Universitat Autònoma de Barcelona
Facultat de Ciències

Entrega 5 Modelització

Autor:

Gerard Lahuerta

1601350

30 de Març del 2023

Contents

1 Exercici 1

Context:
Degut a la crisis sanitària provocada pel virus *SARS-CoV-2*, els hospitals de la regió han començat a saturar-se i col·lapsar.

Per aquest motius, s'ha solicitat un estudi que reculli i analitzi el nombre d'ingressos des de l'inici de la pandemia segons les edats de la població.

Així, en cas de patir alguna crisis sanitària similar poder exercir una millor resposta i habilitar els recursos suficients per a atendre correctament a la població afectada (ja que els recursos que es necesiten depenen de la franja d'edat de la persona afectada).

Informació d'interés:
Per dur a terme aquest estudi s'ha utilitzat la base de dades que ens otorga la generalitat de Catalunya: COVID-2021 Catalunya comarcas.
Si es vol consultar com s'han utilitzat les eines informàtiques per a resoldre el model es pot observar a 2.
Comentar que en l'annex hi ha tota la progarmació feta per a analitzar les dades i els models, molta de la que és recull no és utilitzada peró ha servit per a entendre millor els models pel que s'ha decidit introduïrla.

Objectiu:
Modelitzar el nombre d'ingressos totals (per franjes d'edat) que es poden donar en cas de patir una crisis sanitària similar a la provocada pel virus *SARS-CoV-2*.

Assumpcions:
Per a fer el model s'han tingut en compte les següents assumpcions:

- El comportament d'hospitalitzacions en cas de patir una altre pandemia serà molt similar a la provocada pel *SARS-CoV-2*.
- Els ingressos d'un pacient a un centre sanitari (en aquest cas un hospital) gasten permanentment els recursos destinats al seu cuidat; és a dir, si una persona ingressa dos cops afecta d'igual manera que dues persones ingressessin només un cop.
- Les dades contenen els registres d'ingressos hospitalaris de tots els hospitals del territori durant tota la pandemia; ja que han estat actualitzades des del 22 de de Juliol del 2022, moment on la pandemia sembla ser casi finalitzada.
- Els pacients ingressats consumeixen la mateixa quantitat de recursos.
- El tems d'ingrés a l'hospital és idèntic per a cada pacient.
- Els diversos grups d'edats tenen el mateix nombre de població.
- És completament impossible que hi hagi més ingressos totals que la població total d'un grup d'edat; es tractaria d'un situació endèmica on l'enfermetat no es supera en un temps significativament curt o una enfermetat de tal magnitud que incapacita a la totalitat de la població molt ràpidament.
- Els recursos no tenen problemes de malproduccions.
- Assumim no hi ha erros de càlcul numèric.

Paràmetres:

Per a modelitzar el nombre d'ingressos totals per rang d'edats i poder, així, calcular-lo i trobar la seva incertesa i la discrepància s'han utilitzat els següents paràmetres:

- Grup d'edat: E, $[anys]$
- Població d'un grup d'edat: P, $[persones]$
- Ingressos totals: I, $[persones]$

Recalcar que a partir d'ara es representaran les unitats expressades de la següent manera:

- Unitats d'*anys* es representaran com a.
- Unitats de *persones* es representaran com p

Relacions i estudi previ de les dades:

Per tal de proposar un model suficientment precís al comportament experimentat per la pandemia del *SARS-CoV-2*, primerament estudiarem la relació entre els paràmetres:

- Els ingressos totals d'un subgrup de la població és proporcional al nombre d'individus del subgrup (en aquest cas el subgrup és la franja d'edat): $I \propto P$.
- Els ingressos totals d'un subgrup de la població és proporcional al grup d'edat de la població: $I \propto E$.

També, s'ha observat el nostre conjunt de dades per a deduïr visualment alguna tendència que pugui servir per a modelitzar el comportament d'ingressos totals segons el grup d'edats.

Es pot observar com la distribució de grups dedat és uniforme (tal i com s'ha assumit anteriorment).

També es pot observar com la distribució d'ingressos totals dels diversos grups d'edats és diferent (havent-hi pics d'ingressos molt diferents entre grups d'edat).

Estudiem ara el gràfic de la suma d'ingressos per grups d'edats.

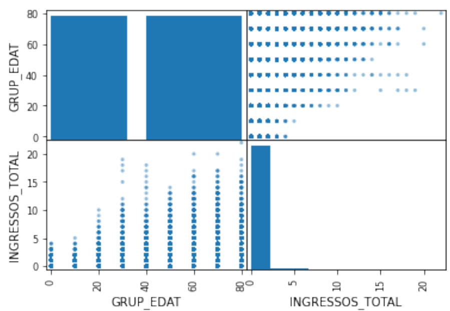


Figure 1: Representació de les dades obtingudes

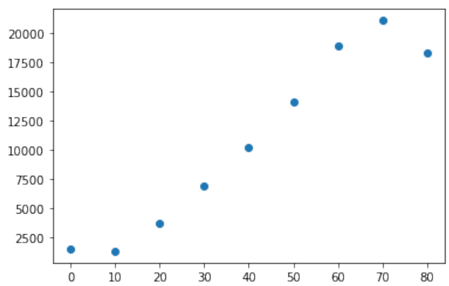


Figure 2: Gràfic dels ingressos totals per grup d'edat

D'aquest gràfic observem de forma més directa les possibles relacions del nombre d'ingressos totals i el grup d'edat.

