Alma Mater Studiorum · Università di Bologna

**Scuola di Scienze**
**Dipartimento di Fisica e Astronomia**
**Corso di Laurea in Fisica**

# Analysis of COVID-related symptoms from a database of Italian Tweets

**Relatore:**
**Prof. Daniel Remondini**

**Co-relatore:**
**Dott. Francesco Durazzi**

**Presentata da:**
**Kim Maria Damiani**

Anno Accademico 2020/2021

**Sommario**

Twitter è un social network diffuso che consente la trasmissione di informazioni in tempo reale e viceversa costituisce una fonte di dati testuali ad accesso libero e omogenei in lunghezza. Proponiamo l'analisi di un database di tweet italiani contenenti i più comuni sintomi di COVID-19 autoriferiti, per indagare l'evoluzione della pandemia in Italia dalla fine di settembre 2020 alla fine di gennaio 2021. Disponendo di un database che contiene parole legate a febbre, tosse e mal di gola, filtriamo manualmente i tweet che descrivono realmente sintomi attribuibili al COVID e discutiamo l'utilità di tale selezione manuale. Successivamente confrontiamo le nostre serie temporali con i dati giornalieri dei nuovi ricoveri in Italia, con l'obiettivo di costruire un semplice modello di regressione lineare, che incorpori il ritardo osservato dalla pubblicazione dei tweet menzionanti i singoli sintomi ai nuovi ricoveri. Discutiamo sia i risultati che i limiti della regressione lineare, poiché i nostri dati suggeriscono che la relazione tra le serie temporali di tweet con sintomi e i nuovi ricoveri varia verso la fine dell'acquisizione.

**Abstract**

Twitter is a highly popular social media which on one hand allows information transmission in real time and on the other hand represents a source of open access homogeneous text data. We propose an analysis of the most common self-reported COVID symptoms from a dataset of Italian tweets to investigate the evolution of the pandemic in Italy from the end of September 2020 to the end of January 2021. After manually filtering tweets actually describing COVID symptoms from the database - which contains words related to fever, cough and sore throat - we discuss usefulness of such filtering. We then compare our time series with the daily data of new hospitalisations in Italy, with the aim of building a simple linear regression model that accounts for the delay which is observed from the tweets mentioning individual symptoms to new hospitalisations. We discuss both the results and limitations of linear regression given that our data suggests that the relationship between time series of symptoms tweets and of new hospitalisations changes towards the end of the acquisition.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

In a connected world where social media use is part of daily life, the field of social media analysis is in full expansion, one of its branches being studying the evolution of users response to an emergency, which exploits their rapid reaction on instant messaging platforms such as Twitter. Compared to previous studies in this field, such as the characterisation of the information network for the outbreak of Zika in the US in 2015-16, COVID-19 represents an unprecedented phenomenon in terms of time duration and global impact, being the most severe global crisis to date whose public conversation can be studied in real time.

The direct, spontaneous expression of users is an invaluable source of information for analysis of immediate social impact. A recent work studied the complex network structure of retweets related to the current pandemic in the English twittersphere [4]. Super-communities were identified according to the prevalent user categories and the degree of internationality (international sci-health, national elite, political actors and other); their evolution in terms of growth and activity as well as their interaction patterns over time were assessed. The above work highlighted the key role of sci-health experts as a trusted source of information at the outbreak of the largely unknown pandemic, as well as the increase in communities size and activity and attention shift towards national elite and politics, simultaneously with the explosion of cases. It was found that this attention shift intensified as time passed and the pandemic changed from an external news event to a local reality that had important health and social effects. This trend highlighted the growing politicisation of the debate in parallel with the imposition of lockdown in several countries. This suggests it is important that scientists and health institutions maintain a regular tweeting activity and reshape their content to involve themselves in local discussions, targeting and merging with the stable national Twitter communities. Otherwise, an exclusively scientific discussion risks losing audience when the health crisis starts to heavily impact society and to feed country-specific debates. We note how there are possible implications for information dissemination along the unfolding of long-term events like epidemic diseases on a world-wide scale. Another work conducted

sentiment analysis on Australian COVID related tweets regarding vaccines and identified the prevalent topics (attitudes toward COVID-19 and its vaccination, advocating infection control measures against COVID-19, and misconceptions and complaints about COVID-19 control) and emotions (trust and anticipation on the positive side, fear on the negative)[7]. This approach may have implications for an efficient campaign to spread awareness and trusted information on COVID vaccine focused on discussing and solving doubts and fears of the population.

These analyses, as well as ours, are possible thanks to the Twitter API, a service to retrieve public tweets (i.e. those published by public accounts) filtering them by date, language, location and keywords. We chose Twitter because it provides open data, in that free access can be requested by anyone for research purposes in view of free and reproducible science. Moreover, with its 280 characters limit per post, the dimension of the signal is comparable for all samples, which permits a targeted retrieval of information, as well as constituting an advantage in machine learning analysis. Thus, with 12.8 million Italian users (as of 2020), Twitter is a powerful, innovative tool that we chose to use to investigate the evolution of the pandemic in Italy. Our thesis analysed a dataset of tweets previously selected from the Twitter API according to the presence of keywords related to fever, cough and sore throat, from the end of September 2020 to the end of January 2021. We first performed manual annotation, which was necessary to filter according to context tweets (1) actually mentioning users or other people experiencing symptoms associated to COVID within 3 days from the publication date and (2) mentioning taking COVID tests, which were classified according to the outcome if mentioned. After pre-processing of the tweets, we compared the occurrence of the most frequent words and pair of words for symptoms tweets with respect to the rest of the tweets and we visualised the co-occurrence network in both cases. We applied Latent Dirichlet Allocation to all our tweets and discussed if it highlights differences in terms of document topic probabilities for true and fake symptoms.

We chose as time series for further investigation our total volume of pre-selected tweets, any symptom, individual symptoms, COVID tests with symptoms. We investigated whether descriptive statistics aggregated on the whole temporal extension of our study were stable in time, by computing them in a monthly window. Furthermore, since the direct information from the users is a complementary approach to the official data provided by health authorities, we decided to compare our time series with data of new daily hospitalisations. Since differences in the respective time development were to be expected (tweets mentioning symptoms preceding hospitalisations), we measured similarity via Pearson's linear correlation $\rho$ with lag $l$. We found the following optimal values, where the lag $l$ is the delay (in days) of new hospitalisations with respect to our time series: ($\rho^* = 0.93$, $l^* = 17$) for cough, ($\rho^* = 0.91$, $l^* = 17$) for fever, ($\rho^* = 0.97$, $l^* = 22$) for sore throat; globally, true symptoms have ($\rho^* = 0.95$, $l^* = 20$). The behaviour of the correlation coefficient near the maximum allowed us to estimate the propagation of time uncertainty on the optimal lag. In fact, by associating symptoms to the publication

date within a time interval of 3 days, we inevitably introduced time uncertainty in the daily series. Finally, we performed linear regression using individual symptoms tweets as predictors to estimate the new daily hospitalisations $\widehat{y}$, after randomly dividing the dataset into a training and validation set. Multivariate regression suggests sore throat tweets lagged of 22 days are the most predictive symptom for our dataset, as it dominates among coefficients. For this reason, we report as result the univariate regression with sore throat tweets as predictor: $\widehat{y} = (808 \pm 25) + (42.9 \pm 1.1)x_s$, with RMSE$= 120$. We assessed that performing univariate regression with each filtered symptom decreased the RMSE compared to not filtered symptoms, which supports the usefulness of manual annotation, especially for cough and fever tweets. We stress that this model has a limited validity in time due to the non stationarity of the pandemic evolution.

# Chapter 2

# Statistical methods

## 2.1 Dataset and rules for manual selection of tweets

Our database is composed of 7618 public tweets acquired from September 30th 2020 to January 26th 2021 (inclusive) for a total of 119 days. The tweets of our database had been previously selected through the public *Twitter API* according to the presence of words describing COVID symptoms. The keywords (not necessarily consecutive words) are found in Tab. 2.1. We analysed only the columns containing text and time of publication. Tweets were acquired in real time after a single initial request; the only time frame when no tweets were collected was from 2020-12-17 17:30 to 2020-12-18 08:00 due to a disconnection of the server.

| Keywords |
|---|
| ho temperatura 38/39/40/41 |
| ho febbre 38/39/40/41 |
| mi febbre 38/39/40/41 |
| ho tosse |
| ho mal di gola |
| ho tossire |
| tossendo |

Table 2.1: Keywords used to collect tweets potentially describing symptoms from Twitter API.

**Symptoms** The self-reported symptoms of COVID-19 analysed in this thesis are *cough* ("tosse"), *fever* ("febbre") of or above 38°C and *sore throat* ("mal di gola"). Their mention was not considered in case of a clear non infectious origin, such as chronic cough (ex. due to smoking or reflux) or sore throat due to prolonged speaking or screaming. These cases will be referred to as "fake symptoms" to distinguish them from tweets that have been recognised as actually reporting symptoms, which will be referred to as "true symptoms" or simply "symptoms". Cold has not been considered as a symptom on its own, consistently with the fact that it had not been included in the keywords to create the data set; however, its occurrence was then analysed in combination with sore throat and cough. A second categorisation is the self-reported performing of tests, which has

been divided in positive, negative or unknown outcome.

**People** It is to be noted that in case a tweet mentioned symptoms and/or tests regarding more than one person, it was counted as a single occurrence. The people mentioned in the tweet were not necessarily traceable through the user (ex. sick person in public transportation or sick customer in a shop the user had visited in the last 3 days). Tweets referring to self-report of symptoms by public figures (ex. politicians, athletes, influencers, TV people) have been disregarded, as this was usually mentioned in several tweets containing links to news articles or general comments. Twitter search has been used where needed (and when possible) to solve doubts regarding the context of the tweet, especially to exclude the sarcastic report of symptoms.

**Time** Since the objective is to study the time evolution of present self-reported symptoms, a time constraint of *3 days* has been chosen with respect to the date of publication of the tweets (i.e. symptoms experienced up to 3 days earlier). Therefore, references to confirmed infections or death in an undefined past have been disregarded. For the same reason, lack of smell or taste has not been counted as symptoms nor positivity to serological tests has been considered, as their occurrence can extend through a much longer period of time compared to the duration of the infection.

## 2.2 Preprocessing

Before performing text analysis, cleaning tweets was needed to reduce the dimensionality of the problem. We converted the tweets to lower case and removed mentions and punctuation (this included emojis). We tokenised the tweets and performed *stop words* removal using a list of Italian articles, pronouns, prepositions and filler words. After stop words removal, we were left with 75% of the initial tokens. We made sure neither words containing negation nor time references were included in the stop words, as we were interested in capturing the negation of symptoms and in time indicators within 3 days of the publication date, respectively. We then performed *lemmatisation*, which is the process of grouping together the inflected forms of a word so they can be analysed as a single item, identified by the word's lemma, or dictionary form. While most of our analysis was performed in MATLAB, since MATLAB does not include Italian among supported languages, we used the lemmatiser "Treetaggerwrapper" in Python. We preferred a lemmatiser to an aggressive stemmer (i.e. a tool which would reduce inflected words to their word stem, base or root form) for better preservation of the meaning.

## 2.3 Pointwise mutual information (PMI)

The *mutual information* (MI) of two discrete random variables $X$ and $Y$ is defined as

$$\mathrm{I}(X;Y) = \sum_{y\in\mathcal{Y}} \sum_{x\in\mathcal{X}} p_{(X,Y)}(x,y) \log\left(\frac{p_{(X,Y)}(x,y)}{p_X(x)\,p_Y(y)}\right) \tag{2.3.1}$$

where $p_{(X,Y)}$ is the joint probability mass function of $X$ and $Y$, and $p_X$ and $p_Y$ are the marginal probability mass functions of $X$ and $Y$ respectively. It quantifies the "amount of information" which can be obtained about one random variable by observing the other random variable; more precisely, the reduction on the uncertainty of one due to the observation of the other. If the logarithm is in base 2, mutual entropy is measured in bits. In order to understand its interpretation, we first define entropy and conditional entropy within information theory.

*Entropy* $H(X)$ of a random variable $X$ is a measure of its uncertainty, in other words a measure of how much "choice" is involved in the selection of an outcome given the probability distribution of the event. As Shannon observed [11], if there is such a measure, say $H(p_1,p_2,\cdot,p_n)$, it is reasonable to require of it the following properties:

1. $H$ should be continuous in the probability distribution $p_i$.

2. If all the $p_i$ are equal, $p_i = 1/n$, then $H$ should be a monotonic increasing function of $n$. In fact, with equally likely events there is more choice, or uncertainty, when there are more possible events.

