Alma Mater Studiorum · Università di Bologna

Scuola di Scienze
Corso di Laurea Magistrale in Fisica del Sistema Terra

Deterministic and probabilistic verification of multi-model meteorological forecasts on the subseasonal timescale

Relatore:
Prof. Silvana Di Sabatino

Correlatori:
Dott. Piero Malguzzi
Dott. Daniele Mastrangelo

Presentata da:
Alfonso Ferrone

Sessione I
Anno Accademico 2015/2016

Sommario

In questo studio, un multi-model ensemble è stato implementato e verificato, seguendo una delle priorità di ricerca del Subseasonal to Seasonal Prediction Project (S2S). Una regressione lineare è stata applicata ad un insieme di previsioni di ensemble su date passate, prodotte dai centri di previsione mensile del CNR-ISAC e ECMWF-IFS. Ognuna di queste contiene un membro di controllo e quattro elementi perturbati. Le variabili scelte per l'analisi sono l'altezza geopotenziale a 500 hPa, la temperatura a 850 hPa e la temperatura a 2 metri, la griglia spaziale ha risoluzione $1° \times 1°$lat-lon e sono stati utilizzati gli inverni dal 1990 al 2010. Le rianalisi di ERA-Interim sono utilizzate sia per realizzare la regressione, sia nella validazione dei risultati, mediante stimatori nonprobabilistici come lo scarto quadratico medio (RMSE) e la correlazione delle anomalie.

Successivamente, tecniche di *Model Output Statistics* (MOS) e *Direct Model Output* (DMO) sono applicate al multi-model ensemble per ottenere previsioni probabilistiche per la media settimanale delle anomalie di temperatura a 2 metri. I metodi MOS utilizzati sono la regressione logistica e la regressione Gaussiana non-omogenea, mentre quelli DMO sono il *democratic voting* e il *Tukey plotting position*. Queste tecniche sono applicate anche ai singoli modelli in modo da effettuare confronti basati su stimatori probabilistici, come il *ranked probability skill score*, il *discrete ranked probability skill score* e il *reliability diagram*. Entrambe le tipologie di stimatori mostrano come il multi-model abbia migliori performance rispetto ai singoli modelli. Inoltre, i valori più alti di stimatori probabilistici sono ottenuti usando una regressione logistica sulla sola media di ensemble. Applicando la regressione a dataset di dimensione ridotta, abbiamo realizzato una curva di apprendimento che mostra come un aumento del numero di date nella fase di addestramento non produrrebbe ulteriori miglioramenti.

Abstract

In this study, a multi-model ensemble is implemented and verified pursuing one of the research priorities of the Subseasonal to Seasonal Prediction Project (S2S). The re-forecasts from the CNR-ISAC and the ECMWF IFS monthly prediction systems, each including a control run and four perturbed members, are linearly combined and regressed against the ERA-Interim reanalyses for the winter season. The regression technique is applied on two meter and 850 hPa temperature, and geopotential height at 500 hPa on a 1°× 1°lat-lon grid, for the period ranging from 1990 to 2010. ERA-Interim reanalyses are also used to verify the results through non-probabilistic scores, namely root mean square error (RMSE) and anomaly correlation.

Model output statistics (MOS) techniques and direct model output (DMO) are subsequently applied to the multi-model ensemble to obtain forecast probabilities of weekly averaged 2-meter temperature anomalies. The MOS methods tested are logistic regression and non-homogeneous Gaussian regression, the DMO ones are democratic voting and the Tukey plotting position. The same techniques are employed on the two models separately for comparison purposes based on probabilistic scores, such as ranked probability skill score, discrete ranked probability skill score and reliability diagram. Both probabilistic and non-probabilistic verification results show that the multi-model forecasts outperform the single-model counterparts. Moreover, the method that produces the highest skill scores is logistic regression, when the ensemble mean was used as the sole predictor. By applying the same technique to reduced datasets, we computed a learning curve, which demonstrates that extending the number of training dates will not likely lead to further improvements.

Contents

Chapter 1

The subseasonal timescale

The predictability on the sub-seasonal timescale is, nowadays, an active and challenging field of study. Forecasts over this time range represent a fundamental tool for their impact on management decisions in agriculture and food security, water, disaster risk reduction and health. An in-depth analysis of all the practical applications of this products, together with an evaluation of the possible social and economical advantages, can be found in the S2S Research Implementation Plan [2013].

Historically, the scientific community focused mostly on the medium-range forecasts and the seasonsal ones, while the sub-seasonal time range received considerably less attention, being often considered a "predictability desert". However, recent studies (such as van den Hurk et al. [2012], Sobolowski et al. [2010], Lin and Wu [2011], Baldwin et al. [2003], Woolnough et al. [2007], Fu et al. [2007]) and books (like Lau and Waliser [2011]) suggested the existence of some important sources of predictabilityin the monthly timescale, too. In addition to the improvements in the model development and the availability of more powerful computing resources, these factors led to a growing interest toward the subject, which resulted also in the implementation of a Subseasonal to Seasonal Prediction (S2S) Project by the World Meteorological Organization (WMO), started in 2013[1].

1.1 Objective of the thesis

The focus of this thesis is on a particular case of the subseasonal time range: we analyze the products of two monthly forecasting systems (the ECMWF-IFS and the CNR-ISAC ones) trying to improve, through some statistical techniques, their performances in predicting some probabilistic and non-probabilistic quantities. The whole analysis can be split in two main parts.

In the first one, we implement a multi-model combination of the ECMWF-IFS and the CNR-ISAC ensemble forecasts through linear regression. Then, we evaluate the resulting fields against the ERA-Interim reanalysis using non-probabilistic scores. This procedure ideally aims to extract from the two fore-

[1] Additional information on the background of the S2S Project, together with its objectives and research priorities can be found on the official website:
http://s2sprediction.net/static/about#objectives

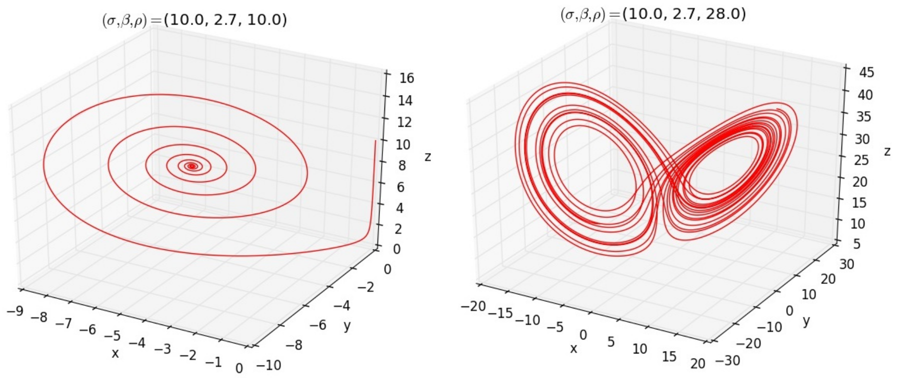
casts all the available information, assigning to the two models the optimal weight for predicting the desired anomalies. The idea behind it is that some complementarity exists between the two prediction systems, in order to produce an output field based (locally) on the most skillful model, and contemporarily filtering out some noise not correlated to the observed anomalies.

The second part focus on the realization and verification of probabilistic forecasts. In summary, we extract the terciles of the temperature at 2 meter from the ERA-Interim reanalysis and compute the probability that the predicted temperature falls above or below each of these two quantiles. For the task, we apply both "direct model output" techniques and regression methods. The latter require a training phase and are more computationally demanding, therefore we expect some improvements in the probabilistic skills in order to justify the added complexity. All the methods are tested on both the multi-model and the two single models, in order to check if improvements can be seen also in the probabilistic scores. In the best-case scenario, different sources of skill from the ECMWF-IFS and the CNR-ISAC ensembles will be combined in a unique product, from which the Model Output Statistics (MOS) will produce more skillful forecasts than the one obtainable from the single models.

Before diving into the statistical description of the various methods, in this chapter we summarize the major sources of predictability on the timescale considered: these are the physical reasons why long-range forecasts are possible in the first place. In order to provide a more complete exposition, we do not focus only on the features used in our analysis, but we give a brief overview of the phenomena commonly studied in the scientific literature and regarded as important factors for enhancing forecast skill on the subseasonal scale. So, in the following sections, we describe both the impact of modelling certain components of the Earth system and the effect of some circulation patterns over this particular window of predictability. Of all the elements in the list, some are indeed present in the prediction systems from which our analysis begins: for example the ECWF-IFS model contains an ocean-atmosphere coupling that can affect the forecast skills, as described later. On the other hand, we did neither checked the presence of the discussed circulation patterns, nor evaluated their impact of the forecast skill. This is partially due to the lack of some fundamental variables from the fields analyzed, but also to the large amount of time required for such computation. We considered more important to use this time in testing different algorithms, trying to identify the one best suited for our task. However, in an hypothetical extension of this study, it would be interesting to include such analysis.

1.2 Atmospheric predictability

The underlying assumption behind our analysis is that some atmospheric, land or oceanic process act as a source of predictability over this time range. Without such sources, even the best statistical techniques are useless: we cannot extract information from the data if there is none. In this section, we briefly analyze why a predictability limit exists in the first place, and then we focus on the specific reasons why, over the subseasonal timescale, some skill remain even after the two-weeks range, commonly considered the limit for skillful weather forecast at the mid-latitudes.



(a) Stable

(b) Unstable

Figure 1.1: Two integration of the Lorenz system for different choices of parameters and initial conditions. The equations are:

$$\begin{cases} \frac{dx}{dt} = \sigma(y - x) \\ \frac{dy}{dt} = \rho x - y - xz \\ \frac{dz}{dt} = xy - \beta z \end{cases}$$

We used a 4$^{\text{th}}$ order Runge Kutta method for the numerical integration of the system. In **(a)** the initial condition chosen are $(x_0, y_0, z_0) = (-0.01, -0.01, 10)$, slightly closer to the negative equilibrium point, while in **(b)** we chose randomly the initial condition $(x_0, y_0, z_0) = (-7.4, -8.7, 14.4)$. The values of the parameters are shown at the top of the two panels.

1.2.1 Predictability limit

Weather forecasts for short or medium ranges is normally regarded as a problem depending mainly on the atmospheric initial conditions [Bjerknes, 1904]. Normally, the numerical models used in predicting future states of the atmosphere contain some set of differential equation, with the exact nature of the set and the approximation used depending on an extremely high number of factors. Due to the non-linearity in these equations, it is possible for two initial conditions very similar to result in totally different forecast, if the numerical integration covers a sufficiently large time range [Lorenz, 1969]. This behavior makes us consider the atmosphere an unstable system. Note that this existence of a finite limit of predictability is an intrinsic property of some non-linear sets of differential equations, as shown in Lorenz [1963].

In order to illustrate the concept, he used a simplified 3-variable model, characterized by some nonlinearities in the set of equations and time-indipendent coefficients (autonomous system)[2]. The most interesting feature of this system is the existence, for some choice of coefficients, of a chaotic behavior that, in the end, leads to unpredictability.

Naturally, this does not happen always. For example, we show in Figure 1.1 the result of two different integration of the system, performed changing one of the parameters (ρ). Without focusing on the details, the trajectory in **(a)** is

[2] A complete explanation of the Lorenz System can be found in [Kalnay, 2003, Chapter 6].

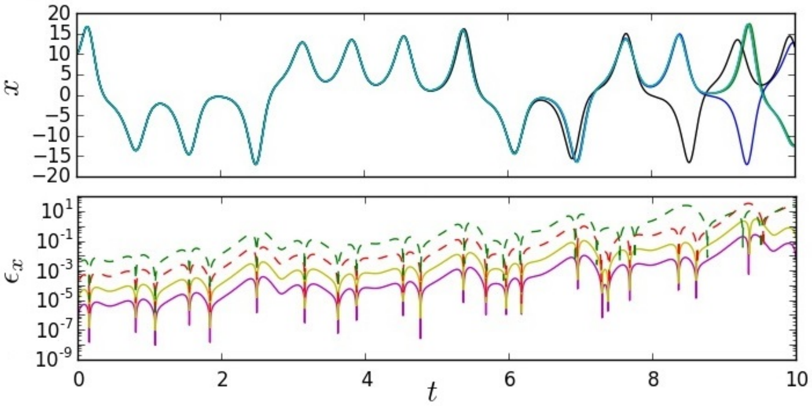


Figure 1.2: Comparison of five different solutions of the Lorenz system, using the parameters $(\sigma, \beta, \rho) = (10.0, 2.7, 28.0)$. In the top panel we show their x-component as a function of the time of integration. We define the reference solution as the one starting from the initial condition $(x_0, y_0, z_0) = (10.0, 10.0, 10.0)$, shown in light blue in this first plot. The other curves are obtained by perturbing the initial y-component by adding to it the following quantities: $\epsilon_1 = 10^{-5}$ (shown in red in the top panel), $\epsilon_2 = 10^{-4}$ (in green), $\epsilon_3 = 10^{-3}$ (in blue), $\epsilon_4 = 10^{-2}$ (in black). The difference between each of these curves and the control one along the x-component (ϵ_x) is shown in the bottom panel, where the color assigned to each curve this time are: green for ϵ_4, red for ϵ_3, yellow for ϵ_2 and violet for ϵ_1. Note that, after $t = 6$, significant differences start to appear and at $t = 8$ the curve corresponding to ϵ_4 is already on a different orbit (the negative one). Time is measured in arbitrary units, like x, y and z components.

clearly stable, and spirals toward an equilibrium point. The other curve, in **(b)**, shows a more complex behavior. However, simply by looking at the figure we cannot say if the behavior is chaotic or not, we need at least a second integration starting from slightly different initial condition. An example of the results from this kind of test are shown in Figure 1.2, where we show the x-component of the system from 5 different integration. Such test reveals that, after an initial period in which the curves are close together, they will start to follow different paths depending on how close they started. In the end, for some of them the gap grows to the point in which the curves complete different number of orbits around the two equilibrium points. So, in this case, two initial conditions differing for an arbitrarily small displacement will eventually result in totally different trajectories after some time. This kind of behavior is the reason for unpredictability: in order to know exactly the evolution of the system, we also need a perfect knowledge of the initial condition. Lorenz [1993] identified in the amplification of the small differences the cause of the lack of periodicity or stationarity in the solution, and therefore the existence of this limit of predictability.

Obviously, the atmosphere is an extremely more complex system, modeled by a significantly higher number of differential equation. Nevertheless, even in this case a predictability limit exists and Lorenz himself proposed some approximated values. He identified in about three days the range in which small errors in the coarser structure (the one resolved by current observing networks) of the atmosphere double. More recent estimates for this doubling time are equal to two days, as seen in Simmons et al. [1995]. So, in presence of this sole contribute, some hope for predictability after some weeks of forecast can exist. The second factor considered is the presence of error in the finer structure (like the position of the clouds), and he evaluated in hour or less the doubling time for this kind of error. This does not forbid skillful long-range forecast, because usually we do not make prediction for these small features. However, the finer structure has an impact on the coarser one, and this is where a great limitation for the predictability range comes from. So, due to this effect, after about one day, some error start to appear in the coarser structure and then they behave as the ones present from the beginning. Because it is nearly impossible to make complete observation of the finer structure, cutting in half the error on this kind of initial conditions is nearly impossible and the result would be disappointing, due to its short doubling time.

However, he proposed that some quantities, such as weekly averages of temperature and total precipitation, can be predictable on longer timescales: this kind of predictability is indeed the focus of the thesis.