Es pot visualitzar una tendència similar a la cúbica (tot i que aquesta deducció pot ser fer un *overfitting* de les dades).

Una altre aproximació pot ser mitjançant una recta o una sigmoide.

Model:

Plantejarem ara 3 models diferents i escullirem el millor validant-los amb les dades obtingudes.

- **Model lineal:**

  Una primera aproximació és mitjançant un model lineal ja que s'observa (a la figura 2) com a mida que augmenta el grup d'edat augmenta també el nombre d'ingressos totals.

  A més, és una de les relacions plantejades: $I \propto E$.

  Així doncs obtenim el següent model:

$$I = m \cdot E + n, \text{ on } m, n \in \mathbb{R} \tag{1}$$

  On m és el pendent de la recta i n l'ordenada a l'origen (que es poden estimar segons el nostre model.

  Aquest paràmetres simbolitzen:

  - Quan més població és ingressada en pertànyer a un grup d'edat més elevat és: m[1].
  - El nombre mínim de població que serà afectada per l'efermetat: n.

  D'aquesta forma, mitjançant un anàlisi dimensional deduïm les unitats dels paràmetres:

$$[I] = [m \cdot E + n] \Longrightarrow p = [m] \cdot a + [n] \Longleftrightarrow \left\{ \begin{array}{ccc} [m] & = & p \cdot a^{-1} \\ [n] & = & p \end{array} \right. \tag{2}$$

  Podem estimar aquest paràmetres fent una recta de regressió del nostre model (per així tenir l'error més petit possible i sigui l'ho màxim precís).

  Sinó, també podem estimar-los mitjançant aproximacions del tipus:

  - El paràmetre $m = \frac{\Delta I}{\Delta E}$.
  - El paràmetre $n = I(E = 0)$.

  Per a fer una primera aproximació farem servir aquest segon procés.

[1] Cal recalcar que aquest paràmetre pot servir per a comparar quant més susceptible és una població d'un grup d'edat respecte a un altre.

Mitjançant la informació obtinguda representem el model platejat:

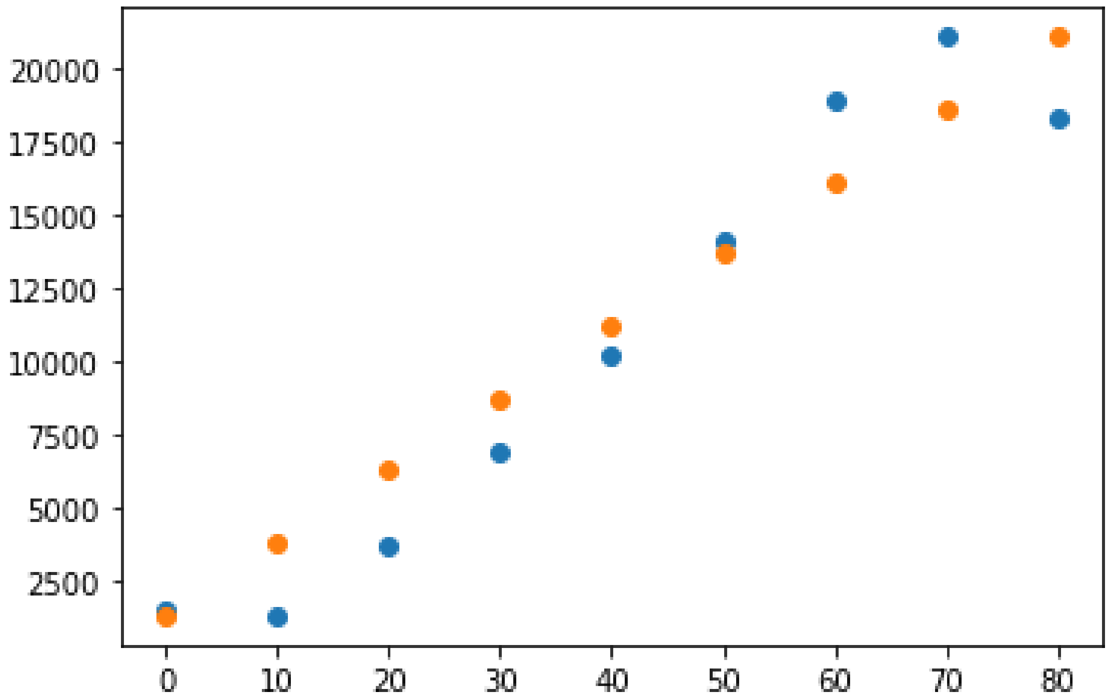


Figure 3: Prediccions del model lineal

Observem fàcilment com el model és prou bo ja que quadra suficientment les caràcterístiques esmentades anteriorment sobre el comportament que hauria de tenir el model.