3. If a choice is broken down into two successive choices, the original $H$ should be the weighted sum of the individual values of $H$. For example, assume there are three possibilities with $p_1 = \frac{1}{2}$, $p_2 = \frac{1}{3}$, $p_3 = \frac{1}{6}$. The final results have the same probabilities if we first choose between two possibilities each with probability $\frac{1}{2}$ , and if the second possibility occurs we make another choice with probabilities $\frac{2}{3}$, $\frac{1}{3}$. In this special case, we require $H(\frac{1}{2},\frac{1}{3},\frac{1}{6}) = H(\frac{1}{2},\frac{1}{2}) + \frac{1}{2}\,H(\frac{2}{3},\frac{1}{3})$ .

It can be proven that the only $H$ satisfying the three above assumptions is of the form:

$$H(X) = -\sum_X P_X(x) \log P_X(x) \tag{2.3.2}$$

To understand the concrete interpretation of $H(X)$, we suppose $x$ is chosen randomly from the distribution $P_X(x)$ and someone who knows the distribution $P_X(x)$ is asked to guess which $x$ was chosen by asking only yes/no questions. It can be proved that if the guesser uses the optimal question-asking strategy, which is to divide the probability in half on each guess by asking questions like "is $x$ greater than $x_0$ ?", then the average number of yes/no questions it takes to guess $x$ lies between $H(X)$ and $H(X)+1$. This

gives quantitative meaning to "uncertainty": it is the number of yes/no questions it takes to guess a random variable, given knowledge of the underlying distribution and taking the optimal question-asking strategy.

The *conditional entropy* $H(X|Y)$ is the average uncertainty about $X$ after observing a second random variable $Y$, and is given by

$$H(X|Y) = \sum_y P_Y(y)[-\sum_x p_{X|Y}(x|y)\log(p_{X|Y}(x|y))] \tag{2.3.3}$$

where $P_{X|Y}(x|y) = P_{XY}(x,y)/P_Y(y)$ is the conditional probability of $x$ given $y$.

From the definitions of $H(X)$ and $H(X|Y)$, it follows from Eq. 2.3.1 that

$$I(X;Y) = H(X) - H(X|Y) \tag{2.3.4}$$

Mutual information is therefore the reduction in uncertainty about variable $X$, or the expected reduction in the number of yes/no questions needed to guess $X$ after observing $Y$. It is equal to 0 when $X$ and $Y$ are independent; it reduces to the uncertainty associated to $X$ (or equivalently $Y$) if the two variables are connected by a functional relationship.

Mutual information is the expected value of *pointwise mutual information* (PMI), which refers to single events:

$$\mathrm{PMI}(X;Y) = \log\left(\frac{p_{(X,Y)}(x,y)}{p_X(x)\,p_Y(y)}\right) \tag{2.3.5}$$

In the case of words occurrences and co-occurrences, it is clear that if two words are independent, then PMI is 0. If PMI is greater (less) than 0, the two words are more (less) likely to co-occur than if they were independent. When either one of the words (or even both of them) has a low probability of occurrence if singularly considered but its joint probability together with the other word is high, the two are likely to express a unique concept.

## 2.4 Latent Dirichlet allocation (LDA)

Latent Dirichlet allocation (LDA) is a model-based clustering method which was originally proposed in evolutionary biology and bio-medicine to detect the presence of structured genetic variation among a group of individuals. It assumes there are K populations, each of which is characterised by a set of allele frequencies at each locus. By analysing multilocus genotype data, it allows to infer population structure and probabilistically assign individuals to populations [10]. From 2003, LDA was applied to the field of machine learning as a document topic modelling method[3], to discover underlying topics (populations) in a collection of documents (individuals) and infer word probabilities (allele distribution) in topics, thus assigning documents to a topic. It is an unsupervised

learning method, in that the possible topics are not known a priori (only their number is chosen), i.e. they are hidden (latent). Thus, it differs from topic classification, in which the algorithm learns from a dataset that has been previously annotated with topics (supervised learning).

**Mixture representation and exchangeability** LDA models a collection of $D$ documents as topic mixtures $\theta_1, \cdots \theta_D$ over $K$ topics (i.e. probability vectors of length $K$), which are in turn mixtures $\phi_1 \cdots \phi_K$ of $V$ words (i.e. probability vectors of length $V$), $V$ being the number of words in the vocabulary.

The intuition for the LDA model stems from De Finetti representation theorem, for which it is necessary to introduce the concept of *exchangeability*. A finite set of random variables is said to be exchangeable if the joint distribution is invariant to permutation. An infinite sequence of random variables is infinitely exchangeable if every finite subsequence is exchangeable. De Finetti's representation theorem states that the joint distribution of an infinitely exchangeable sequence of random variables is as if a random parameter were drawn from some distribution and then the random variables in question were independent and identically distributed, conditioned on that parameter.
A document is a sequence of $N$ words denoted by $\mathbf{w} = (w_1, w_2, \cdots, w_N)$, where $w_n$ is the $n$th word in the sequence. In LDA, we assume that words are generated by topics (by fixed conditional distributions) and that those topics are infinitely exchangeable within a document. By de Finetti's theorem, the probability of a sequence of words and topics must therefore have the form:

$$p(\mathbf{w}, \mathbf{z}) = \int p(\theta) \left( \prod_{n=1}^{N} p(z_n|\theta) p(w_n|z_n) \right) d\theta \tag{2.4.1}$$

where $\theta$ is the random parameter of a multinomial over topics. Since by this theorem any collection of exchangeable random variables has a representation as a mixture distribution, the word mixture and topic mixture representation can now be understood, provided we assume that we can neglect the order of words ("bag of words" assumption) and the order of topics. While exchangeability is a major simplifying assumption in the domain of text modelling, its principal justification is that it leads to methods that are computationally efficient. Moreover, intuitively, we are able to discern the topics of a document reading its words independently of their order.

**Dirichlet distribution** After understanding the concept of "latent" in the name of LDA, "Dirichlet" derives from the fact that the model assumes that the topic mixtures $\theta_1 \cdots \theta_D$ and the topics $\phi_1 \cdots \phi_K$ follow a prior distribution which is a Dirichlet distribution, with concentration parameters $\alpha$ and $\beta$ respectively. We recall the definition of Dirichlet distribution, choosing the distribution $\theta$ of topics in each document which is

a $K$ dimensional Dirichlet (the same applies to the distribution of words in each topic, which is a $V$ dimensional Dirichlet):

$$p(\theta|\alpha) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \prod_{j=1}^{k} \theta_j^{\alpha_j - 1} \tag{2.4.2}$$

$\{\theta_i\}_{i=1}^{k=K}$ belong to the standard $K-1$ simplex (i.e. the multi-dimensional extension of the triangle), or equivalently $\sum_{i=1}^{K} \theta_i = 1$ and $\theta_i \geq 0$ for all $i \in \{1, \ldots, K\}$, which means that each realisation of a Dirichlet distribution is again a distribution.
The Dirichlet distribution is frequently used in the Bayesian statistics for its important property of being conjugate to the multinomial, meaning that given a multinomial observation, the posterior distribution of a Dirichlet distribution (here $\theta$) is still a Dirichlet distribution. The parameter $\alpha$ controls the mean shape and sparsity of $\theta$. Its effect has an immediate visual interpretation in a 3-dimensional problem, where the simplex is simply a triangle. Small $\alpha_i$ indicate sparsity, in that they make the distribution significantly different from zero only near the $i$th vertex of the triangle. On the contrary, for large $\alpha_i$ the distribution collapses near the centre. In LDA, using a small $\alpha$ means that a document usually contains a small number of topics.

**Generative process** LDA assumes the following generative process whereby documents are generated:

1. For each document $\mathbf{w}$, sample a topic mixture $\theta \sim \text{Dirichlet}(\alpha)$.
2. For each topic, sample $\phi \sim \text{Dirichlet}(\beta)$
3. For each document, for each word position in the document:
   - Sample a topic index $z_n|\theta \sim \text{Categorical}(\theta)$[1], where the random variable $z_n$ is an integer from 1 through $K$.
   - Sample a word $w_n|z_n, \phi \sim \text{Categorical}(\phi_z)$, where the random variable $w_n$ is an integer from 1 through $V$ and it represents the corresponding word in the vocabulary. Since $\phi$ is determined by $\beta$, we will use the following notation $w_n \sim p(w_n|z_n, \beta)$.

Under this generative process, the joint distribution of a document $\mathbf{w}$ (i.e. with words $w_1, \cdots w_n$) with topic mixture $\theta$ and with topic indices $z_1, \cdots z_n$ is given by

$$p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta) = p(\theta|\alpha) \prod_{n=1}^{N} p(z_n|\theta) p(w_n|z_n, \beta) \tag{2.4.3}$$

[1]We recall that the categorical distribution is the particular case of a multinomial distribution for one trial.

where $N$ is the number of words in the document. Summing the joint distribution over $z$ and then integrating over $\theta$ yields the marginal distribution of a document $\mathbf{w}$:

$$p(\mathbf{w}|\alpha,\beta) = \int_{\theta} p(\theta|\alpha) \prod_{n=1}^{N} \sum_{z_n} p(z_n|\theta) p(w_n|z_n,\beta) d\theta \tag{2.4.4}$$

Finally, taking the product of the marginal probabilities of single documents, we obtain the probability of a corpus $D$:

$$p(D|\alpha,\beta) = \prod_{d=1}^{M} \int_{\theta_d} p(\theta_d|\alpha) \left( \prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn},\beta) \right) d\theta_d \tag{2.4.5}$$

There are three levels to the LDA representation. The parameters $\alpha$ and $\beta$ are corpus-level parameters, assumed to be sampled once in the process of generating a corpus. The variables $\theta_d$ are document-level variables, sampled once per document. Finally, the variables $z_{dn}$ and $w_{dn}$ are word-level variables and are sampled once for each word in each document.

**Inference** While identifying the topics in the documents, LDA does the opposite of the generation process by asking what is the hidden structure that likely generated the observed collection. Since the exact evaluation of the posterior distribution is intractable, LDA begins with random assignment of topics to each word and iteratively improves the assignment of topics to words through the so called Gibbs sampling[2]. Gibbs sampling does not explicitly represents $\phi$ or $\theta$ as parameters to be estimated, but instead considers the posterior distribution over the assignments of words to topics, $P(\mathbf{z}|\mathbf{w})$. Then estimates of $\theta$ and $\phi$ can be obtained by examining this posterior distribution.

The steps of the first iteration are:

1. Randomly assign the $K$ topics to all the words in each document
2. Create a document-wise topic count (a local statistic to each document), resulting in a document-topic matrix.
3. Create a topic-wise assignment of word count from all documents (a global statistic for the whole vocabulary), resulting in a topic-word matrix.

After the first iteration, we wish to optimise the initial document-topic and topic-word matrices obtained by iterating over all the documents and all the words.

4. Resample a word and remove the topic assignment (i.e. we assume the current word has been incorrectly assigned while all the others have been correctly assigned).

[2]Gibbs sampling iteratively draws an instance from the distribution of each variable, conditional on the current values of the other variables, with the aim of estimating complex joint distributions.

5. Decrement the count for the respective topic allocated from the document-topic matrix.

6. Decrement the count for the respective topic allocated from the topic-word matrix.

7. The probability for topic assignment $j$ knowing all the other assignments and the observed words is given by [5]:

$$P(z_i = j|\mathbf{z_{-i}}, \mathbf{w}) \propto \frac{n_{-i,j}^{(w_i)} + \beta}{n_{-i,j}^{(.)} + V\beta} \frac{n_{-i,j}^{(d_i)} + \alpha}{n_{-i}^{(d_i)} + K\alpha} \tag{2.4.6}$$

where $n_j^{(d)}$ is the number of times a word from document $d$ has been assigned to topic $j$, $n_j^{(w)}$ is the number of times word $w$ has been assigned to topic $j$ in the vector of assignments $\mathbf{z}$, and the subscript $-i$ indicates that the current assignment $z_i$ is excluded from the count. The first ratio gives the probability of word $i$ under topic $j$, while the second gives the probability of topic $j$ under document $i$. (Here we have assumed for simplicity that the parameters $\alpha$ and $\beta$ have a single value instead of being vector-valued).

8. For a given word $w_i$ in a document $d_i$ find the topic $j$ for which this probability is maximum and reassign the word to topic $j$. In other words, through this product probability, LDA identifies the new topic, which is the most relevant topic for the current word.

LDA (steps 2-8) is performed for a large number of iterations for the step of choosing the new topic $j$ until a steady-state is obtained.