1.2.2 Sources of predictability in the subseasonal timescale

A legitimate question arises when dealing with long time ranges, like in our case: "given the predictability limit of two weeks, how can a monthly forecast be skillful?"

The answer lies in both the kind of products we seek and in some phenomena playing a significant role in this timescale. First of all, when dealing with such forecasts, it is always fundamental to distinguish between what can be skillfully predicted and what is unpredictable. For example, daily details of sinoptic-scale features are not a valid candidate for a monthly forecasting system, while a shifts in the probabilities regarding fields averaged over several days may be a more reasonable choice [Hamill et al., 2004]. So, like in many application over similar time ranges, we perform a weekly average of the fields, that removes part of the unpredictable signal from the data. In addition, the second part of the thesis focus on probabilistic forecasts: instead of assigning a precise value for each grid point and time step, we output the probability that the anomaly of the variable considered is above or below some thresholds. This is also a common practice when dealing with this extended range, as underlined in the S2S Research Implementation Plan [2013]. However, in this chapter we do not focus on the statistical techniques for extracting the useful information from the forecasts: the exact procedure will be discussed in details in the following chapters. For now, it is important to emphasize that our forecast products are not the same as the usual medium-range forecast.

In the following paragraphs we describe the physical sources of predictability for sub-seasonal predictions. The first noticeable feature of many of these forecast is the presence of information from the ocean, land and cryosphere, in addition to the atmospheric initial conditions. It can be argued that, unlike what happens for seasonal forecasts, the time range is too short so the variability of the ocean does not bring enough additional information, and as a result it is often difficult to beat persistence [Vitart, 2004]. Nevertheless, an oceanic model and some information regarding sea ice initial conditions is part of many prediction systems and we will later give a brief description of their role.

In addition, there are some patterns of variability that, due to their low-frequency, can contribute to extend the limits of skillful forecast. Depending

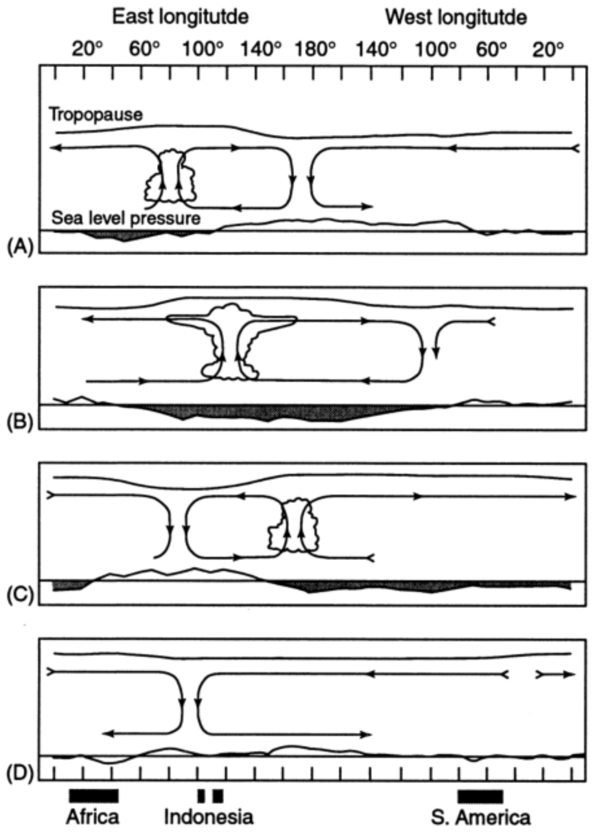


Figure 1.3: Schematized representation of the anomaly pattern of the MJO. The four panels show the time evolution. The time gap between two consecutive ones is equal to 10 days. On the x-axis of each figure there is the longitude, while the y-axis represents the height. The upper line represents the height of the tropopause, while the bottom line shows the surface pressure (areas with negative anomalies of surface pressure highlighted in gray) and finally the streamlines show the zonal-vertical circulation. The figure is directly taken from [Holton et al., 2013, p. 386].

on these, it is possible the presence of windows of enhanced predictability, even if how to determine their presence and especially how to take advantage of their presence is still unclear. They are some of the scientific issues of the S2S Project. We describe, in the following paragraphs, three fundamental patterns and how each of them can affect the prediction of the timescale considered, especially through their interactions. Obviously, this is not an exhaustive list. The persistence of many other patterns can affect predictability, like the Pacific-North American pattern (PNA), the East Atlantic (EA), the West Pacific (WP) and the tropical/Northern Hemisphere (TNH). All of them are mentioned in the S2S Research Implementation Plan [2013]. However, we focus on a subset of all the possible phenomena, and in particular on those considered the more influential or the most interesting ones from a scientific viewpoint. So, we begin by giving a brief overview of all the patterns and processes involved, and then we will discuss their importance in a separate paragraph.

Madden-Julian Oscillation

A particular attention has been given in the scientific literature to the Madden Julian Oscillation (MJO), due to the improvements expected in forecast over the subseasonal timescale resulting from a more skillful prediction of this phenomenon[3].

The phenomenon can be categorized as an important intraseasonal oscillation in the equatorial circulation, on the timescale of 30 to 60 days. A schematized description of its structure can be seen in Figure 1.3, in which each of the panels show a longitude-height section (over the equator) at 10 days interval, containing information about the anomalies from the mean equatorial circulation.

MJO consists in an eastward propagation of a pattern of enhanced and suppressed precipitation, mainly on the Indian and Pacific Oceans. Generally, the anomalous rainfall starts over the Indian Ocean. Looking at some metheorological variables, a negative sea-level pressure can be seen over the region, together with an increase in the convergence of boundary layer moisture, a rise of the temperatures in the troposphere and an increase in height of the tropopause. This pattern propagates eastward, at about 5 m/s, reaching its maximum intensity over the western Pacific and finally weakening when it pass over the central Pacific. However, sometimes the patterns does not disappear completely and traces can be found also over other regions of the Globe [Holton et al., 2013, Chapter 11].

Arctic Oscillation and North Atlantic Oscillation

The weather in the extratropical regions is frequently characterized by recurring circulation patterns. Among them there is the Arctic Oscillation (AO), also called the Northern Hemisphere annular mode.

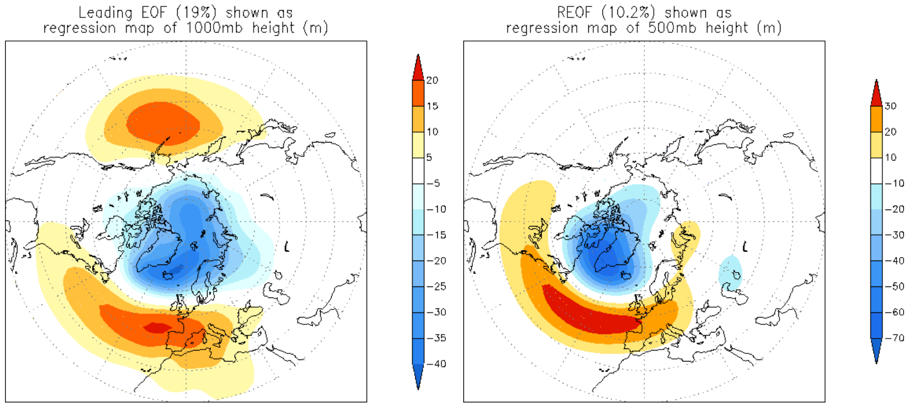
Its index is defined by projecting the daily 1000 mb anomalies over a loading pattern. This pattern is the leading mode of the Empirical Orthogonal Function (EOF) obtained by the analysis of the monthly mean the 1000 mb height anomaly during the period 1979-2000, in the region between 20°and 90°N.[4] We show in Figure 1.4-(a) the loading pattern as defined by the NOAA website.

From a more meteorological point of view, it can be generally described as a pattern of zonal wind circulation around the Arctic, near the 55°N of latitude: in the positive phase strong winds create a "ring" around the Arctic, confining the colder air at the Pole, while in the negative phase this belt weakens and deviates from the ring-shape, allowing southward movement of colder airmasses.

Another interesting pattern over the Northern Extratropics is the North Atlantic Oscillation (NAO). There is not a unique definition of its spatial structure and therefore there is not a universally accepted index. However, many modern definitions are based on the PCA applied to sea level pressure over some

[3] Due to its importance, inside the S2S Project already exists a task force focused on the phenomenon. As for all the information regarding the S2S Project, additional information can be found on the website or in the S2S Research Implementation Plan [2013].

[4] The source of this definition is the NOAA website, and in particular the pages focused on the Arctic Oscillaton and its loading patterns:
http://www.cpc.ncep.noaa.gov/products/precip/CWlink/daily_ao_index/ao.shtml


Leading EOF (19%) shown as regression map of 1000mb height (m)
20
15
10
5
−5
−10
−15
−20
−25
−30
−35
−40
REOF (10.2%) shown as regression map of 500mb height (m)
30
20
10
−10
−20
−30
−40
−50
−60
−70

(a) AO

(b) NAO

Figure 1.4: Loading patterns for the Arctic Oscillation, **(a)**, and North Atlantic Oscillation, **(a)**. The first(AO) is the leading mode of EOF analysis of monthly mean 1000mb height over the period 1979-2000, while the second (NAO) is defined as the first leading mode of REOF (see text) analysis of monthly mean 500mb height over the period 1950-2000. Both pictures, together with their definition, are taken from the NOAA website.
http://www.cpc.ncep.noaa.gov/products/precip/CWlink/pna/nao_loading.html,
http://www.cpc.ncep.noaa.gov/products/precip/CWlink/daily_ao_index/loading.html.

region in the Northern Hemisphere, taking the leading EOF.[5] Alternative definitions are possible, like the one underlying the loading pattern in Figure 1.4-(b), directly taken from the NOAA website[6]. The procedure for computing their index is based on the application of Rotated Principal Component Analysis (RPCA) [7] to monthly standardized 500-mb height in the region between 20°N and 90°N, over the period 1950-2000. Standardized anomalies are obtained using the 1950-2000 climatological daily mean and standard deviation, and a linear interpolation operation of the monthly pattern is applied for computing daily values.

Looking again at the meteorological implication of the pattern, NAO describes the behavior of a pressure dipole. One of the centers is an area of low pressure near Iceland, while the other is a high pressure region located near the Azores. The pressure difference between these two point has historically been used for measuring NAO, although nowadays this simple index has been discarded due to the movements of the pressure centers on a seasonal basis. During a positive NAO, the pressure differece between the two centers rises, while it weakens during the negative phase.

[5] The definition of some indexes, with their advantages and disadvantages, can be found on the NCAR website, in the dedicated section:
https://climatedataguide.ucar.edu/climate-data/hurrell-north-atlantic-oscillation-nao-index-pc-based.

[6] See *http://www.cpc.ncep.noaa.gov/products/precip/CWlink/pna/nao.shtml.*

[7] For additional information on the procedure, see Barnston and Livezey [1987].

Predictability from circulation patterns

Many studies tried to detect how knowledge of the state (or phase) of the patterns just described can influence the predictability over the subseasonal timescale. So, in this section we breifly report some examples from the scientific literature.

We begin with the links between MJO and NAO: in Vitart and Molteni [2010] there is evidence that the MJO produced by the model has a significant impact on the weekly mean probabilistic skill scores over the extratropical region in the Northern Hemisphere, in particular between the 19th and the 25th day of the forecast. Their model was able to show the increased probability of the positive and negative NAO after some specific phases of MJO. In addition, Lin et al. [2010] showed that the skill in predicting NAO over a range up to one month is influenced by the presence of MJO in the initial conditions. This examples show how more accurate initial conditions in the tropics and in particular of the MJO can positively influences forecast over the subseasonal timescale in the extratropical region of the Northern Hemisphere, and in particular of the NAO pattern.

Naturally, this interaction between tropics and extratropics is not limited to a one-way influence. In fact, as suggested again in the S2S Research Implementation Plan [2013], understanding the link in both directions can improve the representation and prediction of the patterns of low-frequency variability in the tropics. This knowledge can, in turn, be used to further improve forecasts in the extratropical regions. Examples of studies focusing on the influence of extratropics on the tropical atmosphere are reported in Lin et al. [2007], Ray and Zhang [2010] and Lin and Brunet [2011], while in Lin et al. [2009] there is a study of the two-way interaction.

In summary, significant potential improvements in subseasonal forecast skills are possible if the models are capable to represent correctly these patterns and their connections, being therefore able to make use of the windows of enhanced predictability. However, these are not the only sources of skillful predictions over the time range considered. In the following paragraphs, we analyze other promising factors, giving for each of them a brief overview of the phenomenon, followed by some examples taken from the scientific literature.

Stratospheric Processes

Stratospheric processes are sources of predictability over the subseasonal timescale. As for the role of the circulation patterns, the importance of the modelization of the stratosphere has not been fully understood, even if an increasing number of studies shows its influence over the extratropical regions. The scientific literature suggests that its impact on averaged skill scores is rather limited, while more significant effects can be seen on the prediction of NAO and the southern annular mode[8], especially during a sudden stratospheric warming.

[8] The Antarctic Oscillation (AAO), also known as southern annular mode is a low-frequency mode of atmospheric variability in the southern extratropics. Its index is obtained by projecting the daily height of the 700 mb surface over its loading pattern. The latter is defined as the leading mode of the EOF for the 700 hPa height over the 1979-2000 period. Additional information can be found on the NOAA website:
http://www.cpc.ncep.noaa.gov/products/precip/CWlink/daily_ao_index/aao/aao.shtml

For example, Baldwin and Dunkerton [2001] showed evidence of propagation, on the monthly timescale, of easterly and westerly anomalies from the stratosphere to the troposphere, followed by negative NAO/AO conditions. In addition, Jung et al. [2010] showed that relaxing the stratosphere to the observed data, for the extratropical region, leads to a reduction of the forecast errors over Europe and the high latitudes. Another interesting result can be found in Hendon et al. [2000], which shows how a better resolution of the stratosphere produces a small reduction of the Root Mean Square Error (RMSE) in the range between the 15th and the 20th day of the forecast. Regarding the role of sudden stratospheric warming, Scaife et al. [2005] studied the impact of this phenomenon on the winter 2005/2006, suggesting its influence on the NAO and cold anomalies over Europe, but other studies produced conflicting results (Jung et al. [2010]).

It is difficult to quantify the improvements that can be expected from stratospheric processes into the extended range forecast models. The major contribution are expected in winter and during sudden stratospheric warming events.

Polar prediction

Historically, atmospheric teleconnections and some typical phenomena of the tropical atmosphere (like MJO) have been regarded as the main sources of forecast skill. This view has lead to a poor understanding of the predictability over the polar region in the sub-seasonal timescale. In addition, many models lack fundamental components for making accurate forecasts over these region, like for example an adequate representation of sea ice[9] , as remarked in the S2S Research Implementation Plan [2013].

Many of the possible sources of forecast skill over these regions are local features: sea ice, snow cover and land surface with the hydrological cycle. Sea ice deserves a special attention: it can represent a source of memory absent at the lower latitude, and Holland et al. [2011] showed that this can result in an increased range of predictive skill. The role of sea ice on the mid-latitudes is, however, uncertain. A similar effect can originate from widespread snow cover anomalies that, due to their radiative and thermal effects, can influence the forecasts over the sub-seasonal timescale, as described in Sobolowski et al. [2010] and Lin and Wu [2011]. Another interesting factor is the dynamics of the local troposphere, which can provide some sources of enhanced predictability due to rather persistent flow anomalies, as shown in Jung et al. [2011].