Tot i així s'observa com és incapaç de predir els mínims i màxims que hi trobem en les dades.

Estudiarem més a fons les capacitats i la viabilitat del model en l'apartat de validació.

- **Model cúbic:**
  Una segona aproximació seria fer un model cúbic: $y = x^3$.

  Aquest model té sentit ja que:

  1. A una edat molt temprana l'ésser humà és molt susceptible a patir greument enfermetats pel que requerirà atenció mèdica.
  2. A una edat relativament madura el cos humà està en el moment més saludable i en bon estat pel que no requerirà d'atenció mèdica en la majoria de casos.
  3. A una edat més adulta el cos humà comença a patir el desgast físic i biològic de l'edat i pot requerir atenció mèdica més habitualment.
  4. A una edat gran l'esser humà pateix de moltes malalties i qualsevol enfermetat pot requerir d'intervenció mèdica per a gestionar l'enfermetat.

  D'aquesta forma observem que existirà un (com a mínim) un punt d'inflexió i un mínim absolut en la funció que parametritzi el nombre d'ingressos segons el grup d'edat.

  Podem doncs definir el següent model.

  $$I = \alpha E^3 + \beta E^2 + \gamma E + \delta, \text{ on } \alpha, \beta, \gamma, \delta \in \mathbb{R} \tag{3}$$

  Els paràmetres ara tenen una interpretació més complexa però la seva estimació és bastant senzilla, ja que, amb les explicacions anteriors podem deduïr el següent:

  - El cos humà té una millor resposta del sistema imnunològic a l'edad pre-adolectent/adoleccent (grup d'edat 10 i 20).
  - El cos humà pateix més enfermetats a l'edat més adulta (grup d'edat 80).
  - El cos humà comença a patir de forma més significativa els efectes de l'edat a partir de la mitjana d'edat de la població (punt d'inflexió), grup d'edat 40.
  - El cos humà pateix significativament a temprana edat (grup d'edat 0).

  Amb aquesta informació es pot plantejar un sistema (mitjançant les dades obtingudes) que ens doni els valors dels paràmetres.

  Seguidament podem fer un anàlisi dimensional per obtenir les dimensions dels paràmetres:

  $$[I] = [\alpha E^3 + \beta E^2 + \gamma E + \delta] \Rightarrow p = [\alpha]a^3 + [\beta]a^2 + [\gamma]a + [\delta] \Leftrightarrow \left\{ \begin{array}{rcc} [\alpha] & = & p \cdot a^{-3} \\ [\beta] & = & p^{-2} \\ [\gamma] & = & p \cdot a^{-1} \\ [\delta] & = & p \end{array} \right. \tag{4}$$

  Una altra forma de trobar els paràmetres és interpolant les dades que disposem, tot i així preferim com a una primera aproximació utilitzar els sistema d'equacions formulat anteriorment.

Representem ara el model plantejat per a verificar el seu comportament:

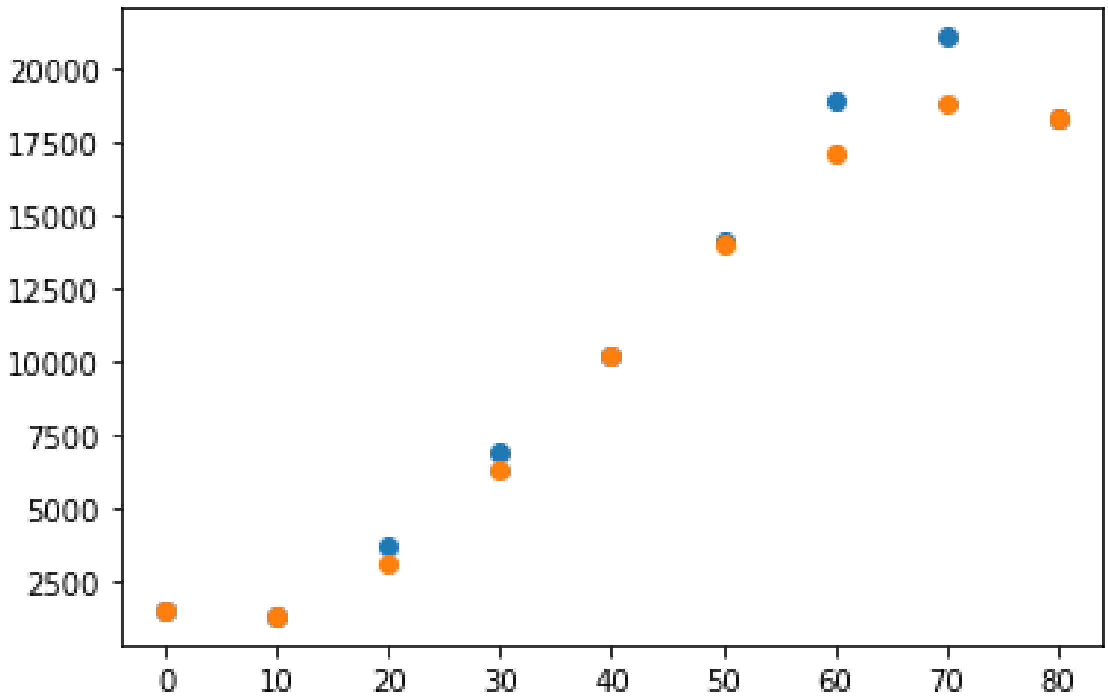


Figure 4: Prediccions del model cúbic

Es pot observar com el model cúbic quadra (de primera instància) prou bé els resultats.