## 2.5 t-distributed stochastic neighbour embedding (t-SNE)

t-SNE is a method of dimensionality reduction which converts a high-dimensional data set $X$ of datapoints $x_1, x_2, ..., x_n$ into a two or three-dimensional set $Y$ of map points $y_1, y_2, ..., y_n$ that can be displayed in a scatterplot. The advantage of t-SNE is that it is capable of capturing much of the local structure of the high-dimensional data very well, while also revealing global structure such as the presence of clusters at several scales. It finds applications in natural language processing to visualise topics of a text corpus and in medicine such as in RNA single-cell sequencing to visualise clusters of cells of the same type. The main idea is to convert distances into probability distributions, for both the high and low dimensional spaces, and find iteratively a low-dimensional data representation that minimises the mismatch between the two distributions.

**Data points joint probability** Firstly, t-SNE converts the high-dimensional Euclidean distances between datapoints into conditional probabilities that represent similarities. The similarity of datapoint $x_j$ to datapoint $x_i$ is defined as the conditional probability $p_{j|i}$ that $x_i$ would pick $x_j$ as its neighbour if neighbours were chosen in proportion to their probability density under a Gaussian centred at $x_i$ [12]:

$$p_{j|i} = \frac{\exp -||x_i - x_j||^2/2\sigma_i^2}{\sum_{k \neq i} \exp -||x_i - x_k||^2/2\sigma_i^2} \tag{2.5.1}$$

where $\sigma_i$ is the variance of the Gaussian that is centred on datapoint $x_i$ and $p_{j|i}$ is normalised over all pairs of points involving $x_i$. We define a symmetrised joint probability of finding datapoints $i$ and $j$ together

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n} \tag{2.5.2}$$

Because the density of the data is likely to vary, in general $\sigma_i$ does not have a global value but it should vary with the datapoint: in dense regions a smaller value of $\sigma_i$ is usually more appropriate than in sparser regions. Any particular value of $\sigma_i$ induces a conditional probability distribution over all the other datapoints given datapoint $x_i$, $P_i$, whose entropy increases with $\sigma_i$. To understand how $\sigma_i$ is determined, we introduce the notion of perplexity.

**Perplexity** The perplexity of a probability distribution $p(x)$ is given by

$$Perp(p) = 2^{H(p)} \tag{2.5.3}$$

where $H(p)$ is the Shannon entropy of $p$ measured in bits. It is independent of the basis provided the basis of the exponentiation and of the logarithm is the same, as it can be seen by rewriting the definition as

$$Perp(p) = 2^{-\sum_x p(x) \log_2 p(x)} = \frac{1}{\prod_x p(x)^{p(x)}} \tag{2.5.4}$$

From this, if we consider an $N$-sided fair dice, the entropy is maximum and the perplexity is $N$

$$\frac{1}{\left(\frac{1}{N}^{\frac{1}{N}}\right)^N} = N \tag{2.5.5}$$

Thus, perplexity can be interpreted as the number of sides of a fair die that when rolled produces a sequence with the same entropy as the given probability distribution.
t-SNE sets $\sigma_i$ in such a way that $P_i$ has a fixed perplexity $Perp(P_i)$ that is specified by the user. Perplexity provides a smooth measure of the number of effective neighbours, thus this translates into scaling the variance of the gaussian so that a fixed number of points fall in the mode of the gaussian, allowing the algorithm to adapt to the different densities in space.

**Map points joint probability** If we defined the joint probabilities in the low-dimensional space, $q_{ij}$, using the Gaussian distribution as we did for $p_{ij}$, we would encounter the so-called "crowding problem", which arises because all the pairwise distances cannot be preserved when embedding a high dimensional points into a lower dimension. When trying to model local structure (neighbours) faithfully, dissimilar high dimensional datapoints have to be mapped too far apart in the map. This hinders the segregation of map points that should represent datapoints neighbours from moderately distant points, with the consequence that gaps between natural clusters are lost. Mapping dissimilar high dimensional datapoints too far apart in the map is instead allowed if we use a heavy tailed distribution - compared to the gaussian - in the low dimensional space. In fact, if two high dimensional points have a distance $||x_i - x_j||_X = d^*$ and $p_{ij} = p^*$, in order to have the same probability density $q_{ij} = p^*$ the map points have to be more distant $||y_i - y_j||_Y > d^*$. For this reason, t-SNE employs Student's t-distribution with one degree of freedom in the low-dimensional map, which is a heavy-tailed distribution compared to the gaussian, so that the joint probabilities are defined as

$$q_{ij} = \frac{(1 + ||y_i - y_j||^2)^{-1}}{\sum_k \sum_{l \neq k} (1 + ||y_k - y_l||^2)^{-1}} \tag{2.5.6}$$

normalised by all pairs of points. This distribution is used because large clusters of points that are far apart interact in just the same way as individual points, due to the fact that the numerator approaches an inverse square law for large $||y_i - y_j||$.

**Cost function** A natural measure of the faithfulness with which a probability distribution $Q$ models a reference probability distribution $P$ is the *Kullback-Leibler divergence*. The cost function $C$ is chosen to be a single Kullback-Leibler divergence between a joint probability distribution, $P$, in the high-dimensional space and a joint probability distribution, $Q$, in the low-dimensional space, which is defined as

$$C = KL(P||Q) = \sum_i \sum_{j \neq i} p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{2.5.7}$$

KL divergence is asymmetric (therefore it is not a distance); this reflects the asymmetry in Bayesian inference, which starts from a prior $P$ and updates to the posterior $Q$. The code theory intuition behind this fact is that if we are transmitting information that is distributed according to $Q$, then the optimal (lossless) compression will need to send on average $H(Q)$ bits. In case we expect $Q$ (and design compression accordingly), but the actual distribution is $P$, we will send on average $H(P) + KL(P||Q)$ bits; in short, $KL(P||Q)$ is the "penalty" for using wrong distribution. The asymmetry of the KL divergence allows t-SNE to preserve local structure close, large $p_{ij}$, then if they have small $q_{ij}$ there is a high penalty. Thus similar datapoints are modelled by similar map

points.
The minimisation of the cost function is performed using gradient descent, where the gradient is

$$\frac{\partial C}{\partial y_i} = 4\sum_j (p_{ij} - q_{ij})(y_i - y_j)(1 + ||y_i - y_j||^2)^{-1} \tag{2.5.8}$$

Physically, the gradient may be interpreted as the resultant force created by a set of springs between the map point $y_i$ and all other map points $y_j$. All springs exert a force along the direction $(y_i - y_j)$. The spring between $y_i$ and $y_j$ repels or attracts the map points depending on whether the distance between the two in the map is too small or too large to represent the similarities between the two high-dimensional datapoints. The force exerted by the spring between $y_i$ and $y_j$ is proportional both to its length and to its stiffness, which is the mismatch $(p_{ij} - q_{ij})$ between the joint probabilities of the data points and the map points. We note that because $q$ is modelled by a Student distribution, the t-SNE gradient strongly repels dissimilar datapoints that are modelled by a small pairwise distance in the low-dimensional representation.
The gradient descent has a momentum term to ensure faster convergence and reduce oscillation.

## 2.6 Linear regression

Linear regression is a simple widely used tool of which more advanced statistical learning approaches can be seen as generalisations or extensions. It describes a linear relationship between a set of $d$ *predictors* or *explanatory variables* $\mathbf{x}_k$, $k = 1, \cdots, d$ and a *response* variable $\mathbf{y}$, having made $n$ observations of $\mathbf{x}_k$ and $\mathbf{y}$. It allows to study the fraction of variability of $\mathbf{y}$ explained by $\mathbf{x}_k$ and to predict values of $\mathbf{y}$ for new values of $\mathbf{x}_k$. Let

$$\mathbf{X} = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1d} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nd} \end{pmatrix}$$

be the $n \times (d+1)$ design matrix, $\mathbf{y} = (y_1, \cdots, y_n)^T$ the $n \times 1$ response, and $\boldsymbol{\beta} = (\beta_0, \cdots, \beta_d)^T$ the $(d+1) \times 1$ unknown model parameters. Let $\boldsymbol{\epsilon} = (\epsilon_1 \cdots, \epsilon_n)^T$ be the random *error* term, which represents what the model cannot describe (non-linearity, the existence of other predictors that are not taken into account). Then

$$\mathbf{y} = f(\mathbf{X}) + \boldsymbol{\epsilon} = \mathbf{X} \cdot \boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{2.6.1}$$

defines the *population regression* line, which is the best linear approximation to the true relationship between $\mathbf{X}$ and $\mathbf{y}$.
We assume that $\boldsymbol{\epsilon}$ has mean 0, thus

$$E(\mathbf{y}) = \mathbf{X}\boldsymbol{\beta} \tag{2.6.2}$$

Moreover, we assume that the errors $\epsilon_i$ are distributed with the same variance $\sigma^2$ (a condition known as homoscedasticity or homogeneity of variances) and uncorrelated with each other. In a compact form, this is written as

$$cov(\boldsymbol{\epsilon}) = \sigma^2 \mathbf{I}_n \tag{2.6.3}$$

These assumptions are the hypotheses of the Gauss-Markov theorem, which guarantees the validity of ordinary least squares for estimating regression coefficients (see below). If the data suggests homoscedasticity does not hold, a weighted linear regression is more appropriate. We usually also assume $\epsilon_i$ to be normally distributed: $\epsilon_i \sim N(0, \sigma)$. While this is not required for the computation of point estimates of the parameters, it allows hypothesis testing and calculation of confidence and prediction intervals.
We define the $i$th *residual* as the difference between residual the $i$th observed response value and the $i$th response value that is predicted by our linear model[3]:

$$e_i = y_i - \hat{y}_i \tag{2.6.4}$$

We use our training data to produce estimates for the parameters, which we denote with $\hat{\boldsymbol{\beta}}$, through the least squares approach, which minimises the residual sum of squares (RSS)

$$RSS = \sum_{k=1}^{n} e_k^2 \tag{2.6.5}$$

The least squares regression coefficient estimates $\hat{\boldsymbol{\beta}}$ allow us to predict the value of the response, thus characterising the *least squares line*

$$\hat{\mathbf{y}} = \mathbf{X} \cdot \hat{\boldsymbol{\beta}} \tag{2.6.6}$$

More explicitly, RSS can be written

$$J(\boldsymbol{\beta}) = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{2.6.7}$$

and direct calculation shows that the condition of minimisation

$$\frac{\partial J(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = 0$$

is satisfied by

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \tag{2.6.8}$$

[3]We note that the residuals express the departure of the observed values from the predicted ones, while errors express the departure of the observed values from the real unknown ones. Therefore, the error term is unobserved, contrary to the residuals which are its estimation, and assumptions on the error term are tested on residuals.

From Eq. 2.6.2 it follows directly that $\hat{\boldsymbol{\beta}}$ is an unbiased estimator of $\boldsymbol{\beta}$:

$$E(\hat{\boldsymbol{\beta}}) = \hat{\boldsymbol{\beta}} \tag{2.6.9}$$

This means that by repeating the regression on $N$ samples extracted from the same population, the average of $\hat{\beta}_i$ (least square line) tends to $\beta_i$ (population line). We see that based on the concept of bias one can make an apt analogy between linear regression and estimation of the mean of a random variable, in that the sample mean is an unbiased estimator of the population mean. As in the case of the population mean, the error we associate to a linear regression parameter estimation is the standard error, quantifying how far away this estimation is on average from the true $\beta_i$. Using Eq. 2.6.3 we find

$$cov(\hat{\boldsymbol{\beta}}) = \sigma^2(\mathbf{X}^T\mathbf{X})^{-1} \tag{2.6.10}$$

where an unbiased estimator for $\sigma^2$ is

$$\hat{\sigma}^2 = \mathbf{y}^\mathbf{T}[\mathbf{I}_n - \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T]\mathbf{y}/(n-d) \tag{2.6.11}$$

We now briefly state Gauss-Markov theorem, for which the previous assumptions on the errors are needed (see [9] for a proof). In many problems it is of interest to estimate linear combinations of predictors $\boldsymbol{\beta}$, say, $\mathbf{t}^T\boldsymbol{\beta}$, where $\mathbf{t}$ is any nonzero $d \times 1$ vector of known constants. We define the best linear unbiased estimator of $\mathbf{t}^T\beta$:

**Definition 2.6.1** (Best Linear Unbiased Estimator (BLUE) of $\mathbf{t}^T\boldsymbol{\beta}$)**.** The best linear unbiased estimator of $\mathbf{t}^T\boldsymbol{\beta}$ is

i a linear function of the observed vector $\mathbf{y}$, $\mathbf{a}^T\mathbf{y} + a_0$ where a is an $n \times 1$ vector of constants and $a_0$ a scalar, and

ii the unbiased estimator of $\mathbf{t}^T\boldsymbol{\beta}$ with the smallest variance.