Moreover, the skill over the poles can be influenced by some elements outside the region. For example, Lin et al. [2010] demonstrated that, over long time ranges, phenomena from the lower latitude like the MJO can have some impact on the polar regions due to the presence of Rossby propagation.

Ocean and SST anomalies

A fundamental element in sub-seasona forecasts is the ocean, and in particular the anomalies in the sea surface temperature (SST).

This component plays different roles depending on the forecast length. For the first 15 days, the accuracy of atmospheric initial conditions are the dominant

[9] Although the presence of a sea-ice model/initialization is rather uncommon, some models (for example the UK Met Office model) already include one of them.

factor for a skillful forecast. Therefore, on the weather-scale, a realistic ocean-atmospheric coupling is not a necessary element and often is not implemented.

On the other hand, seasonal prediction is less affected by the atmospheric initial conditions, while the ocean needs longer timescales for varying significantly its state and this imply that it plays a more important role. So, models for seasonal prediction normally include a realistic representation of the ocean and its coupling with the atmosphere.

The subseasonal timescale lies in between these two extremes, with both the atmosphere and the ocean representing possible sources of predictability. The first reason for the importance of a realistic ocean-atmosphere coupling lies in the different timescales on which these two components affect the forecast. While the atmospheric initial conditions dominate the first period, their influence decreases with time, while the oceanic contribution follows an opposite trend, reaching the maximum influence at the end of the period. In addition, its presence can help the representation of some particular phenomena, for example the MJO, which in turn can result in an enhanced forecast skill. Although these behaviors have not been exactly quantified in the scientific literature, it is reported as one of the important reason for the presence of this coupling in the S2S Research Implementation Plan [2013].

Currently, there are some operational monthly forecasts prediction systems in which atmosphere and ocean are uncoupled while in others they are coupled[10]. The contributions of such features are addressed as part of the Subseasonal to Seasonal Project.

The presence of an ocean-atmosphere coupling can also affect the prediction of SST over the monthly timescale. In medium-range forecasts, due to the slow evolution of the ocean, using a fixed anomaly leads to an high skill of the forecast [Jung and Vitart, 2006]. On the subseasonal range it is unclear if this assertion remains true, or if the coupling significantly enhances the predictability [Kumar et al., 2011]. Moreover, the improvements due to a better predictions of the SST need to be clearly quantified [Chen et al., 2012]. Finally, another effect is shown in Fu et al. [2007] and Woolnough et al. [2007], where the skill in predicting the tropical intraseasonal variability improves due to the usage of a coupled model.

[10] For example, the ECMWF-IFS prediction systems model is coupled to an oceanic one, while sea ice initial conditions are persisted up to the 15th day of the forecast and then relaxed toward climatology. On th other hand, the CNR-ISAC model uses a slab ocean where sea ice is fixes in certain conditions and relaxed toward the climatology in others. Both models are used in the following analysis, and more detailed information on them can be found on the S2S Model Archive page: *https://software.ecmwf.int/wiki/display/S2S/Models*.

Chapter 2

Multi-model ensemble

In the first part of the thesis we combine through linear regression a set of reforecasts from the CNR-ISAC and the ECMWF IFS monthly prediction systems. This procedure is applied to temperature at two meter (T2M), temperature at 850 hPa (T850), and geopotential height at 500 hPa (Z500) fields.

This multi-model implementation serves two different purposes. First of all we want to verify, using non-probabilistic scores, potential improvements in the output fields. The verification, alongside the computation of the regression coeficients, is described in this chapter. Then, we want to provide a suitable basis for extracting probabilities. This procedure, however, will be discussed later in the thesis.

Normally, an ensemble forecast is produced by extracting a finite sample of initial condition, theoretically representing the uncertainty on the initial state of the atmosphere, and then integrating these values for the desired time range. The procedure tries to compensate for the lack of knowledge about the evolution of all the initial-state distributions, by approximating it using only a limited set of trajectories. [Wilks, 2011]

In reality, there are different sources of errors, both in the initial condition distribution and in the model itself. The latter often result in the ensembles produced by a single model being unable to truly represent the evolution of the probability distribution. This can let the ensemble forecast become overconfident. Therefore, multi-model ensembles have been used in the scientific literature [Whitaker et al., 2006] in order to improve the skill of probability forecast. This explain why we implement a linear combination of the reforecast before extracting the probabilities. In addition to this objective, with the aid of non-probabilistic scores, we want to check if improvements exist also in the deterministic fields. Each of the initial ensembles is characterized by its own sources of errors, and a compensation of the two biases is, in theory, possible.

In this chapter, we describe all the steps leading form the single-model ensembles to the multi-model. ERA-Interim reanalyses are used both for computing the regression coefficients and for the verification procedures. In addition, to quantify potential imrpovements, we perform a comparison between these scores and those obtained separately from the two single models.

2.1 Dataset

The dataset used is composed of 268 reforecasts initialized in the winter season (December, January and February) for the years between 1990 and 2010.

The reforecasts of the CNR-ISAC monthly prediction system are part of a larger set (originally created for calibration purposes) that covers the 30-year period from 1981 to 2010. For each year, the initialization date is the 1st of January and the last is the 27th of December, with the time interval between two adjacent entries in the dataset fixed and equal to 5 days. The only exception to this rule is in the leap years, when there is a 6-days gap between the 25th of February and the 2nd of March.

On the other hand, the ECMWF-IFS monthly forecasting system performs twice a week a set of reforecasts covering the past twenty years. [Vitart, 2014] The ones used in this analysis were downloaded from the MARS archive. [1]

Due to the differences in the initialization rules, the final dataset contains only the 268 dates in common between the two systems. Therefore, the reforecasts are not evenly distributed in the time period considered.

Another difference concerns the variables of the two kind of reforecasts: both the CNR-ISAC and the ECMWF-IFS provide data for the temperature at two meter and temperature at 850 hPa, but the geopotential height at 500 hPa field is supplied directly from the CNR-ISAC, while it has to be derived from the geopotential at 500 hPa ($Gh500$) field provided by the ECMWF-IFS. However, this is not a difficult task. From the definition of geopotential height:

$$Z500 = \frac{Gh500}{g^*},$$

where g^* is the constant used in the conversion, close to the standard gravity at mean sea level [2].

2.1.1 Ensemble structure and weekly ensemble mean

From now on, all the analyses will be performed on weekly ensemble means. In this section there is a brief description of all the steps leading from the two ensembles to the two values that will be combined later with the linear regression.

First of all, a brief description of the structure of the reforecasts is presented. The ECMWF-IFS reforecasts are composed of one control member and

[1] We downloaded the dataset containing the ECMWF-IFS reforecast from the Meteorological Archival and Retrieval System (MARS), that is the main repository of meteorological data at ECMWF: *https://software.ecmwf.int/wiki/display/WEBAPI/Access+MARS*.

[2] The geopotential height field produced by Globo is measured in geopotential meters, following the standard for the Subseasonal to Seasonal model archive:
https://software.ecmwf.int/wiki/display/S2S/S2S+geopotential+height.
The conversion from geopotential to geopotential height, when the latter is measured in geopotential meters, is performed by dividing the first field by the value 9.8, which throught the thesis we call g^*. Further information on that conversion can be found on the American Meteorological Society: *http://glossary.ametsoc.org/wiki/Geopotential_height*
Note that, when converting to the standard meter, the conversion constant is $g_0 = 9.80665 m/s^2$ instead of g^* [Holton et al., 2013]. This value is the standard acceleration due to gravity as defined in *The International System of Units (SI): Conversion Factors for General Use* [Butcher et al., 2006, p. 10] and *The international system of units (SI)* [2001, p. 52].

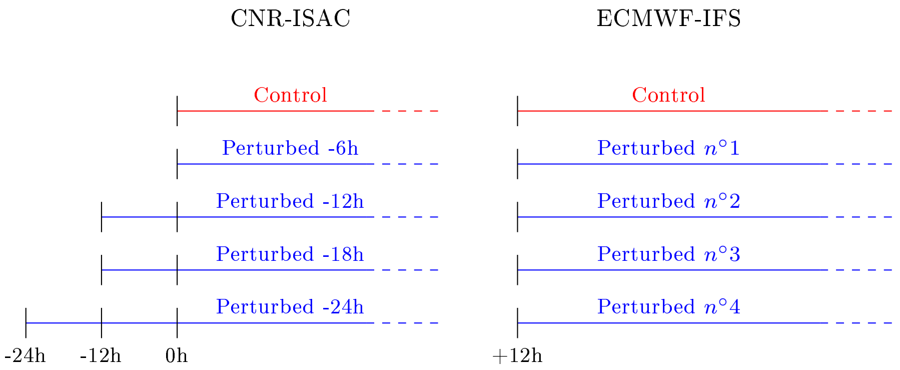


Figure 2.1: Schematized representation of the structure of the two ensembles: on the left the lagged CNR-ISAC ensemble, while on the right the ECMWF-IFS one. Each tick on the lines represent a 12-hour step, which is the resolution of the reforecast.

4 perturbed members, initialized twice a week (on Monday and Thursday) at 00:00UTC. [3] An interesting feature of these reforecast is the variable horizontal resolution of the model. Originally, when a monthly forecasting system was introduced at ECMWF (October 2004), the system and the medium range ensemble prediction system (EPS) were run separately. With the introduction of the Variable Resolution Prediction System (VarEPS) [Buizza et al., 2007], it became possible to change the atmospheric horizontal resolution during the model integration. So, the ECMWF reforecast are run at an higher resolution for the first ten days, and then downgraded until the end of the forecatst (32th day) [Vitart et al., 2008].

The CNR-ISAC Institute produces monthly ensemble forecasts using the atmospheric general circulation model GLOBO. On an operational basis, 40 forecast lagged [4] members are produced, starting from the analyses of GEFS of NOAA-NCEP by using 10 members for each synoptic time of the initialization day. A fixed set of reforecast initialized every 5 days and covering the 30-year period between 1981 and 2010, is used for recalibration. However, this reforecast set contain only a single member, and this represents a rather strong limitation for our analysis. [5] Therefore, in this study we used a different

[3] The model description is provided by the “Model” page on the “Subseasonal to Seasonal Prediction Project” website: *https://software.ecmwf.int/wiki/display/S2S/Models.* Additional information regarding horizontal and vertical resolution, time step and all the technical details can also be found on the same page. A more complete presentation of the ECMWF monthly prediction system and its evolution can be found in the scientific literature, like Vitart [2014], Vitart et al. [2008] and Vitart [2004].

[4] A “lagged ensemble” can be briefly described as an ensemble in which the members are initialized at different times, usually with gaps of 6, 12 or 24 hours so that older forecast can be used to cover the interval before the initialization date. A more detailed description of the procedure known as “lagged ensemble” can be found on [Kalnay, 2003, pp. 231ff].

[5] A brief model description can be found on the “GLOBO monthly forecast” page on the CNR-ISAC webpage:
http://www.isac.cnr.it/dinamica/projects/forecasts/monthly/monthly.htm.
Like for the ECMWF monthly prediction system, additional information are on the “Model” page on the “Subseasonal to Seasonal Prediction Project” website:
https://software.ecmwf.int/wiki/display/S2S/ISAC-CNR+Model+Description.
Both model are, in fact, part of the Subseasonal to Seasonal archive:
https://software.ecmwf.int/wiki/display/S2S/Models.

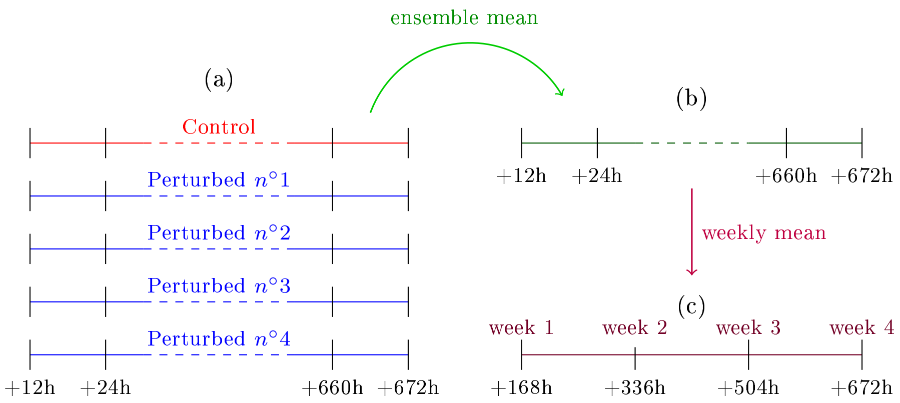


Figure 2.2: Schematized representation of the ensemble mean (from **(a)** to **(b)**, green arrow) and weekly mean (from **(b)** to **(c)**, purple arrow). The diagram is divided in three phases: **(a)** exemplify the structure after the imposition of common bounds for the reforecast, **(b)** is the ensemble mean of **(a)** and **(c)** is the weekly mean of **(b)**. While in **(a)** and **(b)** the ticks represent 12 hours step, in **(c)** the gap between two ticks is equal to 168h hour.

dataset, containing, for each date, 12 perturbed members, besides the control one. This reforecast dataset was produced for the winter season only before the beginning of this study for the purpose of verification of the CNR-ISAC monthly verification system. Like the operational forecasts, it is a lagged ensemble: the control member is initialized at 00:00 UTC, while the integration for the perturbed ones starts at a diferent time for each member (six of them before and the other six after the control). The time gap between two consecutive members is always of 6 hour, so there are six perturbed members starting between -36 and -6 hours from the time of the control and six members between $+6$ and $+36$ hours. The forecast length is 31 days.

Of all this components of the ensemble, only 5 of them have been chosen for the study: the control one and the four perturbed members starting at -24, -18, -12 and -6 hours, The time resolution of the reforecast is of 12 hours in order to have a direct equivalent of the ECMWF-IFS system. The structure of the two ensembles is schematized in Figure 2.1.

The differences just described have to be leveled out before proceeding with the analysis. So, the first step is the decision of a common 1° lat-lon grid. This is achieved by a bilinear interpolation of the CNR-ISAC reforecasts on the desired grid. The ECMWF-IFS ones are already downloaded at the common resolution. + Then, common bounds are imposed on the reforecasts: for each entry in the dataset, the first value is at 12:00 of the initialization date and the last is 660 hours afterwards, at 00:00 UTC of the 29$^{\text{th}}$ day of the reforecast. So, for example, for the 1$^{\text{th}}$ of January, the forecast starts at 12:00 UTC of that date and ends at 00:00 UTC of the 29$^{\text{th}}$ of the same month. In Figure 2.2, a schematic representation of the result is shown in **(a)**.

With this new dataset containing data sharing the same structure, the calculation of the regression input can begin. The starting operation is an ensemble mean performed on the members of each of the two forecasting systems. This is followed by a weekly mean achieved by averaging the reforecast over the closed intervals $[+12h, +168h]$, $[+180h, +336h]$, $[+348h, +504h]$ and $[+516h, +672h]$, where the hours are computed from the 00:00 of the initialization date.

Thus every reforecast now contains only four values, each one referring to the ensemble mean value for one week. The two steps are shown in Figure 2.2 in **(b)** and **(c)**.

2.1.2 The reanalysis

The final ingredients for the realization of the multi-model ensemble are the ERA-Interim re-analyses. [6] For each date covered by the reforecasts in the dataset, the value of the reanalyses for the 00:00 and 12:00 UTC of each day has been downloaded from the MARS archive. This values are combined into files with the same structure of the corresponding reforecast. Obviously, a weekly mean is performed on the elements of the resulting dataset. Naturally, during the download the values have been chosen in such a way that there is no need for adjust the forecast time limits.