També s'observa com cuadra a la perfecció els punts utilitzats per a calcular els seus paràmetres i, per tant, als punts intermitjos no dona molts bons resultats.

Tot i així es tracta d'un bon model pel que analitzarem la seva fiabilitat amb la validació corresponent.

- **Model logístic:**
  De manera similar als anteriors models, el model logístic el podem plantejar de la següent manera $I \propto P_r$, on P_r és la probabilitat d'ingrés d'un grup d'edat. Aquesta conclusió es pot extreure de:

  - L'augment d'ingressos és proporcional a la probabilitat de ser ingressat un individu d'un grup d'edat.
  - L'augment d'ingressos és proporcional a la població d'un grup d'edat.

  El paràmetre P_r pot ser estimat mitjançant les dades obtingudes com la mitjana d'ingressos per individu d'un grup d'edat (ja que s'ha assumit que aquesta mitjana no pot ser mai superior a 1 ja que significaria que s'ha ingresat més població, en total, de la que hi ha d'un grup d'edat).

  Per altra part, podem modelitzar aquesta mitjana (per a fer-la continua com en la resta de models) com una funció sigmoidea $P_r = K \cdot \frac{1}{1+e^{-C\cdot(E-40)}}$.

  En aquest cas la constant K és la mitjana de grup d'edat mitjà i la constant C és la que determina la *abruptitat* de la funció (si C és molt gran significa que hi ha molta diferència entre els efectes de l'enfermetat en individus joves i adults, per contra de quan té valors petits que significa que no té efectes tan radicals).

  Fent un anàlisi dimensional podem observar que:

  $$[I] = [P \cdot K \cdot \frac{1}{1+e^{-C\cdot E}}] \Longrightarrow p = p \cdot [k] \cdot \left[1 + e^{-C\cdot(E-40)}\right]^{-1} \Leftrightarrow \left\{ \begin{array}{rcl} [K] & = & 1 \\ [C] & = & a^{-1} \end{array} \right. \tag{5}$$

  Comentar també, que aquests paràmetres poden ser estimats de la següent forma:

  - El paràmetre K és la mitjana del grup d'edat mitjà.
  - El paràmetre C pot ser estimat sabent la mitjana d'ingressos totals del grup tal que s'assoleix el mínim d'ingressos totals:

    $$C = \frac{\log\left[\dfrac{I_m}{K - I_m}\right]}{E - 40} \quad ^{2}$$

    On I_m és la mitjana d'ingressos totals del grup d'edat.

[2] Comentar que, tot i haber-se omès la base del logaritme, aquest es tracta d'un logaritme neperià

Mostrem a continuació el nostre model graficat:

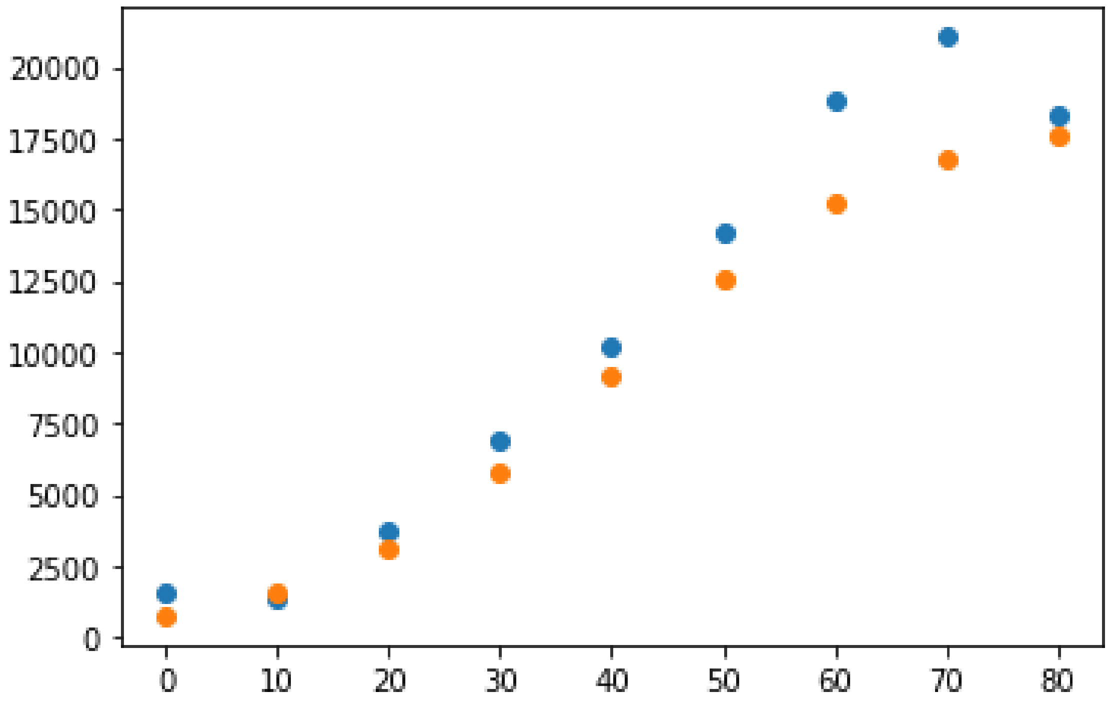


Figure 5: Prediccions del model logístic

Es pot observar com el model logístic no té bones capacitats de predir els ingressos totals de grups d'edats superiors a l'adolescència.

Tot i així, sembla aproximar-se prou bé (ignorant la gran diferència entre les dades predites i obtingudes per a als grups d'edats de 60 i 70 anys).

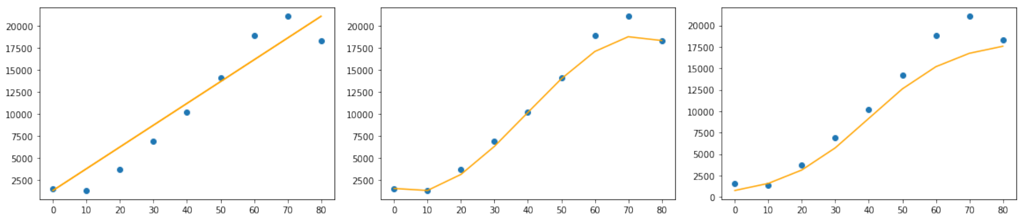
Tot així es seguirà analitzant la discrepància del model per tal de decidir si és suficientment bo com per a tenir-lo en compte o és preferible utilitzar un altre dels plantejats anteriorment.

Anàlisi d'incerteses:

Cal comentar que, en el nostre cas, degut a ser dades d'ingressos hospitalaris, no tenen incerteses ja que tots els ingressos han de ser documentats i són classificats.

D'aquesta forma podem asumir que els nostres paràmetres (que han sigut calculats mitjançant les dades, no tenen errors ni incerteses; pel que tots els models tenen una incertesa del 0%.

Simulació dels resultats:
Fem una simulació dels resultats dels diversos models per comparar visualment el seu comportament i com es diferencien respecte a les dades obtingudes.



(a) *Model lineal* (b) *Model cúbic* (c) *Model logístic*

Figure 6: Simulació dels diversos models

Es pot observar com tots els models tendeixen a donar unes prediccions suficientment bones dels ingressos totals. Calculem l'error dels models i la incertesa de l'estimació.

- **Model lineal:**

Grup d'edats	**0**	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**
Simulació	1347	3812	6277	8741	11206	13671	16136	18600	21065
Dades	1574	1347	3739	6921	10181	14155	18857	21065	18323
Error	227	-2464	-2537	-1820	-1025	484	2722	2465	-2742

Table 1: Analisi de la discrepància del model lineal

És interessant el fet de que en la meitat de les prediccións tingui un error negatiu (sobreestimi) ja que així ens assegurem tenir suficients recursos en cas de tornar a patir una situació com la estudiada.

- **Model cúbic:**

Grup d'edats	**0**	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**
Simulació	1574	1347	3167	6343	10181	13990	17079	18753	18323
Dades	1574	1347	3739	6921	10181	14155	18857	21065	18323
Error	0	0	572	578	0	165	1778	2311	0

Table 2: Analisi de la discrepància del model cúbic

És interesant el fet de que la majoria de les prediccions tenen erros bastants petits i poden ser corretgits fàcilment.

També, s'ha de tenir en compte que per als grups d'edat amb error 0 és degut a que s'han utilitzat per a resoldre el sistema d'equacions per trobar els valors dels paràmetres.

Per tant, cal contemplar una certa discrepància respecte la realitat en aquest punts que el nostre model i dades no mostren ja que en tots els casos que es pugui patir una situació sanitària similar aquest model no predirà amb total seguretat un valor sense error.

- **Model Logístic:**

Grup d'edats	**0**	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**
Simulació	750	1574	3139	5727	9162	12596	15184	16749	17572
Dades	1574	1347	3739	6921	10181	14155	18857	21065	18323
Error	823	-227	600	1194	1020	1559	3673	4316	751

Table 3: Analisi de la discrepància del model logístic

És interesant el fet de que en la meitat de les prediccións tingui un error negatiu (sobreestimi) ja que així ens assegurem tenir suficients recursos en cas de tornar a patir una situació com la estudiada.

Com que el model no conté errors dels paràmtres ni de càlculs; l'únic efecte sobre l'incertesa del model és l'incertesa de les dades:

$$u_v^2 = u_d^2 \Rightarrow u_v = \pm\sigma(I) = \pm 7286 \tag{6}$$

En el nostre cas com no podem tenir població negativa o propera a 0, ja que no es possible o significaria que no esten en un estat de pandemia; podem asumir que l'incertesa del model és $u_v = +7286$.