**Theorem 2.6.1** (Gauss-Markov)**.** *Let* $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ *where* $E(\boldsymbol{\epsilon}) = \mathbf{0}$ *and* $cov(\boldsymbol{\epsilon}) = \sigma^2\mathbf{I}_n$*. Then the least-squares estimator of* $\mathbf{t}^T$ *is given by* $\mathbf{t}^T\hat{\boldsymbol{\beta}} = \mathbf{t}^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ *and* $\mathbf{t}^T\hat{\boldsymbol{\beta}}$ *is the BLUE of* $\mathbf{t}^T\boldsymbol{\beta}$*.*

Finally, we mention the evaluation metrics we will use on the validation set to compare models with different predictors. These are mean absolute error (MAE)

$$\text{MAE} = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j| \tag{2.6.12}$$

and root mean square error (RMSE) [4]

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2} \tag{2.6.14}$$

which are both measured with the same scale as $Y$.

[4] If $n$ indicates the total number of observations (without division into training and validation set) the unbiased estimator of the variance of $\epsilon$ is the residual standard error (RSE)

$$\text{RSE} = \sqrt{\frac{1}{n-d-1}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2} \tag{2.6.13}$$

as $d$ parameters have been estimated from the data. However, when evaluating our model on the validation set we use RMSE, as by definition the parameters have been estimated from the training set and not the validation set.

# Chapter 3

# Text Processing and Analysis

## 3.1 Descriptive statistics

**Bag of words** We recall that an $n$-gram is a contiguous sequence of $n$ items from a given sample of text or speech; in our case, items are tokens of one or two words. We built bags of words in the unigram and bigram models and visualised them as wordclouds. We note that we aggregated in a single token "male gola|testa|pancia" [1], as well as the most frequent expressions to negate symptoms: "niente|nessuno|neanche|non|senza|nemmeno|nè|né" + "male gola|febbre|raffreddore|tosse" [2], by means of regular expressions. More generally, "non" ("not") was aggregated with the following word. This was decided because otherwise the word "non" alone would be so prevalent compared to the rest of the vocabulary that statistics would be disturbed.

We note that since all tweets contained mentions of fever, cough or sore throat, due to the keywords used in the Twitter API, the unigram does not enable us to observe significant differences between true symptoms (Fig. 3.1.1a) and fake symptoms (Fig. 3.1.1b). The prevalent words - apart from symptoms and cold - were COVID test, positive, negative, mask, school, as well as time indicators (now, yesterday, day...) and words indicating anxiety.

As for the bag of bigrams, the most frequent one among symptoms tweets (Fig. 3.1.2a) was "take COVID test", followed by the pairs of symptoms (including cold and headache), a symptom with an additional temporal detail (38/39 C fever, today sore throat, start coughing, still sore throat) or an adjective (mild/strong sore throat, dry cough, high fever...), feeling unwell or, to a lesser extent, feeling better. There are mentions of staying home, going/not going to school. From the manual selection we recall that "staying home" was often linked to surprise of how the symptoms could have been caught considering the user had been staying at home, or to staying home/ not going to school for precaution once

[1] "sore throat/headache/stomachache"

[2] These are variations of "no|not even|neither|nor" + "sore throat|fever|cold| cough"


(a) (b)

Figure 3.1.1: Unigram wordcloud for true symptoms (a) and fake symptoms (b).


(a) (b)

Figure 3.1.2: Bigram wordcloud for true symptoms (a) and fake symptoms (b).

the symptoms had been observed. Mention of school were at times related to discovering schoolmates were experiencing symptoms and hoping not to be infected, or to pondering whether or not to go to school with mild symptoms, which could however be dangerous considering the pandemic. Finally, there are mentions of calling the doctor/taking the temperature/ taking a medicine for temperature/ negative COVID test /not wearing masks. Compared to true symptoms tweets, the most prevalent bigram for fake symptoms tweets was "coughing fit" (166), followed by combination of symptoms and "take COVID test" (102), as it can be seen from (Fig. 3.1.2b). They contain some temporal references outside the range of 3 days from the publication date: "last week", "10 day", "this year",

"last year", "year ago". We observe "without mask" (32) and "do vaccine" (31), the latter mostly related to mentions of symptoms when doing past vaccines such as the flu. To a lesser extent there were words indicating something going down the wrong pipe and mention of public figures that were discussing COVID.

**Co-occurrence matrix** Since the bag of words stores vocabulary tokens and their counts $C$, the co-occurrence matrix $M$, whose entries are the number of times that two words appear in the same tweet, is simply computed as $M = C^T C$.
We calculated pointwise mutual information to observe if a certain (unordered) pair of words was more or less frequent compared to what would be expected for independent words. For this purpose, only words co-occurring more than 20 times were considered. The pairs with higher PMI can be found in Tab. 3.1; reported values are above 3 bits, in decreasing order.

| true symptoms | fake symptoms |
|---|---|
| olfatto + gusto | sapore + odore |
| naso + colare | de + girolamo |
| non riuscire + respirare | perdita + olfatto |
| chiuso + naso | traverso + saliva |
| positivo + contatto | gusto + olfatto |
| medico + chiamare | perdita + gusto |
| ora + mezz | male testa + macron |
| prendere + tachipirina | positivo + sabato |
| senza + mascherina | bene + macron |
| sentire + odore | male testa + video |
| risultato + positivo | positivo + giannini |
| | positivo + risultare |
| | andare + traverso |
| | stare + macron |

Table 3.1: Pairs of tokens with highest pointwise mutual information for true and fake symptoms. Reported values are above 3 bits and in decreasing order.

We observe that fake symptoms comprised retweets of (the same) online news articles mentioning famous people that were experiencing symptoms (Macron, De Girolamo, Giannini...). This explains the high PMI of pairs of tokens which individually are not the most frequent ones.

The relative position of chosen pairs of words related to symptoms is shown in Fig.3.1.3, which shows that the absolute distance of these words in tweets is mostly 1-2 words (thus allowing their appearance in bigrams).

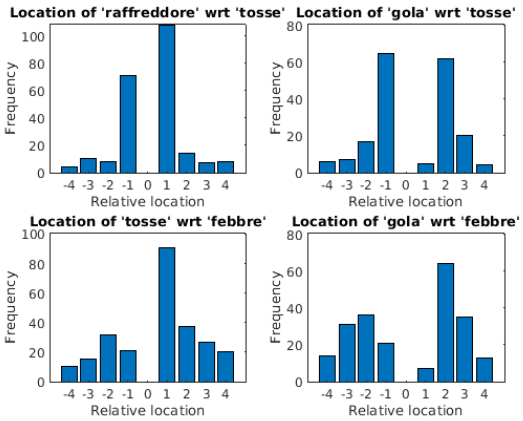



Figure 3.1.3: Relative position of chosen pairs of words related to symptoms

Moreover, we created an unordered graph from the co-occurrence matrix, where each node represented a word and edges were weighted by the co-occurrence of the two words linked, as shown in Fig. 3.1.4 for true symptoms and Fig. 3.1.5 for fake symptoms. Diagonal entries have been set to 0 to avoid self-loops, which would simply represent the total occurrences of a given word. For this purpose we used Gephi [1], an open-source software for network visualisation and analysis. The network has a small diameter of only 4, as expected because all the selected tweets contained mentions of fever, sore throat or cough. For better readability, we filtered out nodes with a degree less than 15 and edges with a weight less than 10 for symptoms and 12 for fake symptoms (the nodes with a degree of only 1 or 2 were often misspelled words.) The spacial distribution is the result of the original Yifan Hu's attraction-repulsion algorithm followed by additional node repulsions, especially at the centre, to avoid label overlapping. The statistics used for colour, label size and node size (modularity, closeness centrality and degree, respectively), were computed before filtering the network.

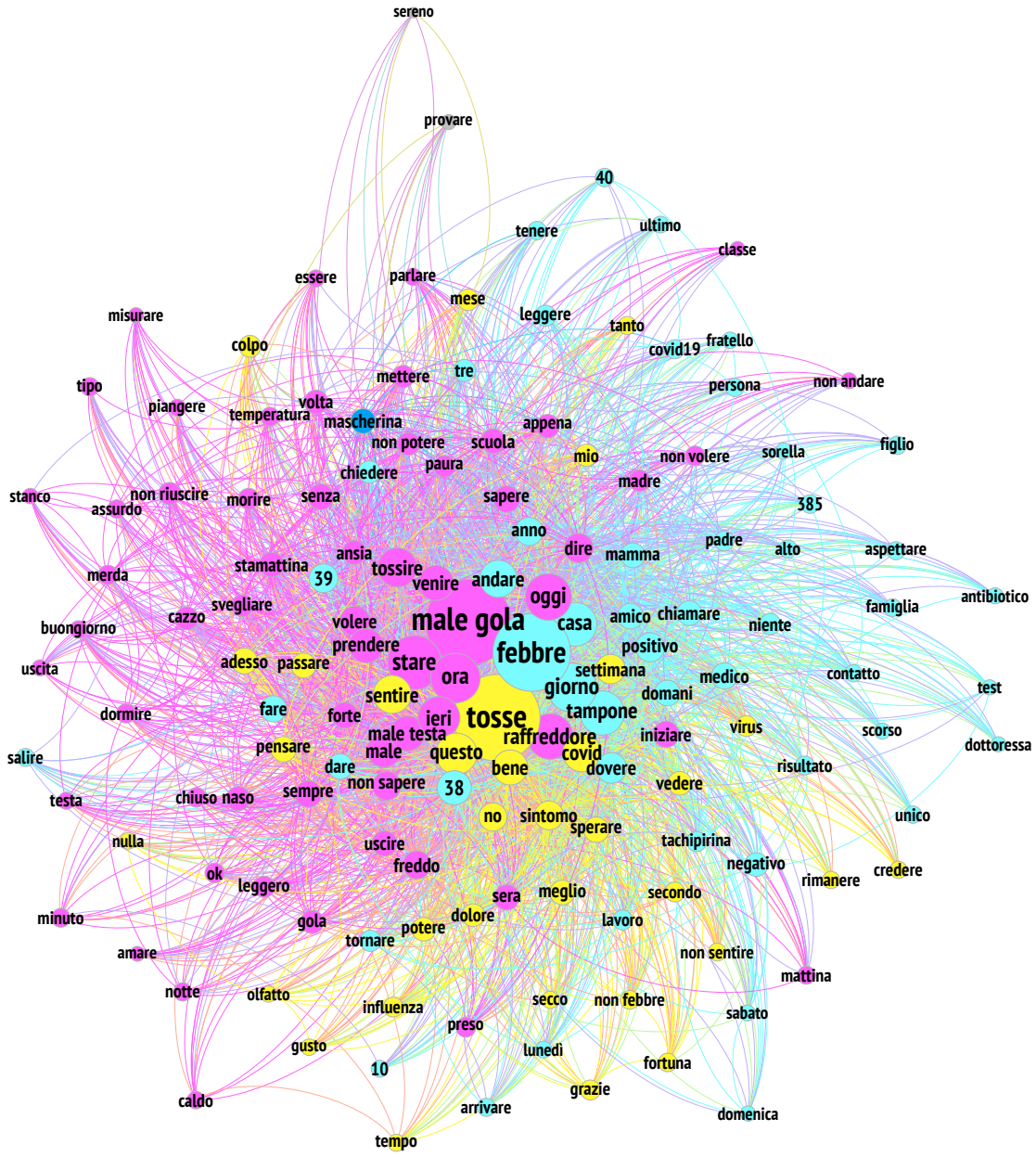



Figure 3.1.4: Co-occurrence network for true symptoms; only nodes with degree above 15 and edges with weight above 10 are shown. Colour represents modularity class, node size represents degree, label size represents closeness centrality.

