
colnames(dibuix2) <- c("Weekly cases (median)", "Upper limit 95%",
"Lower limit 95%", "Registered cases")
#postscript("LaTeX/img/morina_met3_pred2.ps")

```



```

ts.plot(x=dibuix2,
        gpars=list(xlab="",ylab="Cases", lty=c(1:4),xaxt="n"),
        col=c(rep("black",3),rep("red",1)))
axis(side=1,at=c(1:53),labels=dates)
legend("topleft", legend=colnames(dibuix2),
col=c(rep("black",3),rep("red",1)), lty=c(1:4), cex=0.45)
dev.off()

#####
#Camp.de.Tarragona
#####
Tarragona <- grip[,c("Camp.de.Tarragona", "MONTH", "Any")]
N <- c(593241,593997,596328,595836,596508,593391,592010,591625,594861,600603,607999)#,614830)
pob_tarragona <- data.frame(any=any,N=N)
x <- Calcul_X(Tarragona,pob_tarragona,60000)
y <- Tarragona[,2]
n <- length(x)

# Calcul dels limits de la regio de prediccio al 95%:
p <- c(0.61952046, 0.27132991,
       13.24526040, 16.29358851, 1.96926447,
       0.33596875, 0.07183926, 0.09732600,
       0.04878055, 0.08152714, 0.31404050,
       0.38737347, 1.11204728, 4.76460128)
lambda <- vector()
lambda <- p[as.numeric(grip_predit$MONTH)+2]

f <- vector()
g <- vector()
l <- vector()
f[1] <- p[1]
f[2] <- p[1]^2 + p[2]
g[1] <- p[2]
g[2] <- p[1]*p[2]
l[1] <- lambda[1]
l[2] <- p[1]*lambda[1] + lambda[2]

for (i in 3:length(lambda))
{

```

```

f[i] <- p[1]*f[i-1] + p[2]*f[i-2]
g[i] <- p[1]*g[i-1] + p[2]*g[i-2]
l[i] <- p[1]*l[i-1] + p[2]*l[i-2] + lambda[i]
}
# Calcul dels limits de la regio de prediccio
a <- vector()
b <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    if (sumprob < .025)
    {
      a[i] <- k
    }
    k <- k + 1
  }
  k <- 0
  sumprob <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    b[i] <- k
    k <- k + 1
  }
}
for (i in 1:length(lambda))
{
  if (is.na(a[i]) == TRUE) a[i] <- 0
}
# Calcul de la mediana de cada  $X_{t+k}$ :
m <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0

```

```

k      <- 0
while (sumprob < .50)
{
  probk <- probfunc(k,f[i],g[i],l[i])
  sumprob <- probk + sumprob
  m[i] <- k
  k <- k + 1
}
}

# Limit inferior de la regio de prediccio:
liminf2 <- a

# Limit superior de la regio de prediccio:
limsup2 <- b

# Limit superior de la regio de prediccio:
med2 <- m

dates <- substr(as.character(grip_predit$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regio de confianca:
dibuix2 <- ts(matrix(c(med2, limsup2, liminf2, grip_predit$Camp.de.Tarragona/593997*60000),
length(lambda), 4))
colnames(dibuix2) <- c("Weekly cases (median)", "Upper limit 95%",
"Lower limit 95%", "Registered cases")
#postscript("LaTeX/img/morina_met3_pred2.ps")
ts.plot(x=dibuix2,
        gpars=list(xlab="", ylab="Cases", lty=c(1:4), xaxt="n"),
        col=c(rep("black",3), rep("red",1)))
axis(side=1, at=c(1:53), labels=dates)
legend("topleft", legend=colnames(dibuix2),
col=c(rep("black",3), rep("red",1)), lty=c(1:4), cex=0.45)
dev.off()