Amb aquestes circumstàncies, el millor model que s'ha proposat és el cúbic; ja que dona els menors errors i sembla adaptar-se millor a les circumstàncies donades.

També, en cas de tenir una situació similar a la del covid però amb major o menor infecció, podem multiplicar aquest model per T, la divisió entre la taxa d'infecció de la enfermetat que s'està patint i la del *SARS-CoV-2.*

D'aquesta forma es conclou que el millor model és el cúbic per a predir quanta gent serà ingressada en total pels diversos grups d'edats en cas de patir una situació sanitaria similar a la provocada pel *SARS-CoV-2* i que pot ser modificada per a una nova enfermetat simplement multiplicant-la pel factor *taxa d'infecció de l'enfermetat* entre *taxa d'infecció del SARS-CoV-2.*

Anàlisi crítica:

Cal comentar abans d'extreure conclusions que el model ha sigut comparat amb resultats extrets de dades direcates recollides de la pandemia provocada pel *SARS-CoV-2* i, per tant, el model concorda bastant amb aquestes dades però no té perquè predir correctament els ingressos totals per grup d'edats derivats d'una situació sanitaria similar d'un altre virus.

És recomana l'ús d'aquest model per a obtenir una aproximació o estimació a priori de la situació que es pot patir en l'àmbit sanitari per tal de tenir un equipament i medicació mínim per a afrontar de forma inicial i suficientment bona una nova crisis sanitària i no es produeixin situacions similars a les obtingudes en hospitals que han sigut sobrepassats de malalts i que no han pogut obtenir una atenció mínima suficient o accés a material sanitari necesari per al seu tractament.

Comentar també que aquest model pot també disternir de la realitat ja que s'han prés certes assumpcions que no succeeixen en la realitat (materials sense imperfeccions, població amb edats uniformament repartides, que els materials necesaris per al tractament de pacients és similar entre els pacients amb mateixes edats, etc...).

En cas de voler fer un estudi més precís cal tenir en consideració els següents punts:

- La taxa d'infecció per rang d'edats (ja que certes edats o grups de població solen ser més suceptibles de ser enfermats i, per tant, de requerir ingrés hospitalari).
- La qualitat dels materials necessaris i la quantitat necessària per pacient (per expecificar els material necessaris i no reduïr l'estudi a saber quanta gent aproximadament serà hospitalitzada).
- La població no és uniforme (pel que a més població d'una edat més gent serà ingressada).
- Els paràmetres així com els càlculs tenen errors que s'han de considerar.
- Els extrangers que hi resideixen o venen de vacances (temporalment per diversos motius) també poden ser ingressats i cal tenir en compte aquesta població.
- A les dades hi ha individus no categoritzats en el grup d'edat al que pertanyen que han de ser estimats d'una forma més eficient que no pas assumint que la població és uniforme i, per tant, pertoquen els mateixos per cada grup d'edat.

2 Annex:

2.1 Imports i funcions generals

```
import numpy as np
import pandas as pd
import scipy
from scipy import stats
import matplotlib.pyplot as plt
from IPython.display import Math, display
import sympy as sp
from sympy.physics.units.systems import SI
from sympy.physics.units import meter, second, liter,
                               hour, degree, convert_to
from sympy import sin, cos, pi, Abs, N

sp.init_printing()

def show(*args):
    out = ""
    for arg in args:
        if isinstance(arg, (sp.Expr, sp.Eq)):
            arg = sp.latex(arg)
        else:
            arg = str(arg)
        out += arg
    display(Math(out))

qq = "\quad "
```

2.2 Descarga i gestió del dataset

```
# Carrega les dades
import json

j = json.loads(bin_data.decode())

# Crea un data frame
import pandas as pd

cols = [
    j['meta']['view']['columns'][i]['name'] for i in range(21)
]
print(cols)

df = pd.DataFrame(j['data'], columns=cols)

# Acomoda les dades
df['DATA'] = pd.to_datetime(df['DATA'])
```

```
camps = ['CASOS_CONFIRMAT', 'INGRESSOS_TOTAL', 'GRUP_EDAT']


edats = {}
llista_edats_df = df['GRUP_EDAT'].unique().tolist()

for i in llista_edats_df:
    try:
        j = int(i.split(" ")[0])
        edats[i] = j
    except:
        edats[i] = -10

BD = df[camps].replace(edats)


for c in camps:
    BD[c] = BD[c].astype(float)

l = pd.plotting.scatter_matrix(BD)

# OBSERVEM EL CAS INGRESSOS_TOTALS-GRUP_EDAT
BD_aux = BD.copy()
BD_aux.drop(BD[(BD['GRUP_EDAT'] < 0)].index, inplace=True)


l = pd.plotting.scatter_matrix(BD_aux[
    ['GRUP_EDAT',"INGRESSOS_TOTAL"]])
```