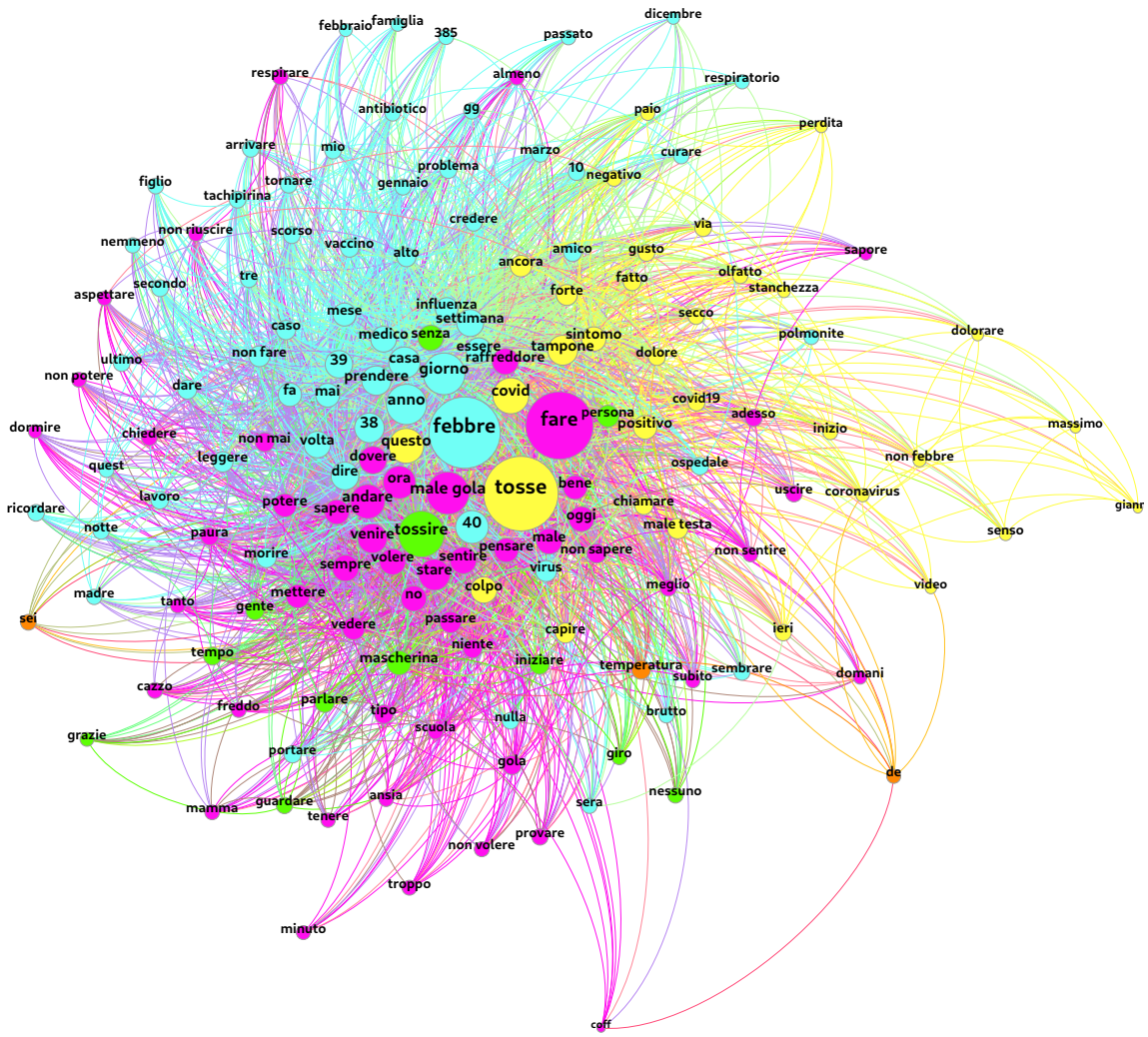



Figure 3.1.5: Co-occurrence network for fake symptoms; only nodes with degree above 15 and edges with weight above 12 are shown. Colour represents modularity class, node size represents degree, label size represents closeness centrality.

## 3.2 Topic modelling

We searched for groups of words (topics) in all tweets using LDA [2]. We wished to investigate whether this allowed to highlight differences between (already labelled) true symptoms and fake symptoms.

Since the topics are learned by the model, the topics (topic words probabilities) will generally differ when LDA is applied to different data (i.e. true vs fake symptoms). Therefore, applying LDA separately to true and fake symptoms would lead to results of difficult interpretation. For this reason, we chose to perform it on all tweets together and subsequently to investigate whether there were observable differences in the collective documents topic distribution of true and fake symptoms.

Since LDA requires the number of topics as input, we performed a parametric search and chose the number of topics that minimised the validation perplexity, finding an optimal value of 16 topics. We then applied t-SNE, which is needed to embed the document topic probabilities in a lower dimension. In this way we visualised clusters of tweets in a 2D scatter plot, as shown in Fig. 3.2.1. Each point represents a tweet, which has been assigned to the topic with highest probability within the tweet.

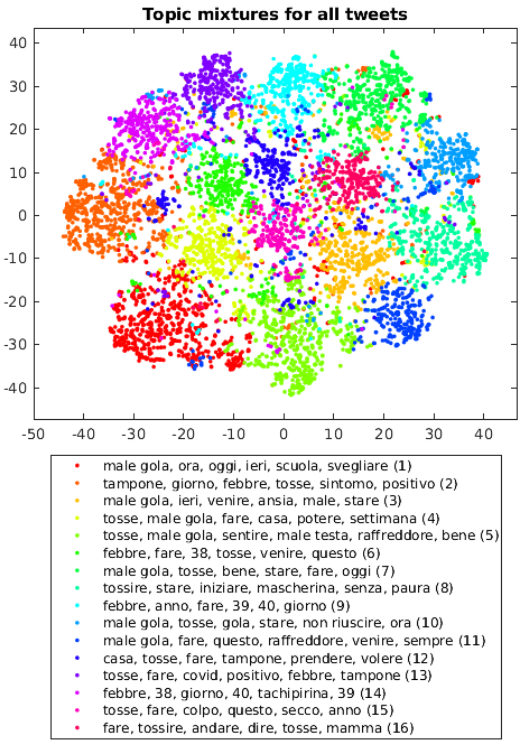



Figure 3.2.1: Topics mixture for all tweets visualised via t-SNE. The optimal number of topics was found to be 16. The perplexity has been set to 50.

We recall that since topics are learned by the algorithm as a mixture of words that occur together, they are not labelled a priori but their interpretation is left to human eye. For our data, being that all tweets contained words related to symptoms, it is clear how the clustering is soft (a word can belong to several topics). In our plots, topics are labelled with their most frequent words; this is only for a compact visualisation, as it is clear that the most common words are repeated and one should refer to the wordcloud or co-occurrence network for each topic instead.

Finally, we computed the mean and standard error of the document topic matrix, for each topic (column), weighting on the number of tokens in each document. This was done separately for true and fake symptoms, as shown in Fig. 3.2.2. Keeping in mind that the outputs of LDA are probability distributions, we found the main discrepancy between true and fake symptoms to be in the probability assigned to topic 1, which is the topic with highest corpus probability. Interestingly, this contained the words ”anxiety”, ”panic”, ”terrible”, ”bad”, ”worse”, ”kill”, ”die”, as well as other words expressing frustration, and was more frequently assigned to symptoms tweets.

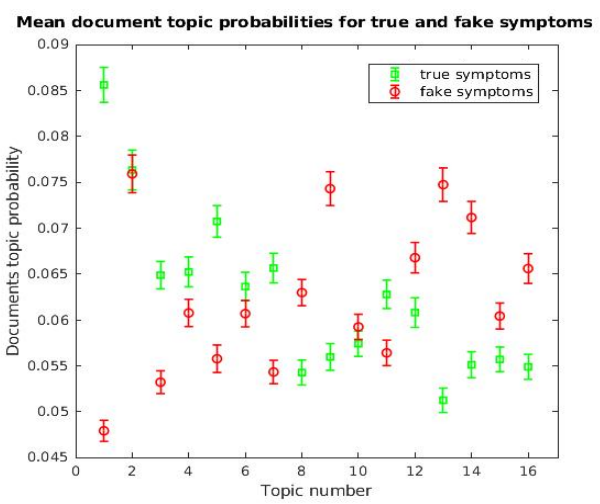



Figure 3.2.2: Mean document topics probabilities for true and fake symptoms, weighted with the number of words in each document. The error bar represents the standard error; for the topics number label refer to Fig. 3.2.1.

# Chapter 4

# Trends of tweets volume and prediction of COVID-19 hospitalisations

## 4.1 Trends of tweets volume

**Building time series** Our database is composed of 7618 public tweets, acquired from September 30th 2020 to January 26th 2021 (inclusive) for a total of 119 days. The daily tweets distribution is shown in Fig. 4.1.1. The manual classification led to 4164 out of the total 7618 tweets classified as symptoms.

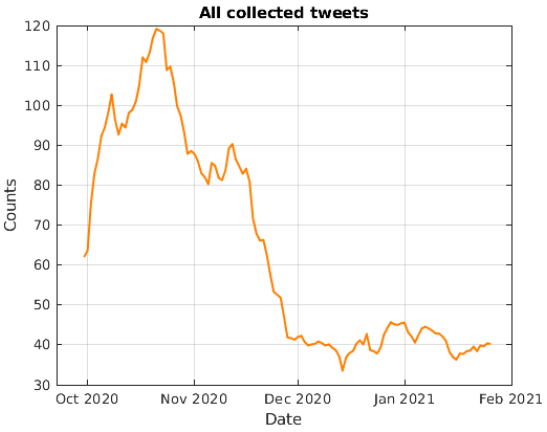



Figure 4.1.1: Daily distribution of all collected tweets.

We remark that in order to select individual symptoms, we kept from the table of symptoms tweets those that did not contain negations, i.e. ”niente|nessuno|neanche|non|senza|nemmeno|né|nè” + ”febbre|tosse|male gola”, as well as ”passato” + ”febbre|tosse|male gola” and ”febbre|tosse|male gola” + ”passato”. This simple filter to intercept the negation of symptoms is needed to deal with users that ponder what to do after realising they have a certain symptom but not another one. This allowed to filter out 21 negated mentions of sore throat, 34 of cough and 113 of fever, leading to 2042 true mentions of sore throat, 1494 of cough and 982 of fever.

This allowed us to build daily time series of each symptom (Fig. 4.1.2) and each possible

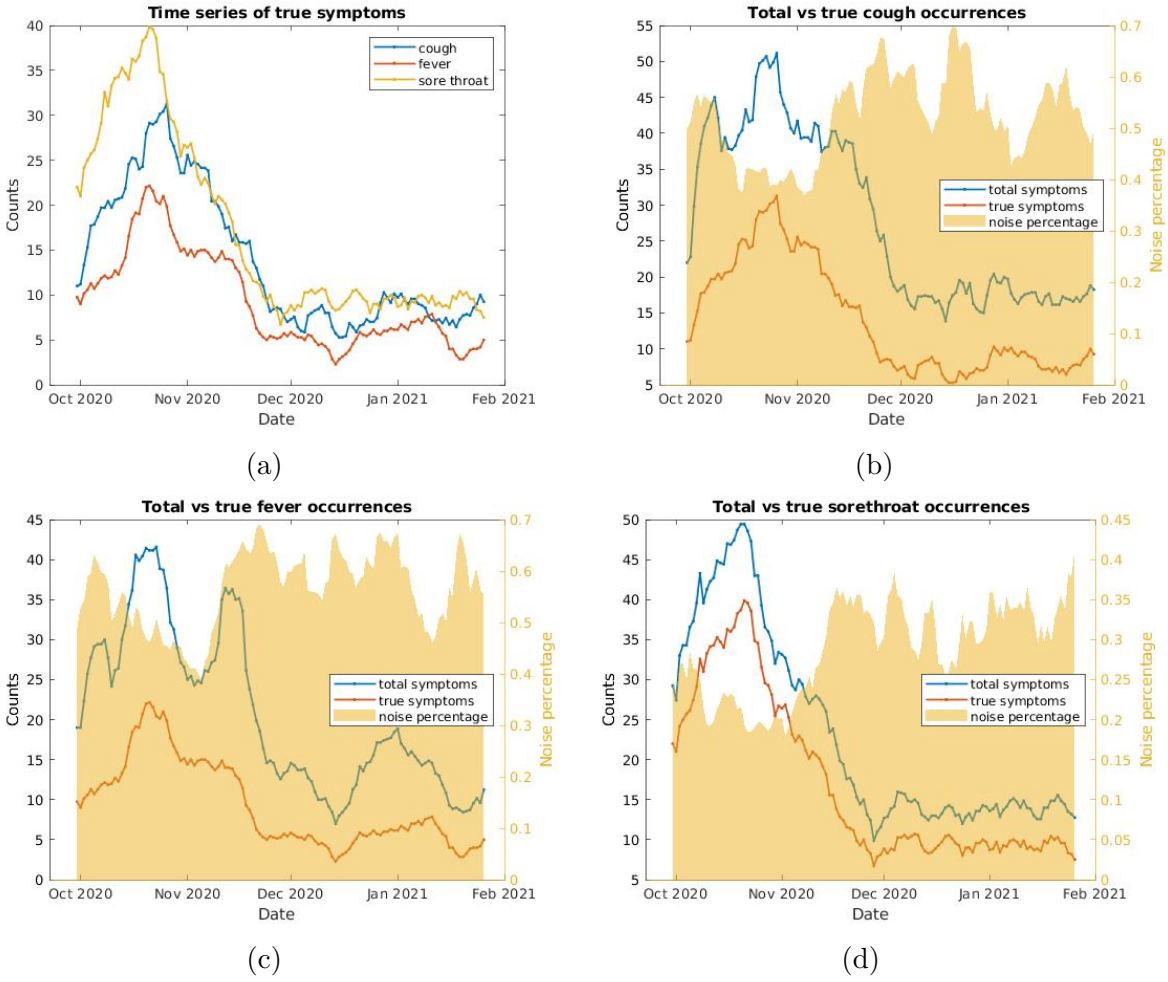
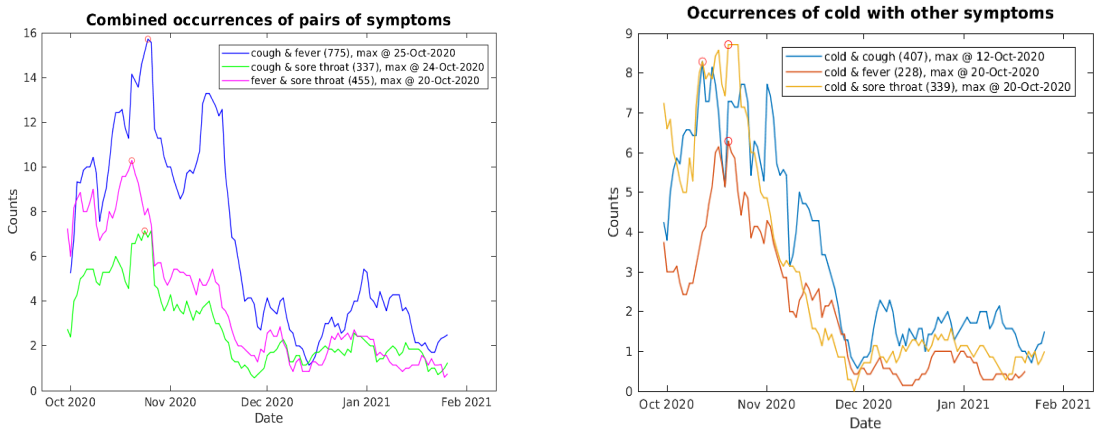