#####
#Girona
#####
Girona <- grip[,c("Girona", "MONTH", "Any")]

```

```

N <- c(844942,845943,845142,843135,844771,839960,837042,838103,843268,850699,861753)#,870481)
pob_girona <- data.frame(any=any,N=N)
x <- Calcul_X(Girona,pob_girona,61000)
y <- Girona[,2]
n <- length(x)

# Calcul dels limits de la regio de prediccio al 95%:
p <- c(0.62856906, 0.28873065,
      14.03755866, 14.49452117,  2.48880183,
      0.52558861,  0.17454352,  0.10925695,
      0.11177493,  0.04212126,  0.33274735,
      0.63706859,  1.03825325,  4.81504621)
lambda <- vector()
lambda <- p[as.numeric(grip_predit$MONTH)+2]

f <- vector()
g <- vector()
l <- vector()
f[1] <- p[1]
f[2] <- p[1]^2 + p[2]
g[1] <- p[2]
g[2] <- p[1]*p[2]
l[1] <- lambda[1]
l[2] <- p[1]*lambda[1] + lambda[2]

for (i in 3:length(lambda))
{
  f[i] <- p[1]*f[i-1] + p[2]*f[i-2]
  g[i] <- p[1]*g[i-1] + p[2]*g[i-2]
  l[i] <- p[1]*l[i-1] + p[2]*l[i-2] + lambda[i]
}

# Calcul dels limits de la regio de prediccio
a <- vector()
b <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .95)
  {

```

```

    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    if (sumprob < .025)
    {
        a[i] <- k
    }
    k <- k + 1
}
k <- 0
sumprob <- 0
while (sumprob < .95)
{
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    b[i] <- k
    k <- k + 1
}
}
for (i in 1:length(lambda))
{
    if (is.na(a[i]) == TRUE) a[i] <- 0
}
# Calcul de la mediana de cada  $X_{t+k}$ :
m <- vector()
for (i in 1:length(lambda))
{
    sumprob <- 0
    k <- 0
    while (sumprob < .50)
    {
        probk <- probfunc(k,f[i],g[i],l[i])
        sumprob <- probk + sumprob
        m[i] <- k
        k <- k + 1
    }
}
}

# Limit inferior de la regio de prediccio:
liminf2 <- a

```

```

# Limit superior de la regio de prediccio:
limsup2 <- b

# Limit superior de la regio de prediccio:
med2 <- m

dates <- substr(as.character(grip_predit$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regio de confiança:
dibuix2 <- ts(matrix(c(med2, limsup2, liminf2, grip_predit$Girona/845943*61000),
length(lambda), 4))
colnames(dibuix2) <- c("Weekly cases (median)", "Upper limit 95%",
"Lower limit 95%", "Registered cases")
#postscript("LaTeX/img/morina_met3_pred2.ps")
ts.plot(x=dibuix2,
        gpars=list(xlab="", ylab="Cases", lty=c(1:4), xaxt="n"),
        col=c(rep("black",3), rep("red",1)))
axis(side=1, at=c(1:53), labels=dates)
legend("topleft", legend=colnames(dibuix2),
col=c(rep("black",3),
rep("red",1)), lty=c(1:4), cex=0.45)
dev.off()

#####
#Catalunya.Central
#####
Central <- grip[,c("Catalunya.Central", "MONTH", "Any")]
N <- c(516857,517831,517477,516557,515213,511707,510619,512050,514646,520104,526959)#,531073)
pob_central <- data.frame(any=any,N=N)
x <- Calcul_X(Central,pob_central,50000)
y <- Central[,2]
n <- length(x)

# Calcul dels limits de la regio de prediccio al 95%:
p <- c(0.62647079, 0.28249746,
13.16716874, 13.12544692, 2.29455829,
0.10973690, 0.05860777, 0.11053944,
0.05060306, 0.05329175, 0.32420126,
0.52502587, 0.84538663, 6.68168639)

```