Like the ECMWF-IFS monthly forecasting system, ERA-Interim provides the geoponential fields at 500hPa. So, following the same procedure described for the reforecasts, the values are converted to geopotential height dividing by the usual constant, g^*.

2.2 Linear regression

Finally, the multi-model is computed through linear regression. [7]

Some notation is introduced to simplify the discussion: $\boldsymbol{M}_{\mathrm{E}}(w,i,j,d)$ and $\boldsymbol{M}_{\mathrm{C}}(w,i,j,d)$ are respectively the ECMWF-IFS and the CNR-ISAC weekly ensemble mean, and $\boldsymbol{O}(w,i,j,d)$ are the weekly averaged verifying reanalysis. All of them are functions of the week $w = 1, 2, 3, 4$, the latitude $i = -90, ..., 90$, the longitude $j = 0, ..., 360$ and the date d.

From these values, the multi-model prediction for the weekly mean of the field ($MM(w,i,j,d)$) can be obtained through:

$$\boldsymbol{MM}(w,i,j,d) = \boldsymbol{C}_0^*(w,i,j) + \boldsymbol{C}_1^*(w,i,j)\boldsymbol{M}_{\mathrm{E}}(w,i,j,d) + \boldsymbol{C}_2^*(w,i,j)\boldsymbol{M}_{\mathrm{C}}(w,i,j,d)$$

However, this is not the formula actually used in the computation. An equivalent version, involving the anomaly field instead of the field itself is preferred, because it needs only two coefficients.

Thus, another small step is required: in order to obtain the anomaly, the mean over the training period has to be subtracted from the instantaneous field. Note that “training period” has been used rather than “the full dataset”. This choice is due to the need of cross validation, that reduces the set in which the algorithm is trained each time. It will be all explained later in the result section, in order to avoid confusion with the current explanation of the regression.

[6] Aditional information of ERA-Interim reanalysis are on the dedicated ECMWF website: *http://www.ecmwf.int/en/research/climate-reanalysis/era-interim*.
A detailed description can be found in Dee et al. [2011].

[7] A similar approach is used in Whitaker et al. [2006], where linear regression is used to combine ECMWF and NCEP reforecasts.

So, naming with $\overline{\boldsymbol{M}}_{\mathrm{E}}(w,i,j)$, $\overline{\boldsymbol{M}}_{\mathrm{C}}(w,i,j)$ and $\overline{\boldsymbol{O}}(w,i,j)$ the time mean (over the reference period) of $\boldsymbol{M}_{\mathrm{E}}(w,i,j,d)$, $\boldsymbol{M}_{\mathrm{C}}(w,i,j,d)$ and $\boldsymbol{O}(w,i,j,d)$ respectively, the anomalies are given by:

$$\begin{aligned}
\boldsymbol{X}_1(w,i,j,d) &= \boldsymbol{M}_{\mathrm{E}}(w,i,j,d) - \overline{\boldsymbol{M}}_{\mathrm{E}}(w,i,j),\\
\boldsymbol{X}_2(w,i,j,d) &= \boldsymbol{M}_{\mathrm{C}}(w,i,j,d) - \overline{\boldsymbol{M}}_{\mathrm{C}}(w,i,j),\\
\boldsymbol{Y}(w,i,j,d) &= \boldsymbol{O}(w,i,j,d) - \overline{\boldsymbol{O}}(w,i,j).
\end{aligned}$$

Using these values, the formula for the multi-model anomaly is:

$$\boldsymbol{X}_{\mathrm{MM}}(w,i,j,d) = \boldsymbol{C}_1(w,i,j)\boldsymbol{X}_1(w,i,j,d) + \boldsymbol{C}_2(w,i,j)\boldsymbol{X}_2(w,i,j,d). \qquad (2.1)$$

It can be proved that $\boldsymbol{C}_1(w,i,j) = \boldsymbol{C}_1^*(w,i,j)$ and $\boldsymbol{C}_2(w,i,j) = \boldsymbol{C}_2^*(w,i,j)$. In addition, the third coeficient of the original regression can be obtained using:

$$\boldsymbol{C}_0^*(w,i,j) = \overline{\boldsymbol{O}}(w,i,j) - \boldsymbol{C}_1(w,i,j)\overline{\boldsymbol{M}}_{\mathrm{E}}(w,i,j) - \boldsymbol{C}_2(w,i,j)\overline{\boldsymbol{M}}_{\mathrm{C}}(w,i,j).$$

2.2.1 Minimization of the cost function

The weighting factors ($\boldsymbol{C}_1$ and $\boldsymbol{C}_2$) chosen are the ones that minimize the cost-function

$$\boldsymbol{J}(w,i,j) = \sum_{d=1}^{m}\big(\boldsymbol{X}_{\mathrm{MM}}(w,i,j,d) - \boldsymbol{Y}(w,i,j,d)\big)^2.$$

In the previous formula (and in the following ones), m stands for the dimension of the training set.

As an intermediate step, some auxiliary quantities are defined:

$$\begin{aligned}
\boldsymbol{P}_{11}(w,i,j) &= \sum_{d=1}^{m}\big(\boldsymbol{X}_1(w,i,j,d)\big)^2,\\
\boldsymbol{P}_{22}(w,i,j) &= \sum_{d=1}^{m}\big(\boldsymbol{X}_2(w,i,j,d)\big)^2,\\
\boldsymbol{P}_{12}(w,i,j) &= \sum_{d=1}^{m}\big(\boldsymbol{X}_1(w,i,j,d)\boldsymbol{X}_2(w,i,j,d)\big),\\
\boldsymbol{P}_{\mathrm{Y1}}(w,i,j) &= \sum_{d=1}^{m}\big(\boldsymbol{Y}(w,i,j,d)\boldsymbol{X}_1(w,i,j,d)\big),\\
\boldsymbol{P}_{\mathrm{Y2}}(w,i,j) &= \sum_{d=1}^{m}\big(\boldsymbol{Y}(w,i,j,d)\boldsymbol{X}_2(w,i,j,d)\big),\\
\boldsymbol{\Delta}(w,i,j) &= \boldsymbol{P}_{11}(w,i,j)\boldsymbol{P}_{22}(w,i,j) - (\boldsymbol{P}_{12}(w,i,j))^2.
\end{aligned}$$

Then, using some algebra, it can be proven that the solution is:

$$\begin{aligned}
\boldsymbol{C}_1(w,i,j) &= \frac{\boldsymbol{P}_{\mathrm{Y1}}(w,i,j)\boldsymbol{P}_{11}(w,i,j) - \boldsymbol{P}_{12}(w,i,j)\boldsymbol{P}_{\mathrm{Y1}}(w,i,j)}{\boldsymbol{\Delta}(w,i,j)}\\
\boldsymbol{C}_2(w,i,j) &= \frac{\boldsymbol{P}_{\mathrm{Y2}}(w,i,j)\boldsymbol{P}_{22}(w,i,j) - \boldsymbol{P}_{12}(w,i,j)\boldsymbol{P}_{\mathrm{Y2}}(w,i,j)}{\boldsymbol{\Delta}(w,i,j)}
\end{aligned}$$

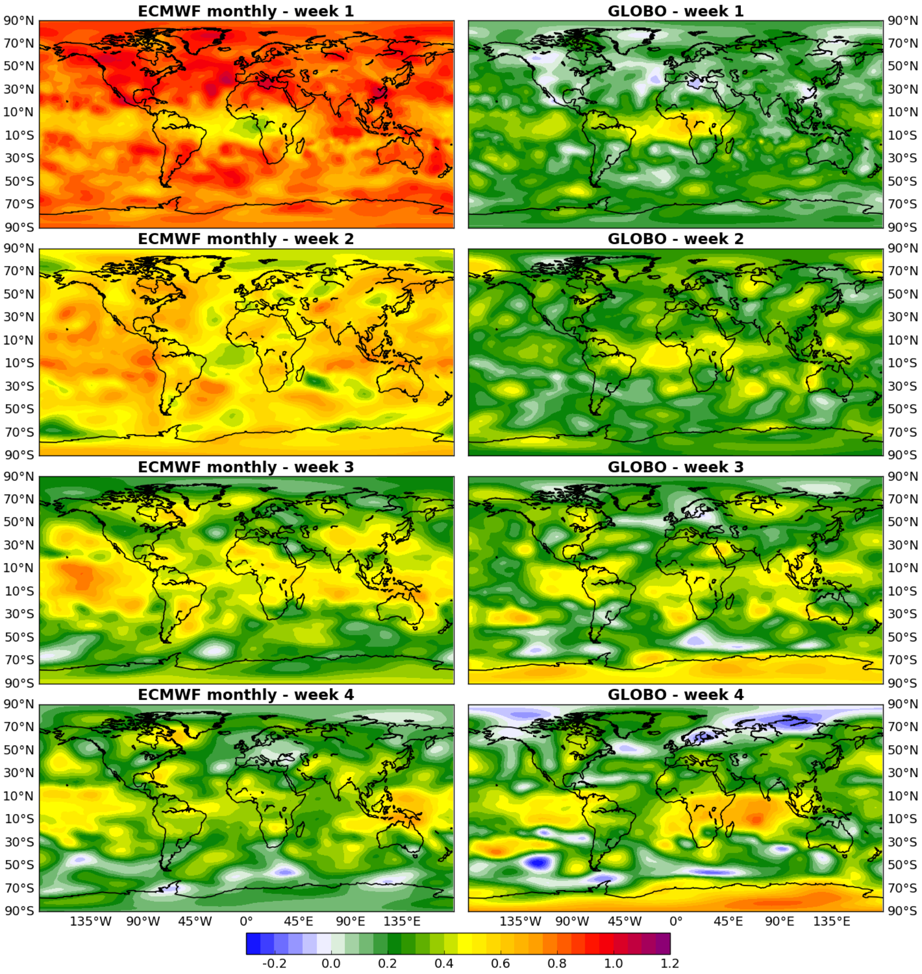


Figure 2.3: Regression coefficients for the 500 hPa Geopotential height anomalies. Values on the left refer to the ECMWF-IFS model ($\boldsymbol{C}_1(w,i,j)$), on the right to GLOBO ($\boldsymbol{C}_2(w,i,j)$).

2.3 Results

Initially, the linear regression is applied using all the winters for the training. Presuming that the performances of the algorithm theoretically improve with larger datasets, the weighting coefficients obtained in this way are probably the best estimate that can be made with the data available.

The result are shown in three different figures, one for each variable: Figure 2.3 shows the geopotential height at 500 hPa, Figure 2.4 the temperature at two meter, Figure 2.5 the temperature at 850 hPa.

In all of them the major role that $\boldsymbol{X}_1$ (the ECMWF-IFS system) plays in defining the final multi-model field is evident. The contribution is more obvious in the first two weeks, probably due to the higher resolution of the ECMWF-IFS model in the first 10 days of the forecast. However, the geographical distribution of the local maxima is rather different between the two models . There is not a

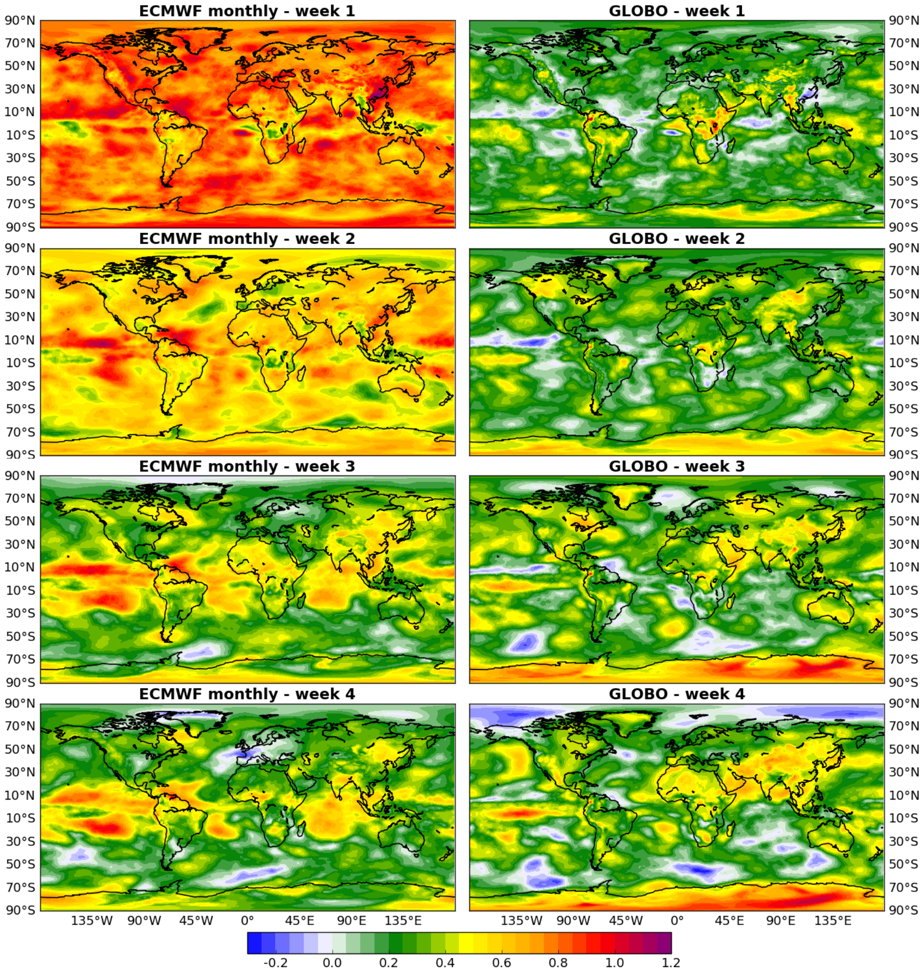


Figure 2.4: As in Figure 2.3, but for the temperature at 850hPa.

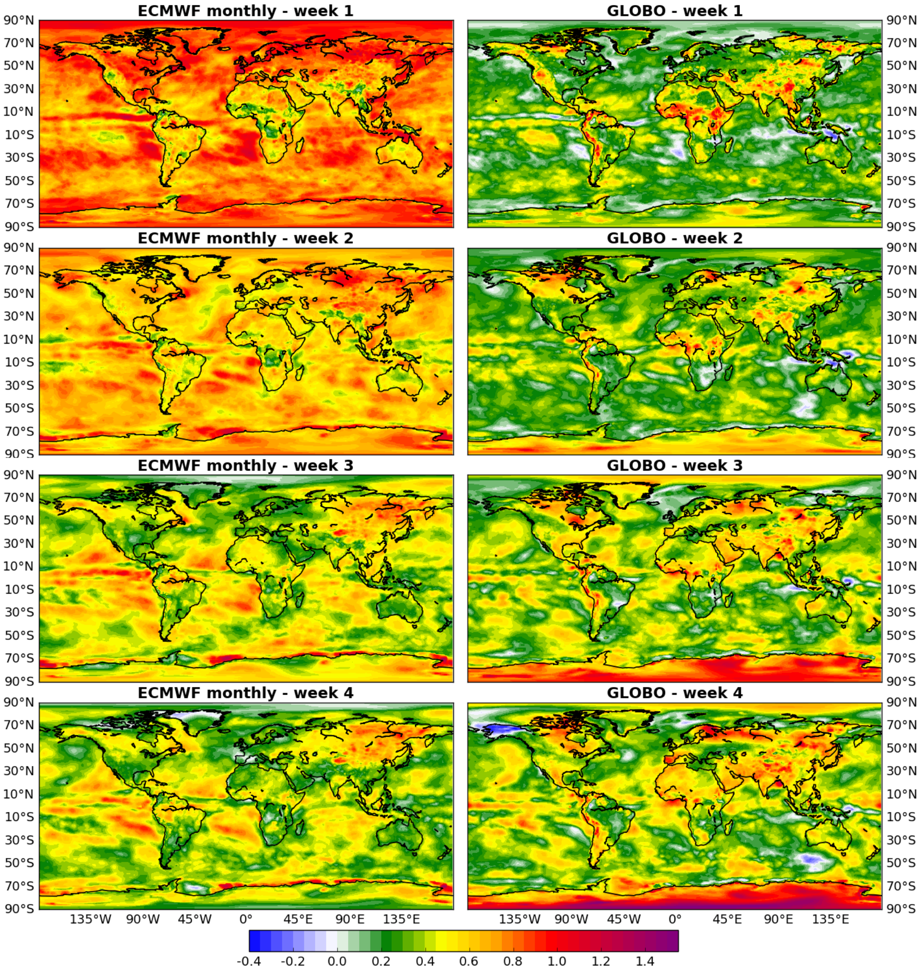


Figure 2.5: As in Figure 2.3, but for the temperature at 2m.