Figure 4.1.2: True symptoms (a). Total vs true occurrences for each symptom: cough ($\rho = 0.953$) (b), fever ($\rho = 0.939$) (c) and sore throat ($\rho = 0.997$)(d).

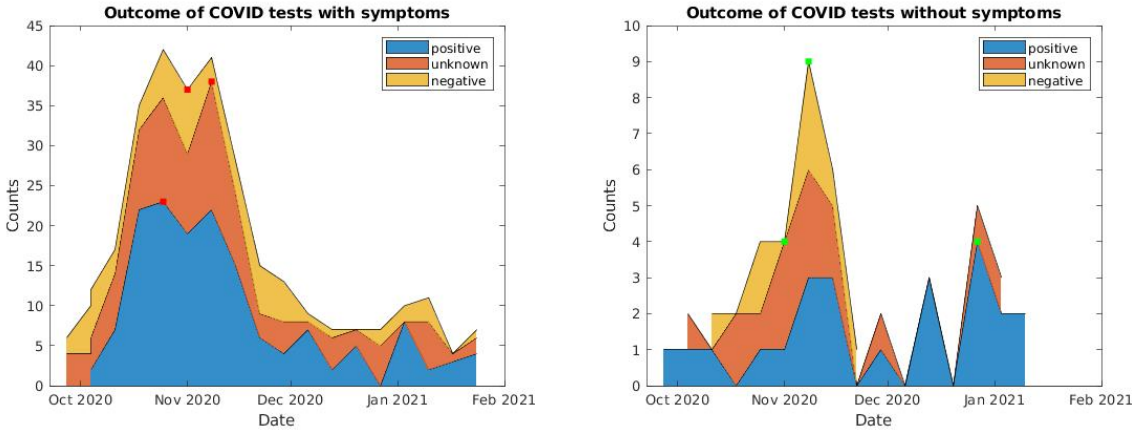
pair of symptoms (Fig. 4.1.3, here we include cold). A moving mean with a window of 7 days was applied to all daily series that follow. We observe how the ratio of a true symptom to total mentions of the symptom fluctuates over time, with the correlation between true and total symptoms being $\rho = 0.953$ for cough, $\rho = 0.939$ for fever and $\rho = 0.997$ for sore throat. The noise exhibits a decrease during the first and major peak, as well as - to a lesser extent - during the second and minor peak, for fever and cough tweets.

We built weekly time series of self-reported COVID tests, which are reported as area plots, divided according to whether symptoms were present (Fig. 4.1.4a) or not (Fig. 4.1.4b). We found that, out of tests mentioning absence of symptoms, the outcome was negative 8 times, unknown 15 and positive 23, while out of tests mentioning symptoms




(a) (b)

Figure 4.1.3: Combined occurrences of pairs of total symptoms tweets (a) and of total cold with other total symptoms tweets (b). Total co-occurrences are shown in brackets. The markers indicate the maximum for each series.




(a) (b)

Figure 4.1.4: Weekly aggregated outcome of COVID tests, divided into positive, unknown or negative outcome. The maximum is shown in red for tests with symptoms (a) and in green for tests without symptoms (b).

there were 56 negatives, 101 unknown and 151 positives. For these time series the maximum occurs near the beginning of November, which corresponds to a first and major peak. The exception is positive outcomes of tests without symptoms, whose maximum occurs during the second peak around the end of December. We remark that the two categories (outcome of the test, presence of symptoms) were labelled separately,

meaning that a minority of the tweets contains mentions of presence/absence of symptoms experienced by one person, and at the same time mention of a different person taking a COVID test. For example, a recently seen relative has just taken a COVID test and the user reports not having symptoms at the moment. As expected, the main contribution to the tweets mentioning taking a test with symptoms reported a positive outcome. The presence of two peaks in the time series of tests with symptoms may be due to the fact that the time period can also extend in the future from the publication date (people mentioning having booked a test within the following 3 days), in contrast with the symptoms time series.

**Comparing time series** We chose to focus on the following time series: all collected tweets, true symptoms and individual symptoms. We compared them with the data of new hospitalisations in Italy provided by INFN (Istituto Nazionale di Fisica Nucleare) [8], which has a scientific collaboration agreement with ISS (Istituto Superiore Sanità). INFN assures that these data are not affected by delays in communication between ASL (Azienda Sanitaria Locale) and regions but are referred to the actual dates. We chose new hospitalisations as a reference instead of official positive tests as the latter depends on the number of total tests, which is not constant and is thus less reliable.

All series are shown z-standardised in Fig. 4.1.5. As it can be seen from it, the series show a first marked peak, in which the delay of new hospitalisations with respect to symptoms tweets is clear, and a second minor peak, for which it is reduced. While the main peak corresponds to the second COVID wave, the beginning of the third wave might be already suggested by the increase in true fever and true cough tweets in the last days of our acquisition, after 20th January and 18th January respectively.

To compare the time series of tweets with the new hospitalisations series, we computed Pearson's linear correlation coefficient $\rho$ with lag, since it is reasonable to expect a delay between the experience of symptoms and the increase in new hospitalisations. For each possible lag $l$, the subseries of overlapping points was z-standardised and the quantity

$$\rho = E[x_{n+l} y_n] = \sum_{n=0}^{N-l-1} (x_{n+l} y_n)/(N-l) \tag{4.1.1}$$

was computed, where $N$ is the length of the time series. Maximising the correlation coefficient allowed to find the optimal values ($l^*$, $\rho^*$). The results are shown in Fig. 4.1.6 and Tab. 4.1.

We observe that sore throat tweets are the series with the highest correlation (0.97) but also the highest lag (22 days). On the contrary, tests with symptoms are the series with the smallest lag compared to new hospitalisations (9 days): they present a double peak, whose second part - centred at the beginning of November - is overlapped with the peak of new hospitalisations. It is clear that tests are booked after the experience of symptoms (by the user or someone else), thus with a smaller lag compared to new

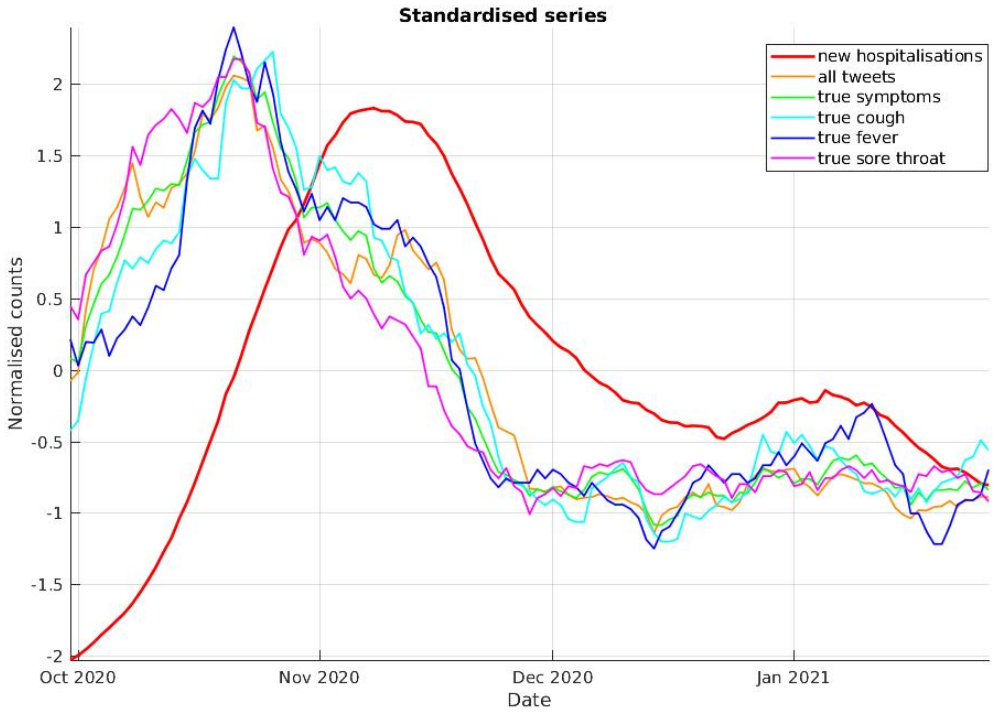



Figure 4.1.5: z-standardised time series of new hospitalisations, all tweets, true symptoms and individual true symptoms.

hospitalisations. However, for further analyses - namely the monthly correlation with new hospitalisations (see below) and the regression model (Sec. 4.2) - we focused only on the series of symptoms. This is because they provided a larger statistics compared to the series of tests with symptoms and also because self-reported tests had not been acquired independently of the symptoms tweet, but were a subset, labelled a posteriori.

The decrease in the delay between the first and the second peak is more clearly visible in Fig.4.1.7, where the series have been shifted by the optimal lag, also showing that - especially for fever - the steepness of descent in the first peak varies with time compared to that of new hospitalisations. Since this suggests non linearity, we calculated the moving correlation coefficient (Fig. 4.1.8a) and the lag (Fig. 4.1.8b) of new hospitalisations with respect to individual symptoms, with a window of length 30 days. We constrained the lag to values $l = 0, 1, ...N - 5$ in order to keep at least 5 points for the computation of $\rho$. This was needed to avoid lags that would discard the majority of data and keep only 2-3 points for each series, causing a misleading $\rho \sim 1$. We observe that for windows starting after day 65 $\rho$ drops, as for lower tweets volume the signal to noise ratio decreases. This effect is dominant for sore throat, for which there is no sign of a second peak. On the other

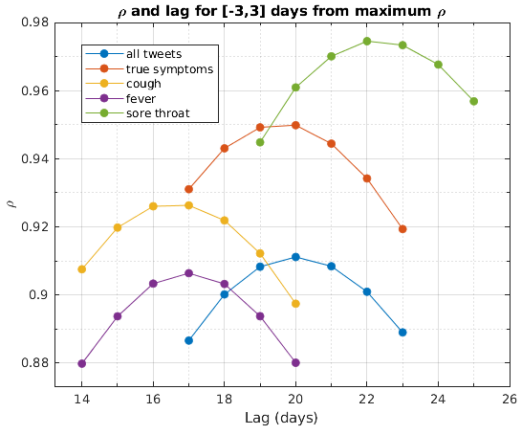



Figure 4.1.6: (a)

| | $\rho^*$ | $l^*$ (days) |
|---|---|---|
| all tweets | 0.91 | 20 |
| true symptoms | 0.95 | 20 |
| true cough | 0.93 | 17 |
| true fever | 0.91 | 17 |
| true sore throat | 0.97 | 22 |

Table 4.1: (b)

Comparison of our time series with new hospitalisations via linear correlation coefficient $\rho$ with lag. Fig. (a) shows the lag and $\rho$ for $[-3, 3]$ days from maximum $\rho$; the values are written in Tab. (b).