2.3 Calculs dels model

```
#Observaci  dels ingressos totals mitjos per habitant
# segons la seva franja de edat
mitja_ingres_edat = {}
edats = BD_aux['GRUP_EDAT'].unique().tolist()

for i in edats:
    mitja_ingres_edat[i] = BD_aux[(BD_aux['GRUP_EDAT'] == i)]
    ["INGRESSOS_TOTAL"].mean()

plt.scatter(edats, mitja_ingres_edat.values() )

#Observaci  de la suma ingressos totals per habitant
# segons la seva franja de edat
per_ingre_edat = {}
edats = BD_aux['GRUP_EDAT'].unique().tolist()

for i in edats:
```

```
    per_ingre_edat[i] = BD_aux[(BD_aux['GRUP_EDAT'] == i)]
    ["INGRESSOS_TOTAL"].sum()
print(per_ingre_edat)

plt.scatter(edats, per_ingre_edat.values() )

# Declaraci  model log stic
from sympy.abc import E, K, C

IM = K/(sp.exp(-C*(E-40))+1)

show("I␣=␣",IM)

# a llament C:
I = sp.symbols("IM")
eq = sp.Eq(I,IM)

c = sp.solve(eq,C)[0]
show("C␣=␣",c)

E = 10
K = mitja_ingres_edat[80]
e = 0
valors = {"E": E, "K": K, "IM": mitja_ingres_edat[e]}
C = sp.re(c.subs(valors))
show(K, qq, sp.re(C))

per_ingre_edat = {}
edats = BD_aux['GRUP_EDAT'].unique().tolist()

for i in edats:
    per_ingre_edat[i] = BD_aux[(BD_aux['GRUP_EDAT'] == i)]
    ["INGRESSOS_TOTAL"].sum()
print(per_ingre_edat)

plt.scatter(edats, per_ingre_edat.values() )

E = []
I = []
P = per_edat[0]
values = {"E": 0, "K": K, "C": C}
for i in range(0,9):
    E.append(i*10)
    values["E"] = i*10
    I.append(P*IM.subs(values))
plt.scatter(E, I )

per_ingre_edat = {}
edats = BD_aux['GRUP_EDAT'].unique().tolist()
```

```python
for i in edats:
    per_ingre_edat[i] = BD_aux[(BD_aux['GRUP_EDAT'] == i)]
    ["INGRESSOS_TOTAL"].sum()
print(per_ingre_edat)

plt.scatter(edats, per_ingre_edat.values() )

E = []
I = []
for i in range(0,9):
    E.append(i*10)
    val = CI.subs(param)
    val = val.subs("E", i*10)
    I.append(val)
plt.scatter(E, I)

from sympy.abc import E
maxi = max(per_ingre_edat.values())
mini = min(per_ingre_edat.values())
m = (maxi -mini)/80

n = mini

RI = m*E+n

per_ingre_edat = {}
edats = BD_aux['GRUP_EDAT'].unique().tolist()

for i in edats:
    per_ingre_edat[i] = BD_aux[(BD_aux['GRUP_EDAT'] == i)]
    ["INGRESSOS_TOTAL"].sum()
print(per_ingre_edat)

plt.scatter(edats, per_ingre_edat.values() )

E = []
I = []
for i in range(0,9):
    E.append(i*10)
    val = RI.subs("E", i*10)
    I.append(val)
plt.scatter(E, I)

incertesa = sum((BD[(BD['GRUP_EDAT'] == -10)]
["INGRESSOS_TOTAL"]))/9
print(incertesa)

incer = {}
```

```
for i in range(0,9):
    incer[i*10] = incertesa/BD_aux[(BD_aux['GRUP_EDAT'] == i*10)]
    ["INGRESSOS_TOTAL"].sum()
print(incer)

from sympy.abc import m, n, E
RI = m*E+n
Cm = m/RI * sp.diff(RI,m)
show(Cm)

Cn = n/RI * sp.diff(RI,n)
show(Cn)

ur_I = sp.sqrt( (Cm.subs("E", 80)*0.2)**2 +
                (Cn.subs("E", 0)*1.6)**2 )
show(ur_I)

maxi = max(per_ingre_edat.values())
mini = min(per_ingre_edat.values())

valors = {"m": (maxi-mini)/80, "n": mini}
ur_I.subs(valors)

from sympy.abc import a, b, c, d, E
CI = a*E**3 + b*E**2 + c*E + d

eq = []
eq2 = []
for i in [0,10,40,80]:
    eq.append(sp.Eq(CI.subs("E",i), per_ingre_edat[i]*(1+incer[i])))
    eq2.append(sp.Eq(CI.subs("E",i), per_ingre_edat[i]))


param = sp.solve(eq,[a,b,c,d])
param2 = sp.solve(eq2,[a,b,c,d])
show(param)
show(param2)

for i in param:
    print(param[i]/param2[i])

CI = a*E**3 + b*E**2 + c*E + d

C_d = d/CI * sp.diff(CI,d)
show(C_d)
param["E"] = 80
C_d.subs(param)

# Declaraci  model log stic
```

```python