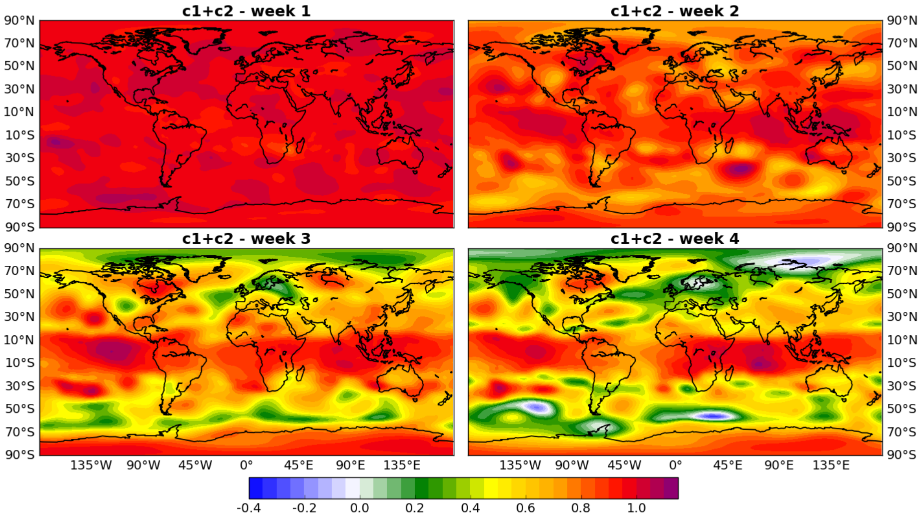


Figure 2.6: Sum of the regression coefficients for the 500 hPa Geopotential height anomalies $(\boldsymbol{C}_1(w,i,j) + \boldsymbol{C}_2(w,i,j))$. Each week is shown in a different panel.

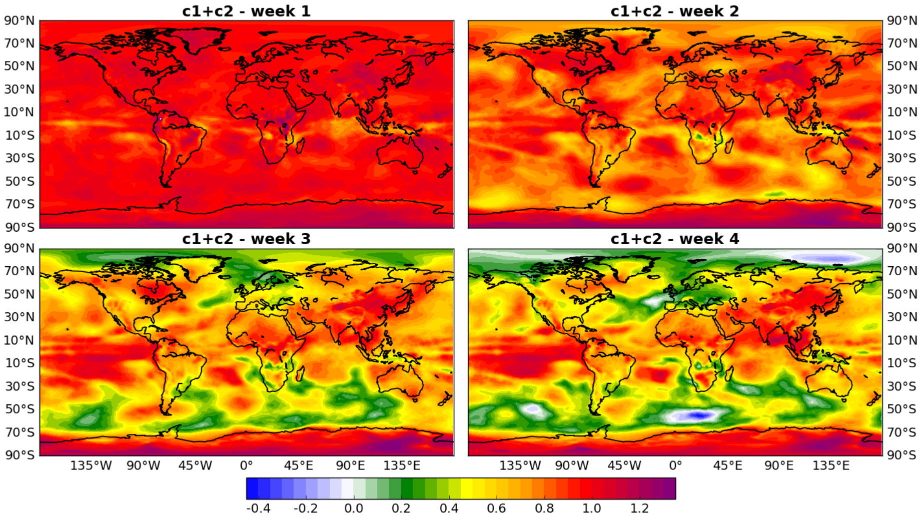


Figure 2.7: As in Figure 2.6, but for the temperature at 850hPa.

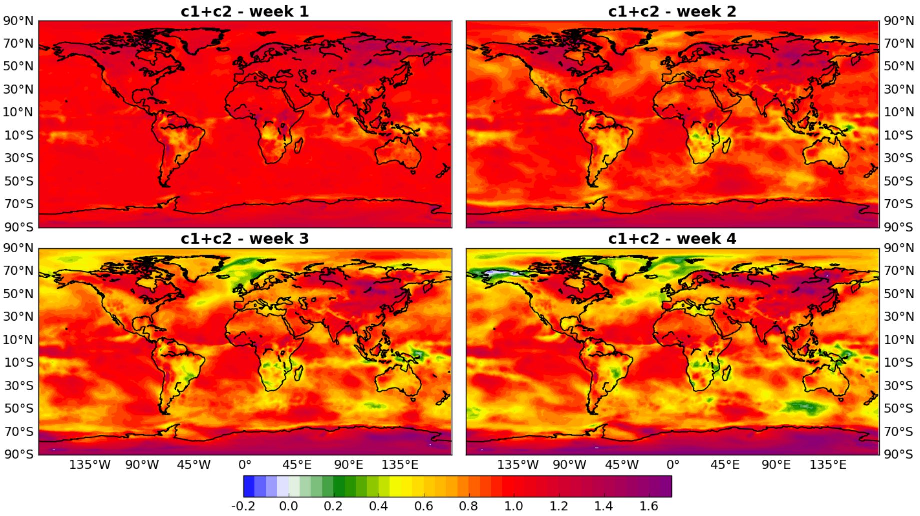


Figure 2.8: As in Figure 2.6, but for the temperature at 2m.

single model that weights more than the other sistematically in aa grid points. On the contrary, there are some regions where the multimodel is more similar to one of its components.

In addition, Figures 2.6 - 2.7 and 2.8 report the quantity $\boldsymbol{C}_1 + \boldsymbol{C}_2$ for $Z500$, $T850$ and $T2M$, respectively. There is not a specific physical or statistical meaning linked to these values, but a rule of thumb can be derived from their definition: when the sum is near zero, the climatology is the best possible forecast given the data.

Some interesting features can be seen in the maps. Considering $Z500$, one of the most evident feature is a region near the equator where $\boldsymbol{C}_1 + \boldsymbol{C}_2$ is close to one for all weeks. A similar phenomenon (with smaller maxima) can be observed near the center of the Antarctic. In the fourth week, there are some areas where the sum of the coefficients is almost null or even negative: the northen part of Europe (also visible in the third week), in some of the ocean slighlty north of the Anctartic and north of Siberia. This behaviour hints the lack of predictability over these regions: $\boldsymbol{C}_1 + \boldsymbol{C}_2 = 0$ imply that the multi-model anomaly is also equal to zero, therefore the climatology represents the best prediction accordig to the linear regression.

The pattern for $T850$ is rather similar, even if the coordinates of the local maxima and minima are slightly shifted and their shape is different. In particular, over the Antacrctic the sum is significantly greater, while over the equator the areas greater or equal to one are more sparsely distributed.

For $T2M$ (Figure 2.8) the pattern is significantly different. The two more noticeable features are the maxima over part of Asia and the Antactica, where the sum reach values greater than 1.5 (unseen for the other variables) and increases with the week number. Besides high values, close to one, over the equator, there are no patterns as marked as for $Z500$ $T850$.

Note that this first analysis is purely qualitative: $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ presented are not tested on an indipendent dataset and the performance of the algorithm is not evaluated. The focus of the section is on the geographical distribution of the weights of each model in the final product. A discussion about the (non-probabilistic) statistical scores is presented in the next section.

2.4 Verification

The results cannot be divided from the statistical evaluation of the performances. The latter requires that we split the dataset into "training" and "validation" sets. This "cross-validation" approach is chosen because it gives an estimation of how well the algorithm generalizes on data that it has not seen previously. (Wilks [2011], Efron and Gong [1983], Elsner and Schmertmann [1994])

First of all, the dataset is split into single winter seasons, each of them including December from one year and January and February from the next one. In the winters 1990-1991, 1991-1992 and 2010-2011 there are only a few dates and they are excluded from the subsequent analysis.[8]

The remaining 18 winters are used for the "k-fold cross-validation". The original version of this algorithm expects a random partition of the dataset in k subsets. Then, one of them is chosen as validation set while the remaining k-1 constitute the training set, on which the regression coefficients are computed. This procedure is repeated k times, choosing a different validation winter each time. (Wilks [2011], Zhang [1993])

However, in this study a different approach to the partition of the dataset is chosen: each of the 18 winters is used as a subset for the k-fold cross-validation, with k = 18.

Although this choice implies that the number of dates in each set is not constant, this subdivision resembles rather closely what the system would encounter in a operational situation.

Hypothetically, two consecutive reforecast can be slightly correlated with each other. In this situation, a set coefficient trained on the first reforecast can perform excessively well on the second, and the performance would not resemble the behaviour of the algorithm with unseen data. The decision of dividing the dataset in winters tries to prevent that. There are several months between reforecast from different winters, this assures us that the probability of a systematic correlation between any data from the training set and the ones used for validation is neglegible.[9]

In summary, the linear regression is performed on the 17 training winters, the resulting $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ are the used for computing the multi-model fields for the validation winter. This fields are the used with the ERA-Interim reanalysis for the calculation of two non-probabilistic scores: root mean square error (RMSE) and anomaly correlation (AC) [Wilks, 2011]. The same scores are also computed for the two model separately, for comparison purposes. The procedure is repeated 18 times, using always a different winter for validation, and

[8] The number of dates for the winters 1990-1991, 1991-1992 and 2010-2011 are respectively 2, 5 and 3. For comparison, the winters from 1993-1994 to 2007-2008 contain 15 dates each, the winter 2008-2009 contain 13 dates and the each of remaining two (1992-1993 and 2009-2010) contain 10 dates.

[9] This approach is vaguely similar to the one described in Wilks [2011] when dealing with serially correlated data. However, instead of choosing a number of consecutive observation L to leave out each repetition of the algorithm, we can directly split the dataset in winters and, between elements of different winters, a gap of several months already exists.

finally all the results are averaged, therefore producing a single value for each score.

2.4.1 RMSE

The first of the non-probabilistic scores is the root mean square error. It is commonly used as measure of accuracy, having the desirable property of retain the units of the forecast variable and therefore being simply interpretable [Wilks, 2011].

Altought it is the simplest and most commonly used one, in this specific case there is a little difference in how the mean is performed. Due to the nature of the lat-lon grid, each point has to be weighted with the area associated with it before the average. In fact, the nearest to the poles a point is, the smaller the area under a unit-degree square on the grid is. In other words, the actual distance on the sphere between two meridians reaches its maximum on the equator and the minimum in the poles.

In order to compensate this effect, the weights matrix is set to:

$$\boldsymbol{W}(i,j) = \cos\big(\phi(i)\big),$$

where $\phi(i)$ is the latitude of the grid point (i,j), in radiants.

The quantity averaged using these weights is the square error between the reforecast field ($\boldsymbol{X}$, it can be the multi-model or one of the two models) and the reanalysis one, and it is given by:

$$\boldsymbol{SE}(w,i,j,d_{\mathrm{V}}) = \big(\boldsymbol{X}(w,i,j,d_{\mathrm{V}}) - \boldsymbol{Y}(w,i,j,d_{\mathrm{V}})\big)^2,$$

So, the root mean square error for a specific validation winter is computed using:

$$\boldsymbol{rmse}(w) = \frac{1}{m_{\mathrm{V}}} \sum_{d_V=1}^{m_{\mathrm{V}}} \sqrt{\frac{\sum_{i=i_{\min}}^{i_{\max}} \sum_{j=j_{\min}}^{j_{\max}} \boldsymbol{W}(i,j)\boldsymbol{SE}(w,i,j,d_{\mathrm{V}})}{\sum_{i=i_{\min}}^{i_{\max}} \sum_{j=j_{\min}}^{j_{\max}} \boldsymbol{W}(i,j)}}$$

At the end of the procedure, for each of the 18 possible choice of the validation winter, there are three vectors: $\boldsymbol{rmse}_{\mathrm{MM}}^{(\mathrm{k})}$, $\boldsymbol{rmse}_{\mathrm{E}}^{(\mathrm{k})}$ and $\boldsymbol{rmse}_{\mathrm{C}}^{(\mathrm{k})}$, representing respectively the scores for the multimodel, the ECMWF-IFS model and the CNR-ISAC one. The superscript $\mathrm{k} = 1, ..., 18$ indicates wich winter is used for validation, with $k = 0$ representing the first (1992-1993) and $k = 18$ the last (2009-2010). From these triplets of vector, the three final scores are derived by averaging over all the 18 values:

$$\overline{\boldsymbol{rmse}}_{\mathrm{MM}} = \frac{1}{18} \sum_{k=0}^{18} \boldsymbol{rmse}_{\mathrm{MM}}^{(\mathrm{k})},$$

$$\overline{\boldsymbol{rmse}}_{\mathrm{E}} = \frac{1}{18} \sum_{k=0}^{18} \boldsymbol{rmse}_{\mathrm{E}}^{(\mathrm{k})} \quad \text{and} \quad \overline{\boldsymbol{rmse}}_{\mathrm{C}} = \frac{1}{18} \sum_{k=0}^{18} \boldsymbol{rmse}_{\mathrm{C}}^{(\mathrm{k})}.$$

The procedure just described can be theoretically applied for an arbitrary choice of the couples $(i_{\min}, i_{\max})$ and $(j_{\min}, j_{\max})$, the borders of the region on which the spatial average is performed.

w	$\overline{rmse}_{\mathrm{MM}}$	$\overline{rmse}_{\mathrm{E}}$	$\overline{rmse}_{\mathrm{C}}$
1	22.5	22.6	30.4
2	67.1	70.8	76.8
3	82.8	90.5	90.5
4	87.0	96.4	94.2

(a) Northern Hemisphere

w	$\overline{rmse}_{\mathrm{MM}}$	$\overline{rmse}_{\mathrm{E}}$	$\overline{rmse}_{\mathrm{C}}$
1	19.9	20.1	24.4
2	52.7	55.8	59.6
3	61.5	68.4	66.2
4	63.0	70.2	66.9

(b) Southern Hemishpere

w	$\overline{rmse}_{\mathrm{MM}}$	$\overline{rmse}_{\mathrm{E}}$	$\overline{rmse}_{\mathrm{C}}$
1	3.95	4.09	5.17
2	8.56	9.05	10.3
3	10.9	11.8	12.3
4	11.7	12.9	12.8

(c) Equatorial Belt

w	$\overline{rmse}_{\mathrm{MM}}$	$\overline{rmse}_{\mathrm{E}}$	$\overline{rmse}_{\mathrm{C}}$
1	25.2	25.4	35.1
2	81.3	85.7	91.1
3	97.1	105	106
4	100	111	108