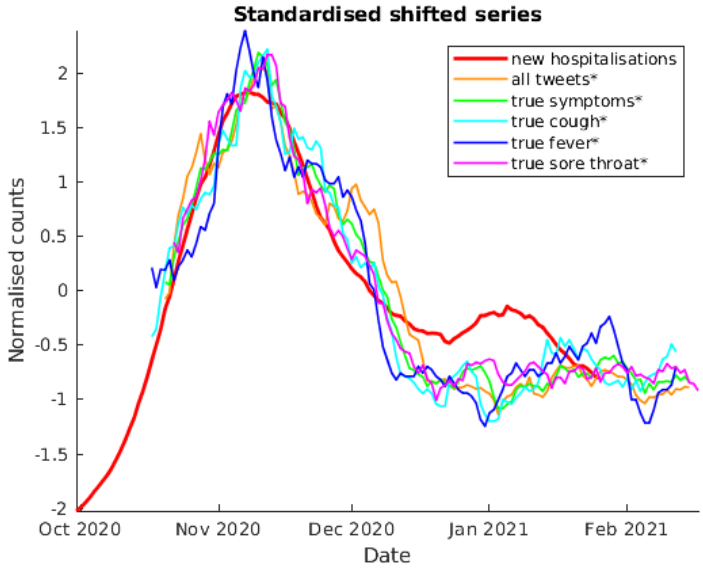
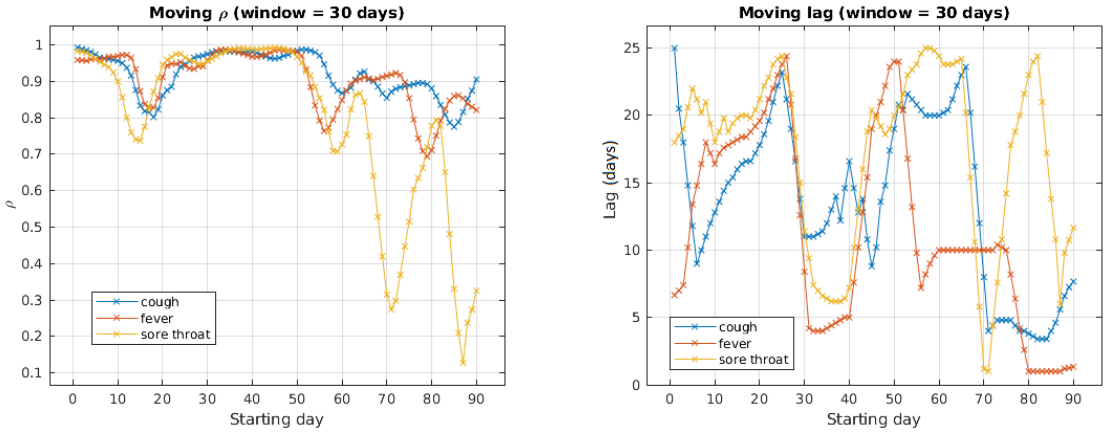



Figure 4.1.7: z-standardised shifted time series of new hospitalisations, all tweets, true symptoms and individual true symptoms. The symbol * indicates that they have been shifted by the optimal lag. The time scale is that of new hospitalisations.

hand, sore throat has the most stable lag for windows starting up to day 25, i.e. windows that span the first and major peak of symptoms. While this analysis provided insight




(a) (b)

Figure 4.1.8: Moving linear correlation coefficient (a) and lag (b) of new hospitalisations with respect to symptoms, with a window of length 30 days.

into the dynamics of the correlation between the series, it is clear that with lags higher than 20 days only less than one third of monthly data is considered in the computation.

Asking ourselves if there could be a closer relationship (in terms of lag) between tweets and news articles about COVID, a Google search was performed to count the number of COVID-related articles published in the news section in Italian during the selected period. Using Python modules "googlesearch" and "requests", we retrieved the URLs and titles of the web-pages written in Italian which contained at least one of the following words in the title: "covid", "coronavirus", "tamponi" (i.e. Covid tests), "positivi" (i.e. tested positive) or "lockdown". The search was performed in the news section and in the same period of time spanned by our time series. As shown in Fig. 4.1.9, from this we observed that a peak formed in the same period as our peak. However, a trough was present - around the end of December - which did not match with our tweets series or new hospitalisations series. We believe the collected data were affected by noise due to a number of blocked re-

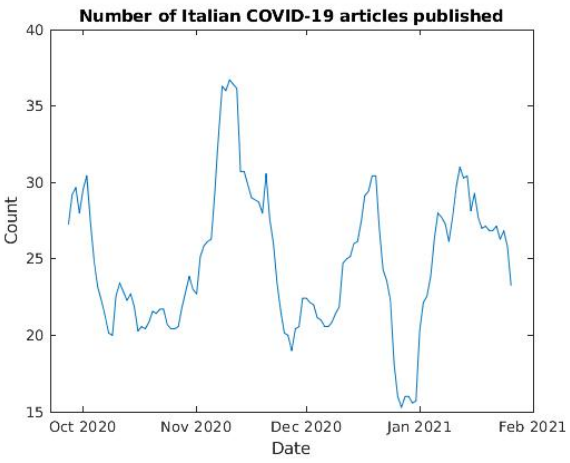



Figure 4.1.9: Daily number of Italian articles regarding COVID retrieved in the news section using "googlesearch"(a).

quests when trying to connect to the web-pages. For a better analysis, paid plans of APIs for Google scraping would be necessary.

## 4.2 Linear regression model

**Univariate regression** We wished to build a univariate linear regression model for new hospitalisations. We used cough, sore throat and fever series (individually) lagged of $n$ days as predictor variable. Since the predicted values of new hospitalisations at time $t$ depend on the individual symptoms tweets at time $t-n$, no predictions were made for the first $n$ days. Using R[6], we first performed a simple partition of the dataframe in training and validation set, assigning the first 100 points to the training set and the remaining 19 to the validation set (from January 8th to January 26th 2021). This did not allow a meaningful prediction of the second peak due to the different temporal dynamics of the two series. In fact, we recall that the lag between new hospitalisations and symptoms significantly decreases from the first and major peak to the second and minor peak. Adding the corresponding symptom series lagged of only a couple of days as predictor in each model did not improve the fit, as the values of the coefficient are still determined by the first and major peak, which thus gives a higher weight to the higher lag.

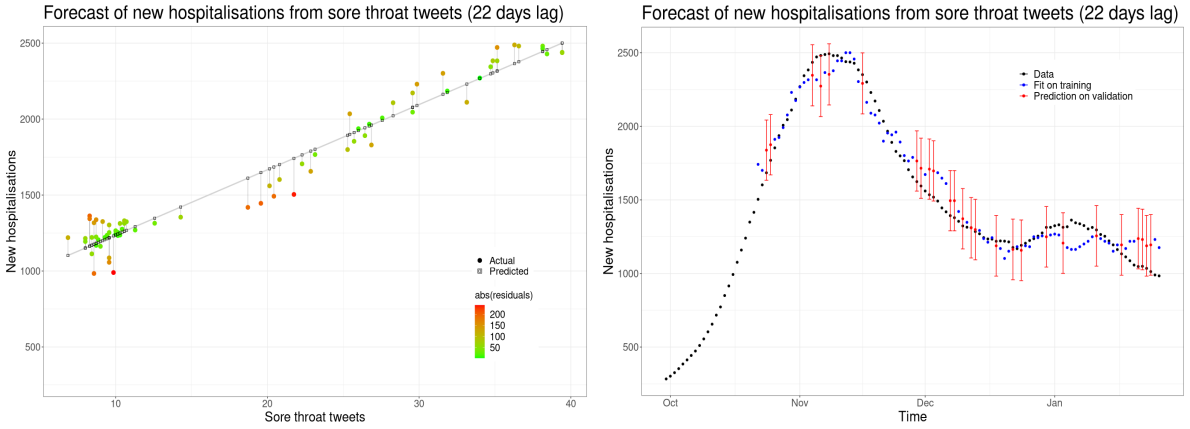
Thus, acknowledging that a linear model is not suitable to predict a non stationary distribution, we decided to perform a random partition, assigning 75% of the data to the training set. In this way, the model is trained with observations scattered around the whole temporal extension, which constitutes our distribution as a whole. We chose as optimal lag $n$ the value that minimised the root mean square error (RMSE) and the mean absolute error (MAE) on the validation set. The results for univariate regression are reported in Tab. 4.2 for true and total symptoms; the line has equation $\hat{y} = \hat{\theta}_0 + \hat{\theta}_1 x_k$, $\hat{y}$ being the predicted values for new hospitalisations and $x_k$ being one of the symptoms.

| | $l^*$(days) | $\theta_0$ | $\theta_1$ | RMSE | MAE |
|---|---|---|---|---|---|
| true sore throat | 22 | $808 \pm 25$ | $42.9 \pm 1.1$ | 121 | 104 |
| true cough | 15 | $810 \pm 46$ | $58.5 \pm 2.9$ | 178 | 146 |
| true fever | 17 | $796 \pm 50$ | $91.3 \pm 4.9$ | 205 | 161 |
| total sore throat | 22 | $716 \pm 31$ | $36.0 \pm 1.1$ | 138 | 124 |
| total cough | 15 | $593 \pm 80$ | $34.3 \pm 2.5$ | 237 | 191 |
| total fever | 14 | $748 \pm 94$ | $37.7 \pm 3.8$ | 271 | 204 |

Table 4.2: Results of univariate regression on symptoms, where the line has equation $\hat{y} = \hat{\theta}_0 + \hat{\theta}_1 x_k$. Parameters are reported with their standard errors; p-values are all under $10^{-9}$.

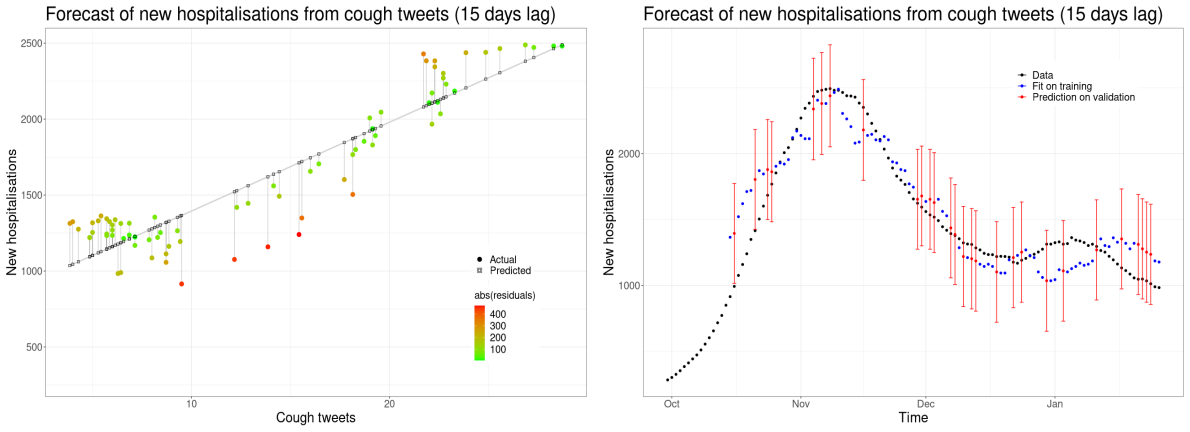
We notice a decrease in RMSE of models using true symptoms as predictors compared

to total symptoms, namely 12% for sore throat, 25% for cough and 24% for fever. This suggests the usefulness of manual filtering, particularly for cough and fever tweets.