```

lambda <- vector()
lambda <- p[as.numeric(grip_predit$MONTH)+2]

f <- vector()
g <- vector()
l <- vector()
f[1] <- p[1]
f[2] <- p[1]^2 + p[2]
g[1] <- p[2]
g[2] <- p[1]*p[2]
l[1] <- lambda[1]
l[2] <- p[1]*lambda[1] + lambda[2]

for (i in 3:length(lambda))
{
  f[i] <- p[1]*f[i-1] + p[2]*f[i-2]
  g[i] <- p[1]*g[i-1] + p[2]*g[i-2]
  l[i] <- p[1]*l[i-1] + p[2]*l[i-2] + lambda[i]
}
# Calcul dels limits de la regio de prediccio
a <- vector()
b <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .95)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    if (sumprob < .025)
    {
      a[i] <- k
    }
    k <- k + 1
  }
  k <- 0
  sumprob <- 0
  while (sumprob < .95)
  {

```

```

    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    b[i] <- k
    k <- k + 1
  }
}
for (i in 1:length(lambda))
{
  if (is.na(a[i]) == TRUE) a[i] <- 0
}
# Calcul de la mediana de cada  $X_{t+k}$ :
m <- vector()
for (i in 1:length(lambda))
{
  sumprob <- 0
  k <- 0
  while (sumprob < .50)
  {
    probk <- probfunc(k,f[i],g[i],l[i])
    sumprob <- probk + sumprob
    m[i] <- k
    k <- k + 1
  }
}

# Limit inferior de la regio de prediccio:
liminf2 <- a

# Limit superior de la regio de prediccio:
limsup2 <- b

# Limit superior de la regio de prediccio:
med2 <- m

dates <- substr(as.character(grip_predit$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regio de confianca:

dibuix2 <- ts(matrix(c(med2, limsup2, liminf2, grip_predit$Catalunya.Central/517831*50000),
length(lambda), 4))

```



```

colnames(dibuix2) <- c("Weekly cases (median)", "Upper limit 95%",
"Lower limit 95%", "Registered cases")
#postscript("LaTeX/img/morina_met3_pred2.ps")
ts.plot(x=dibuix2,
        gpars=list(xlab="", ylab="Cases", lty=c(1:4), xaxt="n"),
        col=c(rep("black",3), rep("red",1)))
axis(side=1, at=c(1:53), labels=dates)
legend("topleft", legend=colnames(dibuix2),
col=c(rep("black",3), rep("red",1)), lty=c(1:4), cex=0.45)
dev.off()

```

Model SARIMA per la Regió de Barcelona

#Base de Dades

```

grip <- read.csv2("diagnosticat.csv", sep = ";")
head(grip)

```

#Gestio de la base de dades

```

grip$Any <- NA

```

```

grip$Any[grip$Temporada=="2010-2011" & grip$Setmana >=23] <- 2010
grip$Any[grip$Temporada=="2010-2011" & grip$Setmana <23] <- 2011

```

```

grip$Any[grip$Temporada=="2011-2012" & grip$Setmana >=23] <- 2011
grip$Any[grip$Temporada=="2011-2012" & grip$Setmana <23] <- 2012

```

```

grip$Any[grip$Temporada=="2012-2013" & grip$Setmana >=23] <- 2012
grip$Any[grip$Temporada=="2012-2013" & grip$Setmana <23] <- 2013

```

```

grip$Any[grip$Temporada=="2013-2014" & grip$Setmana >=23] <- 2013
grip$Any[grip$Temporada=="2013-2014" & grip$Setmana <23] <- 2014

```

```

grip$Any[grip$Temporada=="2014-2015" & grip$Setmana >=23] <- 2014
grip$Any[grip$Temporada=="2014-2015" & grip$Setmana <23] <- 2015

```

```

grip$Any[grip$Temporada=="2015-2016" & grip$Setmana >=23] <- 2015
grip$Any[grip$Temporada=="2015-2016" & grip$Setmana <23] <- 2016

```