(d) Europe

Table 2.1: Root mean square errors for the geopotential height at 500 hPa ($Z500$) anomalies, averaged over the 18 validation winters. The four table present the spatial average over the four different regions defined during the description of the non-probabilistic validation scores. The first column always shows, in blue, the week for the entire row. Each of the remaining column refer to a different model: in the first one the are the values for the multi-model ($\overline{rmse}_{\mathrm{MM}}$), in the second the ECMWF-IFS ones ($\overline{rmse}_{\mathrm{E}}$) and in the third the CNR-ISAC ones ($\overline{rmse}_{\mathrm{C}}$). The value corresponding to the best performances (the lowest ones) for each row is highlighted in red.

w	$\overline{rmse}_{\mathrm{MM}}$	$\overline{rmse}_{\mathrm{E}}$	$\overline{rmse}_{\mathrm{C}}$
1	1.08	1.11	1.45
2	2.70	2.85	3.06
3	3.29	3.59	3.52
4	3.49	3.84	3.71

(a) Northern Hemisphere

w	$\overline{rmse}_{\mathrm{MM}}$	$\overline{rmse}_{\mathrm{E}}$	$\overline{rmse}_{\mathrm{C}}$
1	0.88	0.91	1.13
2	1.83	1.94	2.09
3	2.09	2.32	2.27
4	2.12	2.37	2.27

(b) Southern Hemishpere

w	$\overline{rmse}_{\mathrm{MM}}$	$\overline{rmse}_{\mathrm{E}}$	$\overline{rmse}_{\mathrm{C}}$
1	0.51	0.54	0.75
2	0.84	0.90	1.07
3	0.99	1.08	1.14
4	1.02	1.12	1.13

(c) Equatorial Belt

w	$\overline{rmse}_{\mathrm{MM}}$	$\overline{rmse}_{\mathrm{E}}$	$\overline{rmse}_{\mathrm{C}}$
1	1.03	1.06	1.39
2	2.70	2.85	3.03
3	3.23	3.55	3.46
4	3.32	3.70	3.51

(d) Europe

Table 2.2: As in Table 2.1, but for the temperature at 850 hPa ($T850$).

w	$\overline{rmse}_{\mathrm{MM}}$	$\overline{rmse}_{\mathrm{E}}$	$\overline{rmse}_{\mathrm{C}}$
1	1.35	1.45	2.01
2	2.79	2.90	3.20
3	3.35	3.59	3.60
4	3.52	3.78	3.77

(a) Northern Hemisphere

w	$\overline{rmse}_{\mathrm{MM}}$	$\overline{rmse}_{\mathrm{E}}$	$\overline{rmse}_{\mathrm{C}}$
1	0.59	0.65	0.89
2	1.08	1.16	1.36
3	1.25	1.39	1.48
4	1.31	1.47	1.54

(b) Southern Hemishpere

w	$\overline{rmse}_{\mathrm{MM}}$	$\overline{rmse}_{\mathrm{E}}$	$\overline{rmse}_{\mathrm{C}}$
1	0.41	0.48	0.52
2	0.62	0.69	0.73
3	0.72	0.81	0.80
4	0.75	0.85	0.81

(c) Equatorial Belt

w	$\overline{rmse}_{\mathrm{MM}}$	$\overline{rmse}_{\mathrm{E}}$	$\overline{rmse}_{\mathrm{C}}$
1	1.25	1.32	1.86
2	2.78	2.89	3.14
3	3.35	3.61	3.54
4	3.40	3.66	3.58

(d) Europe

Table 2.3: As in Table 2.1, but for the temperature at 2 metre (T2M).

In practice, in this study four different regions have been chosen, each with its own superscript:

- the Northern Hemisphere (NH), arbitrarily defined as the area ranging from 20°N to 90°N in latitude (and naturally from 0 to 360 in longitude),
- the Southern Hemisphere (SH), from 20°S to 90°S in latitude,
- the Equatorial Belt (EB), 20°S to 20°N,
- the Europe (EU), whose limits are 30 - 80°N and 20°W - 60°E.

The results are summarized in Table 2.1 - 2.2 - 2.3. The most evident feature is that the multi-model outperforms the two models everywhere. This is not surprising. First of all, in its calculation the cost function minimized is the square error, which is an ingredient also of the RMSE. Obviously, the minimization of square error is performed on the training set while the final score is computed on the validation winter, but an improvement in the performance compared to a single models is still expected. Another effect to consider is that the multi-model contains the information from ten ensemble members, while $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ are computed starting from five members each. It is, in fact, known that an increase of the ensemble size can result in improvements in the performances, depending on the specific measure used. [Buizza and Palmer, 1998]

Looking more closely, a rough dividing line can be drawn between the first two weeks and the final ones. In the first period, the RMSE is naturally smaller, with a clear increase from the first to the second week. The ECMWF model outperforms always GLOBO, and its score are very similar to the multi-model one. In the third and fourth week the pattern is different. The first distinction is in the relation between $\overline{rmse}_{\mathrm{E}}$ and $\overline{rmse}_{\mathrm{C}}$. In the third week the two values are close together and none of them is always below the other, unlike the previous case. In the fourth week the situation is even different: while the two scores are always similar, $\overline{\boldsymbol{rmse}}_{\mathrm{C}}$ is lower than $\overline{\boldsymbol{rmse}}_{\mathrm{E}}$ most of the time. Moreover, the gap between $\overline{\boldsymbol{rmse}}_{\mathrm{MM}}$ and $\overline{\boldsymbol{rmse}}_{\mathrm{E}}$ is often more marked than in the first two weeks. Finally, for all the models considered, the RMSE increase over the final two weeks is less sharp than the one between the first and the second, or between the second and the third.

This clear difference in behaviour between the two times ranges is expected. The first two weeks cover a period that correspond almost entirely with the medium range forecast, while the following ones (often referred as extended range) are completely outside the deterministic timescale. Due to the loss of the memory of the atmospheric initial condition, it is a difficult time range for realizing forecasts [Vitart, 2004]. So, a sharp drop in performances cannot be avoided, and a direct comparison between perfromances from the two period can be unfair.

2.4.2 Anomaly correlation

The second non-probabilistic score presented in this study is the anomaly correlation. It is commonly used for measuring similarities in the patterns of the anomalies between the forecast and the verifying values. In the scientific literature there are two different scores sharing the name of "anomaly correlation", and this can lead to some confusion. In this study, the name refers to the "uncentered anomaly correlation", in which the field averaged over the region of interest is not subtracted from the anomalies [Wilks, 2011]. This kind of AC was first defined in Miyakoda et al. [1972], where the score was originally called "correlation for the anomaly"[10].

Because the fields are again on a lat-lon grid, in the computation of the score the spatial average is weighted using $\boldsymbol{W}(i,j)$.

The score is computed firstly for each validation winter separately, like in the previous case. As an intermediate step, three matrix are computed:

$$\boldsymbol{P}_{\mathrm{XY}}(w,d) = \sum_{i=i_{\min}}^{i_{\max}} \sum_{j=j_{\min}}^{j_{\max}} \big(\boldsymbol{X}(w,i,j,d)\boldsymbol{Y}(w,i,j,d)\big)\boldsymbol{W}(i,j),$$

$$\boldsymbol{P}_{\mathrm{XX}}(w,d) = \sum_{i=i_{\min}}^{i_{\max}} \sum_{j=j_{\min}}^{j_{\max}} \big(\boldsymbol{X}(w,i,j,d)\big)^2\boldsymbol{W}(i,j),$$

$$\boldsymbol{P}_{\mathrm{YY}}(w,d) = \sum_{i=i_{\min}}^{i_{\max}} \sum_{j=j_{\min}}^{j_{\max}} \big(\boldsymbol{Y}(w,i,j,d)\big)^2\boldsymbol{W}(i,j).$$

Naturally, the notation is the same of the previous section. Then, the anomaly correlation is given by:

$$\boldsymbol{ac}(w) = \frac{1}{m_{\mathrm{v}}} \sum_{d=1}^{m_{\mathrm{v}}} \frac{\boldsymbol{P}_{\mathrm{XY}}(w,d)}{\boldsymbol{P}_{\mathrm{XX}}(w,d)\boldsymbol{P}_{\mathrm{YY}}(w,d)}$$

[10] The other type of anomaly correlation, not used in this analysis, is called "centered anomaly correlation", where the mean over a given map of M gridpoints is subtracted from the anomaly fields of both the forecast and the verifying observation [Wilks, 2011]. This kind of AC was first introduced in Namias [1952].

For each of the model there are 18 anomlay correlation vector, distinguished by the superscript (k): $\boldsymbol{ac}^{(k)}_{\text{MM}}$, $\boldsymbol{ac}^{(k)}_{\text{E}}$ and $\boldsymbol{ac}^{(k)}_{\text{C}}$, the subscript notation is equal to the $\boldsymbol{rmse}$ vectors one.

These quantities are averaged over all winters, and the final products are:

$$\overline{\boldsymbol{ac}}_{\text{MM}} = \frac{1}{18}\sum_{k=1}^{18} \boldsymbol{ac}_{\text{MM}}, \quad \overline{\boldsymbol{ac}}_{\text{E}} = \frac{1}{18}\sum_{k=1}^{18} \boldsymbol{ac}_{\text{E}}, \quad \overline{\boldsymbol{ac}}_{\text{C}} = \frac{1}{18}\sum_{k=1}^{18} \boldsymbol{ac}_{\text{C}}$$

The same four region (NH, SH, EB, EU) are chosen for the computation of the scores, and the results are shown in Table 2.4 - 2.5 and 2.6. Although the multi-model does not always outperform the other models, it's scores are never lower than the others either. Again, a rough division between the first two weks and the second ones is evident. The most evident difference between the two periods is in the variation of the score between the weeks: while initally the anomaly correlation decreases sharply, the variation between the third and the fourth week is less marked and occasionally remains constant. Often $\overline{\boldsymbol{ac}}_{\text{E}}$ is closer to $\overline{\boldsymbol{ac}}_{\text{MM}}$ than $\overline{\boldsymbol{ac}}_{\text{C}}$, expecially in the first week. Finally, another interesting comparison is the one between the ECMWF model and GLOBO. As in the analysis concerning $\boldsymbol{rmse}$ vectors, the gap between their performance is more evident for the first two weeks, where the $\overline{\boldsymbol{ac}}_{\text{C}}$ is systematically lower than $\overline{\boldsymbol{ac}}_{\text{E}}$. This characteristic is not present for the third and fourth week.

w	$\overline{\boldsymbol{ac}}_{\text{MM}}$	$\overline{\boldsymbol{ac}}_{\text{E}}$	$\overline{\boldsymbol{ac}}_{\text{C}}$
1	0.97	0.97	0.94
2	0.65	0.63	0.55
3	0.35	0.32	0.28
4	0.23	0.21	0.18

(a) Northern Hemisphere

w	$\overline{\boldsymbol{ac}}_{\text{MM}}$	$\overline{\boldsymbol{ac}}_{\text{E}}$	$\overline{\boldsymbol{ac}}_{\text{C}}$
1	0.95	0.95	0.92
2	0.58	0.56	0.47
3	0.31	0.27	0.25
4	0.25	0.21	0.21

(b) Southern Hemishpere

w	$\overline{\boldsymbol{ac}}_{\text{MM}}$	$\overline{\boldsymbol{ac}}_{\text{E}}$	$\overline{\boldsymbol{ac}}_{\text{C}}$
1	0.95	0.95	0.92
2	0.75	0.74	0.65
3	0.56	0.54	0.47
4	0.48	0.45	0.40

(c) Equatorial Belt

w	$\overline{\boldsymbol{ac}}_{\text{MM}}$	$\overline{\boldsymbol{ac}}_{\text{E}}$	$\overline{\boldsymbol{ac}}_{\text{C}}$
1	0.96	0.96	0.93
2	0.55	0.53	0.47
3	0.23	0.23	0.17
4	0.09	0.09	0.09

(d) Europe

Table 2.4: Anomaly correlation for the geopotential height anomalies at 500 hPa ($Z500$), averaged over the 18 validation winters. The four tables present the spatial average over the four different regions defined in the text. The four rows show the values for the different weeks, as specified by the first column, in blue. Each of the remaining columns refer to a different model: in the first one the are the values for the multi-model ($\overline{\boldsymbol{ac}}_{\text{MM}}$), in the second the ECMWF-IFS ones ($\overline{\boldsymbol{ac}}_{\text{E}}$) and in the third the CNR-ISAC ones ($\overline{\boldsymbol{ac}}_{\text{C}}$). The value corresponding to the best performances for each row is highlighted in red.

w	$\overline{\boldsymbol{ac}}_{\mathrm{MM}}$	$\overline{\boldsymbol{ac}}_{\mathrm{E}}$	$\overline{\boldsymbol{ac}}_{\mathrm{C}}$
1	0.95	0.95	0.91
2	0.65	0.63	0.56
3	0.40	0.35	0.35
4	0.32	0.27	0.27

(a) Northern Hemisphere

w	$\overline{\boldsymbol{ac}}_{\mathrm{MM}}$	$\overline{\boldsymbol{ac}}_{\mathrm{E}}$	$\overline{\boldsymbol{ac}}_{\mathrm{C}}$
1	0.92	0.92	0.87
2	0.63	0.60	0.53
3	0.49	0.41	0.42
4	0.49	0.40	0.42

(b) Southern Hemishpere

w	$\overline{\boldsymbol{ac}}_{\mathrm{MM}}$	$\overline{\boldsymbol{ac}}_{\mathrm{E}}$	$\overline{\boldsymbol{ac}}_{\mathrm{C}}$
1	0.90	0.89	0.78
2	0.68	0.67	0.54
3	0.55	0.51	0.45
4	0.53	0.48	0.46

(c) Equatorial Belt

w	$\overline{\boldsymbol{ac}}_{\mathrm{MM}}$	$\overline{\boldsymbol{ac}}_{\mathrm{E}}$	$\overline{\boldsymbol{ac}}_{\mathrm{C}}$
1	0.95	0.94	0.91
2	0.56	0.54	0.47
3	0.26	0.23	0.22
4	0.19	0.16	0.17

(d) Europe

Table 2.5: As in Table 2.4, but for the temperature at 850 hPa ($T850$).

w	$\overline{\boldsymbol{ac}}_{\mathrm{MM}}$	$\overline{\boldsymbol{ac}}_{\mathrm{E}}$	$\overline{\boldsymbol{ac}}_{\mathrm{C}}$
1	0.93	0.92	0.85
2	0.68	0.65	0.56
3	0.48	0.44	0.40
4	0.44	0.40	0.36

(a) Northern Hemisphere

w	$\overline{\boldsymbol{ac}}_{\mathrm{MM}}$	$\overline{\boldsymbol{ac}}_{\mathrm{E}}$	$\overline{\boldsymbol{ac}}_{\mathrm{C}}$
1	0.93	0.92	0.86
2	0.79	0.77	0.69
3	0.75	0.70	0.68
4	0.75	0.70	0.69

(b) Southern Hemishpere

w	$\overline{\boldsymbol{ac}}_{\mathrm{MM}}$	$\overline{\boldsymbol{ac}}_{\mathrm{E}}$	$\overline{\boldsymbol{ac}}_{\mathrm{C}}$
1	0.90	0.88	0.85
2	0.77	0.73	0.69
3	0.69	0.64	0.62
4	0.67	0.61	0.62

(c) Equatorial Belt

w	$\overline{\boldsymbol{ac}}_{\mathrm{MM}}$	$\overline{\boldsymbol{ac}}_{\mathrm{E}}$	$\overline{\boldsymbol{ac}}_{\mathrm{C}}$
1	0.93	0.93	0.85
2	0.62	0.60	0.52
3	0.38	0.34	0.34
4	0.37	0.33	0.32

(d) Europe

Table 2.6: As in Table 2.4, but for the temperature at 2 metre ($T2\mathrm{M}$).