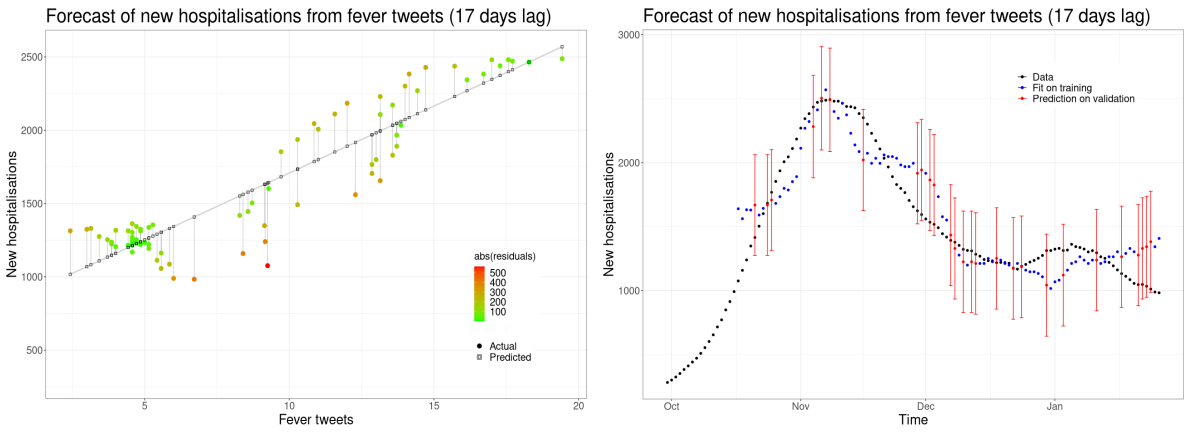

(a) (b)

Figure 4.2.1: Univariate regression of new hospitalisations ($y$) from 22 days lagged sore throat tweets ($x_s$). Figure (a) shows the predicted values (squares), which lie on the regression line with equation $\hat{y} = (808 \pm 25) + (42.9 \pm 1.1)x_s$. Circles represent the actual values of new hospitalisations; they are coloured according to the absolute value of the residuals, which is represented by the length the vertical segment. Figure (b) shows data for new hospitalisations (black), fit on training set (blue) and prediction on validation set (red) with 95% prediction intervals.




(a) (b)

Figure 4.2.2: Univariate regression of new hospitalisations ($y$) from 15 days lagged cough tweets ($x_c$). The regression line has equation $\hat{y} = (810 \pm 46) + (58.5 \pm 2.9)x_c$.




(a) (b)

Figure 4.2.3: Univariate regression of new hospitalisations ($y$) from 17 days lagged fever tweets ($x_f$). The regression line has equation $\hat{y} = (796 \pm 50) + (91.3 \pm 4.9)x_f$.

We plotted the regression line, to which points $\hat{y_j}$ belong, and the corresponding points $y_j$, joined by vertical segments (the residuals), as it can be seen from Fig. 4.2.1a (sore throat), Fig. 4.2.2a (cough) and Fig. 4.2.3a (fever). The partition of observations in training and validation sets is shown in Fig. 4.2.1b, Fig. 4.2.2b and Fig. 4.2.3b respectively.
In order to assess homoscedasticity - one of the key assumptions in linear regression - we performed the studentised Breusch-Pagan test, whose null hypothesis is that homoscedasticity is verified. We were not able to reject it, having obtained a p-value of 0.45 for sore throat, 0.68 for cough and 0.58 for fever.

**Multivariate regression** Finally, we performed multivariate linear regression with sore throat ($x_s$), cough ($x_c$) and fever ($x_f$) tweets as predictors for new hospitalisations, using the previously found optimal lags for each symptom (see Fig. 4.2.4). We found MAE$= 103$ and RMSE$= 120$; the equation of the regression line is

$$\hat{y} = (809 \pm 26) + (42.2 \pm 4.8)x_s + (2.6 \pm 7.4)x_c + (-2.4 \pm 9.9)x_f \tag{4.2.1}$$

Again, homoscedasticity cannot be rejected (p-value=0.27). Interestingly, we observe that the weight of sore throat tweets is dominant, with cough and fever tweets coefficients being almost 20 times smaller and affected by a larger relative error.

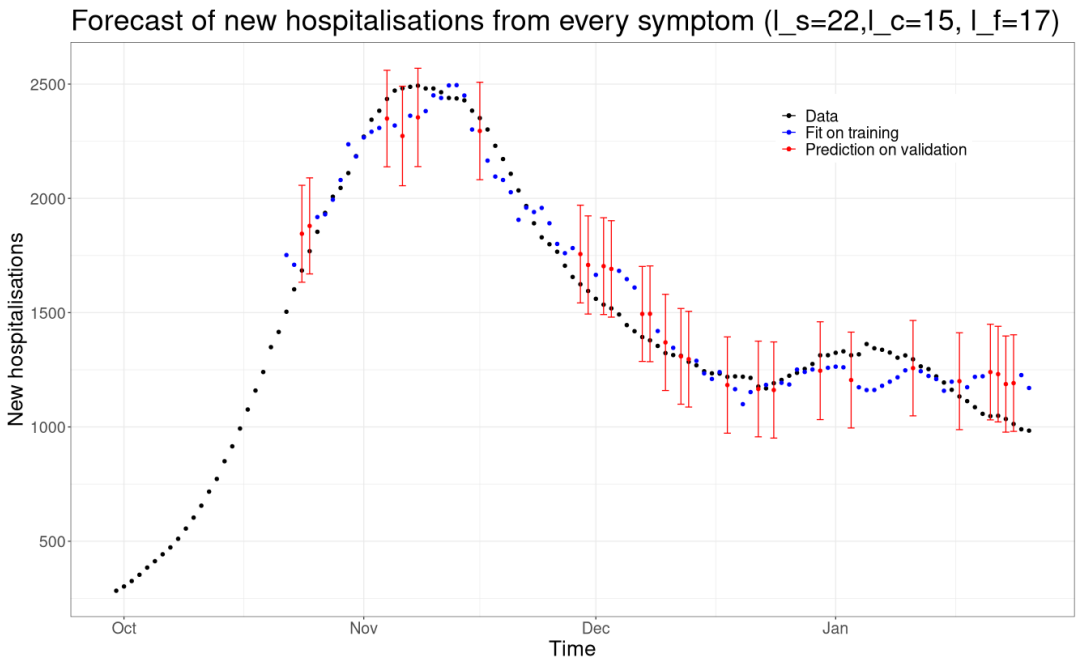



Figure 4.2.4: Multivariate regression of new hospitalisations ($y$) from 22 days lagged sore throat tweets ($x_s$), 15 days lagged cough tweets ($x_c$) and 17 days lagged fever tweets ($x_f$). The regression line has equation $\hat{y} = (809 \pm 26) + (42.2 \pm 4.8)x_s + (2.6 \pm 7.4)x_c + (-2.4 \pm 9.9)x_f$.

# Conclusions

After manual selection of tweets actually describing self-reported COVID symptoms, by comparing the series of each true symptom with its total symptom, we found that sore throat tweets have the highest correlation ($\rho = 0.997$), followed by cough tweets ($\rho = 0.953$) and finally fever tweets ($\rho = 0.939$). Total fever tweets show a more complicated structure in the first peak and a more pronounced second peak compared to true fever tweets. Firstly, we recall that during manual selection we discarded tweets mentioning a light fever (under $38°C$). Secondly, it happened that fever was mentioned hypothetically or sarcastically, such as when users expected to catch fever after realising they had sore throat, or mentioned feeling as if they had a high temperature. In addition, it is true that assessing fever requires a measurement, contrary to sore throat and cough, which could lead to a difference between feeling fever and objectively experiencing it. The above correlations suggest that if one wishes to build a model without manual selection of tweets, the series of sore throat tweets would be more reliable than cough and fever tweets, as we have later verified. In fact, we found a decrease in RMSE of models using true symptoms as predictors compared to total symptoms, namely 12% for sore throat, 25% for cough and 24% for fever, which supports the importance of manual filtering, especially for cough and fever tweets.

An evident feature of our time series is the diminishing in time of the lag between symptoms and new hospitalisations, from the first major peak around November - corresponding to the second COVID wave - to the second minor peak around January - located between the second and third COVID wave (the third wave is not part of our study). The second peak may not be well resolved due to noise: not only statistical fluctuations, but also the overlap of what we labelled as true symptoms with symptoms of diseases other than COVID. In fact, one should also investigate the time evolution of diseases with the same symptoms as COVID, mostly flu but also bronchitis or streptococcus. On the other hand, it is well known that obligation of mask wearing and restrictions on gatherings reduced the incidence of flu compared to previous years. Moreover, the higher number of hospitalisations in the first peak could have caused a longer waiting time from the appearance of symptoms to new hospital admissions, compared to January when the pressure on hospitals was lower, with new hospitalisations being half of those in the first peak. More extensive investigation would be needed to test these hypotheses, possibly

with a wider time scale to determine if the variation is only local or not. These could start at least 20 days earlier to capture the start of the first peak in tweets and extend until at least April to capture the third wave, whose beginning might be suggested by the last days of our data.

Finally, through univariate linear regression with random train-validation split we found that the optimal delay to predict new hospitalisations from self-reported symptoms is 22 days for sore throat, 15 for cough and 17 for fever. Having found that multivariate regression is dominated by true sore throat tweets, whose coefficient is higher than true cough and fever tweets by a factor 20, we report as final result the regression line with sore throat tweets as predictor: $\widehat{y} = (808 \pm 25) + (42.9 \pm 1.1)x_s$. By considering its RMSE, we conclude our model has a resolution of $\pm 120$ counts in predicting new hospitalisations. We stress that the analysis is limited in time due to the non stationarity of the relationship between predictors and the dependent variable. In fact, while on one hand linear regression suggests sore throat is the best predictor for new hospitalisations - in agreement with the fact that it has the highest global correlation with new hospitalisations ($\rho = 0.97$) - on the other hand it is the symptom with the lowest correlation in the final part of our data, since it does not exhibit a second peak, contrary to new hospitalisations, fever and cough tweets.

We stress that a straightforward linear regression model on COVID symptoms tweets allowed us to predict new hospitalisations 22 days in advance and with high correlation on our dataset extending from September 30th 2021 to January 26th 2021. This suggests the practical usefulness of constantly monitoring social networks posts and possibly performing manual annotation on them, in order to update such models and investigate variations of the lag.

# Acknowledgements

I am grateful to Professor Remondini and PhD student Durazzi for their time, responsivity and close guidance. They stimulated my interest in natural language processing, suggested new questions to investigate from our data and kept the focus on achieving a coherent work. I would also like to thank my parents and friends for their renewed encouragement and inspiration, as well as their interest in this thesis. Last but not least, I thank my grandmother for continuously supporting my university studies.

# Bibliography


[1] The Open Graph Viz Platform. URL https://gephi.org/. 21

[2] Text Analytics Toolbox. URL https://it.mathworks.com/help/textanalytics/index.html?s_tid=CRUX_lftnav. 24

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003. 7

[4] F. Durazzi, M. Müller, and M. e. a. Salathé. Clusters of science and health related twitter users become more isolated during the covid-19 pandemic. *Sci Rep*, 11(19655), 2021. doi: 10.1038/s41598-021-99301-0. 1

[5] T. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 101 Suppl 1:5228–35, 04 2004. doi: 10.1073/pnas.0307752101. 11

[6] R. Hyndman and G. Athanasopoulos. *Forecasting: principles and practice (3rd edition)*. OTexts: Melbourne, Australia, 2021. URL OTexts.com/fpp3. 33

[7] S. W. H. Kwok, S. K. Vadde, and G. Wang. Tweet Topics and Sentiments Relating to COVID-19 Vaccination Among Australian Twitter Users: Machine Learning Analysis. *J Med Internet Res*, 23(5):e26953, May 2021. ISSN 1438-8871. doi: 10.2196/26953. URL https://doi.org/10.2196/26953. 2

[8] L. Lista. Dati dell'istituto superiore di sanità. URL https://covid19.infn.it/iss/. 29

[9] B. K. Moser. 5 - Least-Squares Regression. In B. K. Moser, editor, *Linear Models*, Probability and Mathematical Statistics, pages 81–103. Academic Press, San Diego, 1996. doi: https://doi.org/10.1016/B978-012508465-9/50005-3. URL https://www.sciencedirect.com/science/article/pii/B9780125084659500053. 16

[10] J. K. Pritchard, M. Stephens, and P. Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945–959, 2000. ISSN 0016-6731. URL https://www.genetics.org/content/155/2/945. 7

[11] C. E. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(3):379–423, 1948. doi: https://doi.org/10.1002/j.1538-7305.1948.tb01338.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/j.1538-7305.1948.tb01338.x. 6

[12] L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 12