

# Development of an automatic tool for periodic surveillance of actuarial and demographic indicators

Daniel Alonso, María Luz Durbán Reguera, Bernardo D'Auria

July-August 2021

## Contents

<b>Abstract</b>	<b>2</b>
<b>Introduction</b>	<b>2</b>
<b>Objectives</b>	<b>2</b>
<b>What problem does the application solve?</b>	<b>3</b>
<b>Project repository tree structure</b>	<b>3</b>

# Abstract

As a result of the COVID-19 pandemic and its large impact across Spain, the monitoring of demographic measures as a direct result of deaths related to such pandemic and future similarly deadly events has become increasingly important. It is intended with this project to develop a tool in order to easily monitor a selection of demographic measures relating to collective deaths of individuals as a result of relevant worldwide events like the one mentioned previously. The tool consists of a shiny dashboard (developed in R) where such measures are displayed in different visualizations across time.

# Introduction

The COVID-19 pandemic has led to a widespread and noticeable temporary increase in mortality and reduction in life expectancy throughout Spain. This arises a need to monitor these demographic measures more closely and in real time.

This project consists of a shiny dashboard with several features:

- Visualizing several mortality metrics:
  - Excess mortality
  - Cumulative mortality rate
  - Cumulative relative mortality rate
  - Mortality improvement factor
- Visualizing life expectancy and constructing life tables
- Visualizing a map of Spain with the previous metrics per autonomous community (CCAA)

All metrics are calculated weekly with data stretching back as far back as 2010.

# Objectives

- Provide a simple-to-use, web-based, OS-agnostic tool to compute and visualize common mortality and life expectancy metrics in time series plots/maps
- Provide the user the ability to customize the plot parameters significantly
- Provide the user the ability to download the plots and the data (with or without filtering)

- Allow the user to update and push the data to the corresponding github repository hosting the data from within the application
- Have data updated in real-time from the official Spanish sources and Eurostat (also provided by INE)

## What problem does the application solve?

As we saw during the COVID-19 pandemic, the most widely publicized measures shown to the public in order to explain the status of the pandemic and the country as a whole were always related to incidence of the virus and death counts/recovery counts.

## Project repository tree structure

First of all, the project consists of two repositories and their respective **subfolders/files**:

- **The main project repository** (*tfm\_uc3m*)
  - **api**: Contains all files related to the querying, acquisition, manipulation of new data from the two aggregated data sources (INE and Eurostat), along with various log files used to keep track of some metadata from the data sources (last updated data and provisional data).
    - \* **logs**
      - *earliest\_eurostat\_provisional.log*: Eurostat marks data as provisional when it can still be updated, as new deaths might occur. This file logs the earliest data entry marked as provisional.
      - *last\_eurostat\_update.log*: This file logs the last week for which there is available and updated data from Eurostat for Spain within the database *demo\_r\_mwk2\_05*.
      - *last\_ine\_update.log*: This file logs the last week for which there is available and updated data from INE for table No. *9681*.
    - \* *dbs\_check.py*: This script obtains a small sample of data from Eurostat and Ine and checks when the earliest provisional data point is, what is the latest obtainable week of data from both Eurostat and INE and writes it in the log files contained within **logs**.
    - \* *functions.py*: This script contains all the functions used for fetching, manipulating and outputting the data obtained from the aggregated data sources' respective APIs (Eurostat and INE).

- \* *query.py*: This script will run a pre-constructed pipeline where the data is systematically queried per age group, manipulated and where new entries are added to the *.csv* files contained within the **data** folder.
- **dashboard**
  - \* **www**
    - **maps**
    - **map\_shapefiles**
    - **map\_shapefiles\_ggplot**
- **data**
- **docker**
- **docs**
- **paper**
- *README.md*
- **The data repository** (*tfm\_uc3m\_data*)
  - **ccaa\_guide**
    - \* *guide\_ccaa.json*
    - \* *guide\_ccaa.txt*
    - \* *guide\_ccaa\_python\_dict.p*