Chapter 3

Probabilistic Forecasts

Statistical methods are often used on long-range weather forecasts: the predicatbility of the atmosphere, in a deterministic sense, decrease with the forecast time and statistics is useful, and sometimes necessary, when dealing with such systems.

Before dynamical forecasts informations were widely available, some weather forecasts (on timescales ranging from a day to a week) used to be produced purely by statistical means, without information on the underlying dynamics. This kind of application is known as "classical statistical forecasting". However, with the improvement in dynamical models, these methods became outdated and nowadays they are limited only to some specific time ranges (the extremely short or long ones). [Wilks, 2011].

In this thesis, we use statistics to analyze and postprocess the product of dynamical models. Naturally, statistics is already present in the ensemble forecast used in the previous chapter, but the final values that we found were deterministc fields, containing a precise value for each grid point and forecast week. In this chapter, we seek a different kind of output: we test different methods for predicting probablities. Scpecifically, the probability that the anomaly falls in one of three categories, the lower, middle and upper tercile of the distribution of the ERA-Interim reanalysis (the same used in the previous chapter) is computed.

The results are therefore verified using probabilistic scores such as the ranked probability skill score (RPSS) and, in the end, reliability diagrams. Again, the verification is performed using ERA-Interim reanalyses. The multi-model results are also compared to the probabilities obtained from the CNR-ISAC and the ECMWF-IFS ensembles.

Many different approach exists for dealing with this task. We, however, test only three different classes of algorithms:

- the *Direct Model Output* (DMO) which provides the "reference values". Its elements are the *Democratic Voting* (DV) method and the *Tukey Plotting Position* (TPP);

- the *Logistic Regression* (LR), where different choices of predictors are tested,

- the *Nonhomogeneous Gaussian Regression* (NGR).

These methods can be found in literature applied to similar task, like in Hamill et al. [2004], Whitaker et al. [2006], Wilks [2006] and Wilks and Hamill [2007]. Naturally this list is not exhaustive and more sets of articles will be cited when discussing each algorithm. Note that the scientific literature provides a wider range of methods that can be applied to forecasting probabilities, many of them based not on regressions but on totally different approaches, like kernel dressing methods [Wilks, 2011]. Although their usage could have lead to interesting results, we constrained the analysis to a limited set of techniques in order to better analyze the possible variants, therefore increasing the chances of extracting the best performances from each method.

While this brief introduction acts as an introduction to all the second section of the thesis, in this chapter we focus solely on the DMO techniques. A complete explanation of the remaining algorithms is presented in the following chapters, alongside the exposition of the reasons behind their choice. The dataset used throughout this chapter is the same of the previous one, although the analysis is performed only for one variable (T2M). Also the notation remains the same: $\boldsymbol{X}_{\mathrm{MM}}$, $\boldsymbol{X}_{\mathrm{C}}$ and $\boldsymbol{X}_{\mathrm{E}}$ are the anomaly fields for the multi-model and the two ensemble means, while $\boldsymbol{Y}$ is the anomaly with respect to the corresponding reanalyses.

3.1 Binary Verification Data

Before proceeding with the analysis, we compute a fundamental quantity for DMO as well as LR and NGR: the "binary verification data". This term refers to a tensor, derived from the reanalises, used for training and verification. As the name suggests, its elements are either one or zero, depending on $\boldsymbol{Y}$ and the terziles of its distribution over the training period.

For clarity purpose, in the following explanation one validation winter is chosen (thus also the training set is fixed). Obviously, in the actual analysis this set of operations is repeated for all the 18 possible combination.

The first operation is the computation of the terciles from the training set. For every week w and grid point (i, j), the two terciles are extracted from the slice $\boldsymbol{Y}(w, i, j, :)$[1]. The results of this operation are two tensors, $\boldsymbol{Y}_{1/3}(w, i, j)$ and $\boldsymbol{Y}_{2/3}(w, i, j)$, containing respectively the lower and the upper terciles.

[1] This operation is performed using the function *percentile*, from the *numpy* module (Python 2.7.9). More detailed information can be found on the package website, in the section dedicated to this function:
http://docs.scipy.org/doc/numpy-dev/reference/generated/numpy.percentile.html.
Note that when the tercile (q) lies between two values (v and w), the optional parameter *interpolation* of *percentile* is set to *linear*. This means that q is given by $v + (w - v) * f$, where f is the fractional part of the index surrounded by v and w.

Then, the tensor containing the binary verification data is given by:

$$\boldsymbol{B}(w,i,j,d,0) = \begin{cases} 0 & \text{if } \boldsymbol{Y}(w,i,j,d) > \mathbf{Y}_{1/3}(w,i,j) \\ 1 & \text{if } \boldsymbol{Y}(w,i,j,d) < \mathbf{Y}_{1/3}(w,i,j) \end{cases}$$

$$\boldsymbol{B}(w,i,j,d,1) = \begin{cases} 0 & \text{if } \boldsymbol{Y}(w,i,j,d) < \mathbf{Y}_{1/3}(w,i,j) \text{ or } \boldsymbol{Y}(w,i,j,d) > \mathbf{Y}_{2/3}(w,i,j) \\ 1 & \text{if } \mathbf{Y}_{2/3}(w,i,j) \leq \boldsymbol{Y}(w,i,j,d) \leq \mathbf{Y}_{2/3}(w,i,j) \end{cases}$$

$$\boldsymbol{B}(w,i,j,d,2) = \begin{cases} 0 & \text{if } \boldsymbol{Y}(w,i,j,d) < \mathbf{Y}_{2/3}(w,i,j) \\ 1 & \text{if } \boldsymbol{Y}(w,i,j,d) > \mathbf{Y}_{2/3}(w,i,j). \end{cases}$$

Note that, while the terciles are computed only in the training set, there are two diferent $\boldsymbol{B}$: one for the training set, with $d = 1, ..., m$, and the second for the validation one and with $d = 1, ..., m_{\mathrm{V}}$.[2]

It can be useful to explain the meaning of the last entry in $\boldsymbol{B}$. Its index can assume the values 0, 1 and 2 and the corresponding entry follows these rules:

- $\boldsymbol{B}(w,i,j,d,0)$ contains information about which points are below the firts tercile. It's value is 1 when the reanalysis is below that threshold and 0 otherwise;

- $\boldsymbol{B}(w,i,j,d,1)$ refers to the values between the two terciles. It is equal to 1 when the reanalysis lies in that interval and 0 otherwise;

- finally, in $\boldsymbol{B}(w,i,j,d,2)$ the last region is highlighted, with the tensor entry equal to 1 when the corrsponding value in $\boldsymbol{Y}$ is greater than the upper tercile and 0 otherwise.

3.2 Direct Model Output: Methodology

The first set of algorithms answers the question: "what are the simplest and most direct way of computing probabilities from the ensembles or their linear combination?"

The methods presented in this section are not MOS techniques [Wilks, 2006]. Their basic assumption is that the ensamble behave as a random sample from the real cumulative distribution function (CDF), and then the cumulative probabilities are estimated using a plotting position [Wilks, 2011]. As mentioned before, these methods represent the baseline with which the other algorithms will be compared. Because DMO is among the simplest techniques for extracting probabilities from the forecast field, improvements are expected in using more complex ones like LR and NGR. So, by measuring these improvements, an estimate of the advantage of introducing such elaborate algorithms is obtained.

3.2.1 Implementation of the reduced multi-model ensembles

Before starting with the description of the algorithms, there are some operations concerning the multi-model that are fundamental for their applicability. All the

[2] m and m_{V} are the dimensions of the two sets, like in the previous chapter.

DMO methods need more than one ensemble member to be employed, because all of them rely on the position of the quantile relative to those members.

For the initial ensembles, the ECMWF-IFS and the CNR-ISAC ones, there are five members, so the algorithms ca be used directly. However, in the previous chapter, only the multi-model ensemble mean has been computed. In order to apply DMO methods also to the multi-model, its components need to be computed. This can be achived through a linear combination of the initial ensembles, using a smaller subset of members.

Before proceeding with the analysis, it can be useful to introduce some notation. Again, tensors have been chosen to act as "containers" for all the data:

- $\boldsymbol{E}_{\mathrm{E}}(w,i,j,d,l)$ contains the anomaly fields for all the members of the ECMWF-IFS ensemble;
- $\boldsymbol{E}_{\mathrm{C}}(w,i,j,d,l)$ contains the anomaly fields for all the members of the CNR-ISAC one;
- $\boldsymbol{E}_{\mathrm{MM}}(w,i,j,d,l)$ is the tensor that will be computed in this section and that contains the anomalies for all the multi-model members.

The index l is added to range between the ensemble members, with $l=0$ corresponding to the control one (or their linear combinarion, for $\boldsymbol{E}_{\mathrm{MM}}$), and the other values for the perturbed ones (or a combination containing at least one of them).

So, using $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ from the previous chapter, the multi-model members are computed using:

$$\boldsymbol{E}_{\mathrm{MM}}(w,i,j,d,l_3)=\boldsymbol{C}_1(w,i,j)\boldsymbol{E}_{\mathrm{E}}(w,i,j,d,l_1)+\boldsymbol{C}_2(w,i,j)\boldsymbol{E}_{\mathrm{C}}(w,i,j,d,l_2).$$

with $l_1=0,...,4$, $l_2=0,...,4$ and $l_3=l_1+5l_2=0,...,24$. The multi-model ensamble size is greater than the size of the original ones, due to the way in which it is obtained. The procedure is schematized in Figure 3.1.

Naturally, it can be shown that average of $\boldsymbol{E}_{\mathrm{MM}}$ along the last dimension (l) gives exactly $\boldsymbol{X}_{\mathrm{MM}}$[3]. This is fundamental for the consistency of the analysis: LR and NGR need as one of the predictors the ensemble mean, and the values presented in this section would not be coherent with the next algorithms if $\boldsymbol{X}_{\mathrm{MM}}$ was not the ensemble mean of $\boldsymbol{E}_{\mathrm{MM}}$.

3.2.2 Democratic Voting

The simplest of the two methods in the DMO class is the Democratic Voting. An example of its usage for predicting probabilities can be seen in the scientific litearure in Wilks [2006], in addition to beeing the first of the methods presented for producing such forecasts in [Wilks, 2011, Chapter 7].

Assigning the name $\boldsymbol{Q}(w,i,j,d)$ to a generic quantile (in this case the two candidates are $\boldsymbol{Y}_{1/3}$ and $\boldsymbol{Y}_{2/3}$), the probability is given by:

$$\begin{aligned}Pr\big(\boldsymbol{Y}(w,i,j,d)<\boldsymbol{Q}(w,i,j,d)\big)&=\frac{1}{l_{\mathbf{max}}}\sum_{l=0}^{l_{\max}}I\big(\boldsymbol{E}(w,i,j,d,l)\leq\boldsymbol{Q}(w,i,j,d)\big)\\&=\frac{rank\big(\boldsymbol{Q}(w,i,j,d)\big)-1}{l_{\max}},\end{aligned}$$

[3] **Dimostrazione...**

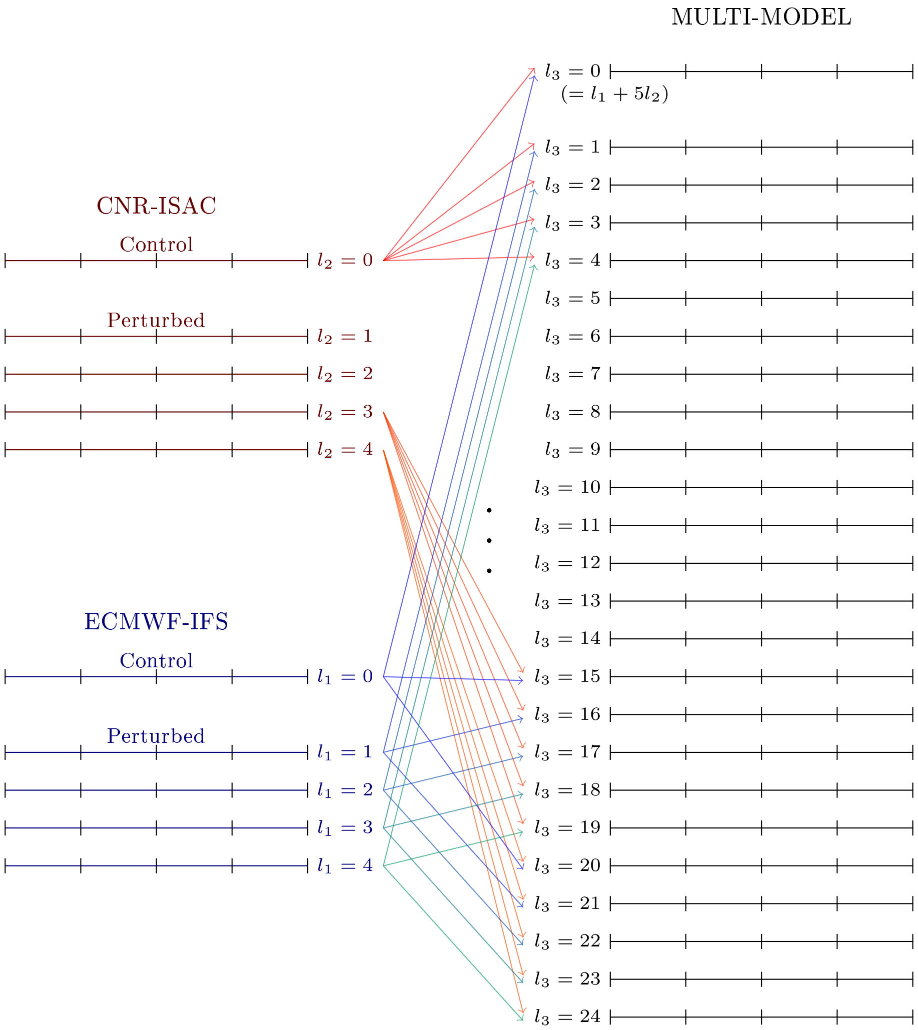


Figure 3.1: Schematised representation of the multi-model ensemble computation. On the left there are the CNR-ISAC ans ECMWF-IFS ensembles (weekly mean anomaly fields), respectively in dark red and blue. On the right there is the resulting multi-model ensemble (in black). The arrows connect the original members with the final ones, obtained from their linear combination. The red and blue arrows can be seen respectively as the multiplication for the coefficient $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$, and the product results are added together to compute the multi-model member indicated by the arrows head. Only some of the connections are shown, the black dots in the middle stand for the missing arrows. Finally, near each member there is its number inside the ensemble (l_1, l_2 and l_3).

where I is a function with a boolean domain: it returns 1 when its argument is true and 0 otherwise. The ensemble dimension is $l_{\max}$ and, from the definitions in the previous section, $l_{\max} = 5$ for $\boldsymbol{E}_{\mathrm{E}}$ and $\boldsymbol{E}_{\mathrm{C}}$ and $l_{\max} = 25$ for $\boldsymbol{E}_{\mathrm{MM}}$. Finally, $rank\big(\boldsymbol{Q}(w,i,j,d)\big)$ is the rank of $\boldsymbol{Q}(w,i,j,d)$ inside a set containing the quantile itself and all the ensemble members $\boldsymbol{E}(w,i,j,d,l=0,...,l_{\max})$.