```

grip$Any[grip$Temporada=="2016-2017" & grip$Setmana >=23] <- 2016
grip$Any[grip$Temporada=="2016-2017" & grip$Setmana <23] <- 2017

```

```

grip$Any[grip$Temporada=="2017-2018" & grip$Setmana >=23] <- 2017
grip$Any[grip$Temporada=="2017-2018" & grip$Setmana <23] <- 2018

grip$Any[grip$Temporada=="2018-2019" & grip$Setmana >=23] <- 2018
grip$Any[grip$Temporada=="2018-2019" & grip$Setmana <23] <- 2019

grip$Any[grip$Temporada=="2019-2020" & grip$Setmana >=23] <- 2019
grip$Any[grip$Temporada=="2019-2020" & grip$Setmana <23] <- 2020

grip$Any[grip$Temporada=="2020-2021" & grip$Setmana >=23] <- 2020
grip$Any[grip$Temporada=="2020-2021" & grip$Setmana <23] <- 2021

grip$Any[grip$Temporada=="2021-2022" & grip$Setmana >=23] <- 2021
grip$Any[grip$Temporada=="2021-2022" & grip$Setmana <23] <- 2022

#package lubridate: calcul mesos cada setmana
#install.packages("MMWRweek")
library("MMWRweek")
grip$DATE <- MMWRweek2Date(grip$Any, grip$Setmana)
grip$MONTH <- format(as.Date(grip$DATE), "%m")
grip$X <- NULL
#grip2 <- grip[grip$Any <= 2018,]
#grip2 <- grip2[grip2$Any > 2010, ]

grip <- grip[!grip$Temporada %in% c("2020-2021", "2021-2022"), ]
grip$Any <- format(as.Date(grip$DATE), "%Y")
grip_predit <- grip[grip$Temporada=="2019-2020", ]
grip <- grip[grip$Temporada!="2019-2020", ]
grip_predit$DATE <- as.Date(paste0(grip_predit$Any, "-", grip_predit$MONTH, "-01"),
format="%Y-%m-%d")
grip_predit2 <- grip_predit %>% group_by(Any) %>% summarise(casosBCN=sum(Barcelona))

#Barcelona
Barcelona <- grip[,c("Barcelona", "MONTH", "Any")]
Barcelona$DATE <- as.Date(paste0(Barcelona$Any, "-", Barcelona$MONTH, "-01"),
format="%Y-%m-%d")
dades <- Barcelona %>% group_by(Any) %>% summarise(casos=sum(Barcelona))
%>% arrange(Any)

ggtsdisplay(grip$Barcelona,main='Barcelona')

```

```

ggAcf(grip$Barcelona, lag.max = 12,main='Barcelona')
ggPacf(grip$Barcelona, lag.max = 12,main='Barcelona')

casosBCN <- ts(dades$casos, start=c(2010, 6), end=c(2019, 5), frequency=12)
auto.arima(casosBCN)

mod <- Arima(casosBCN, order=c(0,0,2), seasonal=c(0,1,1))
checkresiduals(mod)
pred <- forecast(mod, h=12)

dates <- substr(as.character(grip_predit2$DATE), 1, 7)

# Dibuix de les prediccions juntament amb la regi? de confian?a:
dibuix2 <- ts(matrix(c(pred$mean, pred$upper[,2], pred$lower[,2],
grip_predit2$casosBCN), 12, 4))
colnames(dibuix2) <- c("Weekly cases (mean)", "Upper limit 95%", "Lower limit 95%",
"Registered cases")
ts.plot(x=dibuix2,
      gpars=list(xlab="",
      ylab="Cases", col=c("black", "black", "black", "red"), lty=c(1:4), xaxt="n"))
axis(side=1, at=c(1:12), labels=dates)
legend("topleft", legend=colnames(dibuix2), lty=c(1:4), cex=0.45)

```