from sympy.abc import E, K, C

IM = K/(sp.exp(-C*(E-40))+1)

show("I = ",IM)


# a llament C:
I = sp.symbols("IM")
eq = sp.Eq(I,IM)

c = sp.solve(eq,C)[0]
k = sp.solve(eq,K)[0]
show("C = ",c)

E = 10
K = mitja_ingres_edat[80]
e = 0
valors = {"E": E, "K": K, "IM": mitja_ingres_edat[e]}
C = sp.re(c.subs(valors))
show(K, qq, sp.re(C))


E = 10
K2 = mitja_ingres_edat[80]*(1+0.14)
e = 0
valors2 = {"E": E, "K": K2, "IM": mitja_ingres_edat[e]*(1+1.6)}
C2 = sp.re(c.subs(valors2))
show(K, qq, sp.re(C))
show(K2, qq, sp.re(C2))

print(K2/K, sp.re(C2)/sp.re(C))

from sympy.abc import E, K, C

IM = K/(sp.exp(-C*(E-40))+1)

Cc = C/IM * sp.diff(IM,C)
Ck = K/IM * sp.diff(IM,K)

show(Cc, qq, Ck)

valors1 = {"E": 10, "K": 0.259, "IM": mitja_ingres_edat[0],
            "C": 0.079}
valors2 = {"E": E, "C": 0.079, "IM": mitja_ingres_edat[80]}

print(Cc.subs(valors1))

show(sp.sqrt( (Cc.subs(valors1)*0.59)**2 +
```

```
                    (Ck.subs(valors2)*0.14)**2  ))

#SIMULACI
from sympy.abc import E
maxi = max(per_ingre_edat.values())
mini = min(per_ingre_edat.values())
m = (maxi -mini)/80

n = mini

RI = m*E+n



per_ingre_edat = {}
edats = BD_aux['GRUP_EDAT'].unique().tolist()

for i in edats:
    per_ingre_edat[i] = BD_aux[(BD_aux['GRUP_EDAT'] == i)]
    ["INGRESSOS_TOTAL"].sum()
print(per_ingre_edat)

plt.scatter(edats, per_ingre_edat.values() )

E = []
I = []
for i in range(0,9):
    E.append(i*10)
    val = RI.subs("E", i*10)
    I.append(val)
plt.plot(E, I, "-", color ="orange")

for i in range(0,9):
    print(I[i],per_ingre_edat[i*10], per_ingre_edat[i*10]-I[i])

from sympy.abc import a, b, c, d, E
CI = a*E**3 + b*E**2 + c*E + d

eq = []
eq2 = []
for i in [0,10,40,80]:
    eq.append(sp.Eq(CI.subs("E",i), per_ingre_edat[i]*(1+incer[i])))
    eq2.append(sp.Eq(CI.subs("E",i), per_ingre_edat[i]))


param = sp.solve(eq,[a,b,c,d])
param2 = sp.solve(eq2,[a,b,c,d])
show(param)
show(param2)
```

```
per_ingre_edat = {}
edats = BD_aux['GRUP_EDAT'].unique().tolist()

for i in edats:
    per_ingre_edat[i] = BD_aux[(BD_aux['GRUP_EDAT'] == i)]
    ["INGRESSOS_TOTAL"].sum()

plt.scatter(edats, per_ingre_edat.values() )

E = []
I = []
for i in range(0,9):
    E.append(i*10)
    val = CI.subs(param2)
    val = val.subs("E", i*10)
    I.append(val)
plt.plot(E, I, "-", color = "orange")

for i in range(0,9):
    print(I[i],per_ingre_edat[i*10], per_ingre_edat[i*10]-I[i])

# Declaraci  model log stic
from sympy.abc import E, K, C

IM = K/(sp.exp(-C*(E-40))+1)

show("I␣=␣",IM)


# a llament C:
I = sp.symbols("IM")
eq = sp.Eq(I,IM)

c = sp.solve(eq,C)[0]
k = sp.solve(eq,K)[0]
show("C␣=␣",c)

E = 10
K = mitja_ingres_edat[80]
e = 0
valors = {"E": E, "K": K, "IM": mitja_ingres_edat[e]}
C = sp.re(c.subs(valors))
show(K, qq, sp.re(C))


E = 10
K2 = mitja_ingres_edat[80]*(1+0.14)
e = 0
```

```
valors2 = {"E": E, "K": K2, "IM": mitja_ingres_edat[e]*(1+1.6)}
C2 = sp.re(c.subs(valors2))


per_ingre_edat = {}
edats = BD_aux['GRUP_EDAT'].unique().tolist()

for i in edats:
    per_ingre_edat[i] = BD_aux[(BD_aux['GRUP_EDAT'] == i)]
    ["INGRESSOS_TOTAL"].sum()

plt.scatter(edats, per_ingre_edat.values() )

E = []
I = []
P = per_edat[0]
values = {"E": 0, "K": K, "C": C}
for i in range(0,9):
    E.append(i*10)
    values["E"] = i*10
    I.append(P*IM.subs(values))
plt.plot(E, I, "-", color = "orange" )

for i in range(0,9):
    print(I[i],per_ingre_edat[i*10], per_ingre_edat[i*10]-I[i])

print(np.sqrt(np.var(list(per_ingre_edat.values()))))
```