From the formula it is evident that this estimator has a undesirable property: every quantile that has a value lower than all the ensemble member has probbability equal to zero, while if the quantile is greater than all the members, its probability is one [Wilks, 2006, p. 282]. Nevertheless, this algorithm is included in the analysis for its semplicity. In fact, it represent the optimal answer to the question posed at the beginning of the DMO section. This make the DV a good choiche for determine the comparison baseline for the other methods: in each of them will be more elaborate, and if the additional compelxity does not corrispond to an improvement in the performance, they are not worth the extra computations needed for their implementation.

Now that the basics of the algorithm is clear, it can be useful to introduce some notation regarding its prducts. It will be useful in the last part of the chapter, where all the methos will be compared. So, three probability tensors are introduced. The approach is similar to the one used for $\boldsymbol{B}$, but instead of creating a single tensors and using the last dimension to differentiate between the three terciles, each of them is assigned to a separate tensor:

- $\boldsymbol{P}^{\mathrm{DV}}_{\mathrm{inf}}(w,i,j,d)$ is the probability that the observation will be below the lower tercile (for a given week, grid point ad date). It is simply equal to:

$$Pr\big(\boldsymbol{Y}(w,i,j,d) < \boldsymbol{Y}_{1/3}(w,i,j,d)\big);$$

- $\boldsymbol{P}^{\mathrm{DV}}_{\mathrm{mid}}(w,i,j,d)$ is the probability for the region between the two terciles, and it is given by:

$$Pr\big(\boldsymbol{Y}(w,i,j,d) < \boldsymbol{Y}_{2/3}(w,i,j,d)\big) - Pr\big(\boldsymbol{Y}(w,i,j,d) < \boldsymbol{Y}_{1/3}(w,i,j,d)\big);$$

- $\boldsymbol{P}^{\mathrm{DV}}_{\mathrm{sup}}(w,i,j,d)$ is the probability that the observation is above the upper tercile and it is computed using:

$$1 - Pr\big(\boldsymbol{Y}(w,i,j,d) < \boldsymbol{Y}_{2/3}(w,i,j,d)\big).$$

Needless to say, all this quantity are computed for both the training set $(d = 1,...,m)$ and the validation set $(d = 1,...,m_{\mathrm{V}})$. Due to the nature of DMO techniques, there

Naturally, for each of the three models this triplet of tensors is computed. Therefore, the notation is modified one more time, introducing another superscript (on the left hand corner). The resulting names are:

- ${}^{\mathrm{MM}}\boldsymbol{P}^{\mathrm{DV}}_{\mathrm{inf}}$, ${}^{\mathrm{MM}}\boldsymbol{P}^{\mathrm{DV}}_{\mathrm{mid}}$ and ${}^{\mathrm{MM}}\boldsymbol{P}^{\mathrm{DV}}_{\mathrm{sup}}$ for the multi-model derived probabilities,
- ${}^{\mathrm{C}}\boldsymbol{P}^{\mathrm{DV}}_{\mathrm{inf}}$, ${}^{\mathrm{C}}\boldsymbol{P}^{\mathrm{DV}}_{\mathrm{mid}}$ and ${}^{\mathrm{C}}\boldsymbol{P}^{\mathrm{DV}}_{\mathrm{sup}}$ for the CNR-ISAC ones,
- ${}^{\mathrm{E}}\boldsymbol{P}^{\mathrm{DV}}_{\mathrm{inf}}$, ${}^{\mathrm{E}}\boldsymbol{P}^{\mathrm{DV}}_{\mathrm{mid}}$ and ${}^{\mathrm{E}}\boldsymbol{P}^{\mathrm{DV}}_{\mathrm{sup}}$ for the ECMWF-IFS ones.

3.2.3 Tukey Plotting Position

The second method is the Tukey Plotting Position. Like the previous technique, its application to forecasting probabilities can be see in Wilks [2006] and it is also one of the sugested methodd in [Wilks, 2011, Chapter 7]. It is similar to the previous one, as can easily be seen from its formula:

$$Pr\big(\boldsymbol{Y}(w,i,j,d) < \boldsymbol{Q}(w,i,j,d)\big) = \frac{rank\big(\boldsymbol{Q}(w,i,j,d)\big) - 1/3}{(l_{\max}+1) - 1/3}.$$

The choice of this algorithm is due to its lack of the problem affecting DV [Wilks, 2011, p. 282]: if a quantile is lower (greater) than all the ensemble members, its probability is not zero (one). Altough this is not the simplest possible algorithm, at least at a conceptual level, it is included in the analysis because it is as computational intensive as DV. This means that, with no added computing time, it can provide better performance than the other, remaining in the meantime almost of the same simplicity. In addition, except for DV, all the other algorithms tested will introduce some extra complexity, and this complexity needs to be justified by an improvement in the results respecto what can be achieved by a simple algorithm like TPP.

As in the previous case, some probability tensors are introduced:

$$\boldsymbol{P}^{\mathrm{TPP}}_{inf}(w,i,j,d), \qquad \boldsymbol{P}^{\mathrm{TPP}}_{mid}(w,i,j,d), \qquad \boldsymbol{P}^{\mathrm{TPP}}_{sup}(w,i,j,d),$$

computed again from the probability, using the same formulas as for DV. Once more, the calculation is performed on both the training and validation set and for each of the three models (${}^{\mathrm{MM}}\boldsymbol{P}^{\mathrm{TPP}}$, ${}^{\mathrm{C}}\boldsymbol{P}^{\mathrm{TPP}}$ and ${}^{\mathrm{E}}\boldsymbol{P}^{\mathrm{TPP}}$).

3.3 Direct Model Output: Validation

Using the two methods just described, we compute the tercile probabilities on all the winters in which the dataset has been split[4] . Both DM and TPP are not based on regression, so there is no need of a training set. However, we retain the cross validation approach with the same validation winters in order to make comparison with the next algorithms. So, after computing probabilities, these are used for evaluate the preformances of the methos using some probabilistic score.

Unlike the previous chapter, in this case there are no coefficient maps to show before proceeding with the validation. Also showing directly the results is impossible, due to the large number of entries in the dataset. Nevertheless, at the end of the section we show an example output. Some interesting features can be seen, and it will serve as a comparison term for the following chapter.

3.3.1 Probabilistic Scores

The fist of the probabilistic scores we use in the thesis is the Ranked Probability Score (RPS), a common measure used for evaluating probability forecasts for Multiple-category events. It is simply an extension of the Brier Score to a

[4] Additional information on the division of the dataset in validation winters can be found in the previous chapter, where the cross-validation approach was introduced.

situation in which multiple outcomes are possible[5]: it measures the squared error respect to the verifying reanalysis, using the cumulative probabilities (unlike the Brier Score) [Wilks, 2011]. Due to the simplicity of its computation and its beeing sensitive to distance, it is widely used in the scientific literature when dealing with probabilistic forecast. Examples of its application are countless, a small and surely not exhaustive list includes Hamill et al. [2004], Wilks [2006] and Wilks and Hamill [2007], other than beeing cited as one of the scores in [Wilks, 2011, Chapter 8].

Using the binary verification tensors previously defined we first define the cumulative proabbilities as:

$$\begin{aligned}
&\boldsymbol{B}_{\text{cml}}(w,i,j,d,0) = \boldsymbol{B}(w,i,j,d,0)\\
&\boldsymbol{B}_{\text{cml}}(w,i,j,d,1) = \boldsymbol{B}(w,i,j,d,0) + \boldsymbol{B}(w,i,j,d,1)\\
&\boldsymbol{B}_{\text{cml}}(w,i,j,d,2) = \boldsymbol{B}(w,i,j,d,0) + \boldsymbol{B}(w,i,j,d,1) + \boldsymbol{B}(w,i,j,d,2)
\end{aligned}$$

Then, we compute the same quantitym but for the probabilities obtained from the two algorithm. For the moment, we discard the superscript, and we refer gerenically with $\boldsymbol{P}_{inf}$, $\boldsymbol{P}_{mid}$ and $\boldsymbol{P}_{sup}$ to the prediction for each of the three sectors of the distribution. The cumulative prediction are giveb by:

$$\begin{aligned}
&\boldsymbol{P}_{\text{cml}}(w,i,j,d,0) = \boldsymbol{P}_{\text{inf}}(w,i,j,d)\\
&\boldsymbol{P}_{\text{cml}}(w,i,j,d,1) = \boldsymbol{P}_{\text{inf}}(w,i,j,d) + \boldsymbol{P}_{\text{mid}}(w,i,j,d)\\
&\boldsymbol{P}_{\text{cml}}(w,i,j,d,2) = \boldsymbol{P}_{\text{inf}}(w,i,j,d) + \boldsymbol{P}_{\text{mid}}(w,i,j,d) + \boldsymbol{P}_{\text{sup}}(w,i,j,d)
\end{aligned}$$

From them we derive the Ranked Probability Score:

$$\boldsymbol{RPS}(w,i,j,d) = \sum_{c=1}^{3} \big(\boldsymbol{B}_{\text{cml}}(w,i,j,d,c)\boldsymbol{P}_{\text{cml}}(w,i,j,d,c)\big)^2.$$

As can be seen from the formula, it is simly the sum over the categories c in which the distribution has been split (in our case 3) of the squared difference between the corresponding cumulative probabilites of the forecast and of the verifying verification tensor. Like with the Brier Score, the less accurate a forecast, the higher the score, with the perfect prediction having $\boldsymbol{RPS} = 0$. Note also that $\boldsymbol{B}_{\text{cml}}(w,i,j,d,2)$ and $\boldsymbol{P}_{\text{cml}}(w,i,j,d,2)$ are the sum over all the probability categories and therefore both equal to one. This impy that the maximum value that RPS can assume is equal to the number of categories minus one, that is 2 in our case.

Naturally, $\boldsymbol{RPS}(w,i,j,d)$ contains the score for each of the date (of the validation set, in our case). So, we perform an average on the set:

$$\overline{\boldsymbol{RPS}}(w,i,j) = \frac{1}{m_V}\sum_{d=1}^{m_V} \boldsymbol{RPS}(w,i,j,d).$$

[5] Brier score (BS) is a scalar measure for the accuracy of probabilistic forecast in which only two outomes are possible. If we assign the name o_k to the verifying observation (with the possible outcomes beeing $o_k = 0$ and $o_k = 1$) and the name y_k to the prediction, the Brier Score is given by:

$$BS = \frac{1}{n}\sum_{k=1}^{n}(y_k - o_k)^2,$$

where n represent the number of examples in the dataset. Bs is negatively oriented, the best possible values obtainable is 0 and higher values corespond to less accurate forecasts. It can assume only values between and including 0 and 1. [Wilks, 2011]

The second score we computed is directly linked to RPS: it is the Ranked Probability Skill Score. As the name suggest, it compares the skill of our forecast to some reference values. These values are the RPS computed from the climatology. Like the previous one, also this score is widely used trought the literature (the example list is the same as above), and a complete description can be foud in [Wilks, 2006, Chapter 8].

We first define the climaytological probability for the T2M anomaly fallin in each of the three section, that from the definition of terciles is simply:

$$\begin{aligned}
\boldsymbol{P}^{(\mathrm{clim})}_{\mathrm{inf}}(w,i,j,d) &= 1/3,\\
\boldsymbol{P}^{(\mathrm{clim})}_{\mathrm{mid}}(w,i,j,d) &= 1/3,\\
\boldsymbol{P}^{(\mathrm{clim})}_{\mathrm{sup}}(w,i,j,d) &= 1/3.
\end{aligned}$$

These quantities are used for computing $\overline{\boldsymbol{RPS}}^{(\mathrm{clim})}(w,i,j)$, that is simply the average of the ranked probability score over the validation set, computed using $\boldsymbol{P}^{(\mathrm{clim})}_{\mathrm{inf}}$, $\boldsymbol{P}^{(\mathrm{clim})}_{\mathrm{mid}}$ and $\boldsymbol{P}^{(\mathrm{clim})}_{\mathrm{sup}}$ as predictions. Then, the skill score is defined by:

$$\boldsymbol{RPSS}(w,i,j) = 1 - \frac{\overline{\boldsymbol{RPS}}(w,i,j)}{\overline{\boldsymbol{RPS}}^{(\mathrm{clim})}(w,i,j)}. \tag{3.1}$$

Naturally, the values represent already the average over the validation set, because its computed starting from $\overline{\boldsymbol{RPS}}$ and $\overline{\boldsymbol{RPS}}^{(\mathrm{clim})}$.ù As can be easily deduced from the formula, a perfect forecast would have $\overline{\boldsymbol{RPS}}(w,i,j) = 0$ and, if $\overline{\boldsymbol{RPS}}^{(\mathrm{clim})}(w,i,j) \neq 0$, the skill score would be $\boldsymbol{RPSS}(w,i,j) = 1$. A lower value indicates that a less skillfull forecast.

The last score is a modification of RPSS: it is the Discrete Ranked Probability Skill Score (DRPSS). As underlined in Müller et al. [2005], Weigel et al. [2007a], Weigel et al. [2007b] and Weigel et al. [2008], RPSS is biased negatively for small ensembles. For removing this bias, the climatological term in Equation 3.1 need to be substituted with the expectation of $\boldsymbol{RPS}^{(\mathrm{ran})}$ averaged over the validation winters, where $RPS^{(\mathrm{ran})}$ represent the ranked probability score produced by repeatedly resampling from the climatology a number of samples equal to the ensemble size. In Weigel et al. [2007a] however, we found an aletrantive formulation. The term D was introduced, it represent the difference between $\overline{\boldsymbol{RPS}}^{(\mathrm{clim})}$ and teh averaged expectation value of $\boldsymbol{RPS}^{(\mathrm{ran})}$. For our case, where the three categories are all equiprobable, this term can be modelized and it is given by:

$$D = \frac{1}{l_{\max}} \frac{(c_{\max})^2 - 1}{6\, c_{\max}} = \frac{4}{9\, l_{\max}},$$

where $c_{\max}$ is the number of categories in which the probability distribution is divided (in our case, $c_{\max} = 3$) and $l_{\max}$ represent the ensemble size, as in the previous sections. Note that, an increas in $l_{\max}$ result in a smaller correction term D, therefore $\boldsymbol{DRPSS}$ converges toward $\boldsymbol{RPSS}$ for extremely high values of $l_{\max}$. On the contrary, increasing the number of categories leads to higher values of D. Thus, the score is computed from the formula:

$$\boldsymbol{DRPSS}(w,i,j) = 1 - \frac{\overline{\boldsymbol{RPS}}(w,i,j)}{\overline{\boldsymbol{RPS}}^{(\mathrm{clim})}(w,i,j) + D}.$$

The reason behind the introduction of this additional score is in the dimension of the CNR-ISAC and the ECMWF-IFS ensemble used in this analysis. Both are composed of only 5 members and we wanted a score less sensitive to the size for a fairer comparison between the multi-model and the two single models. Anyway, we keep all three scores, for a more complete overview of the performances.

3.3.2 Reliability Diagram

Another powerful tool used for evaluating performances is the Reliabilty Diagram. Like the score presented in the previous sections, its usage is common in the scientific literature and examples of its application can be seen in Hamill et al. [2004], Wilks and Hamill [2007] and Wilks [2009]. A complete explanation of the procedure is also presented in [Wilks, 2011, Chapter 8].

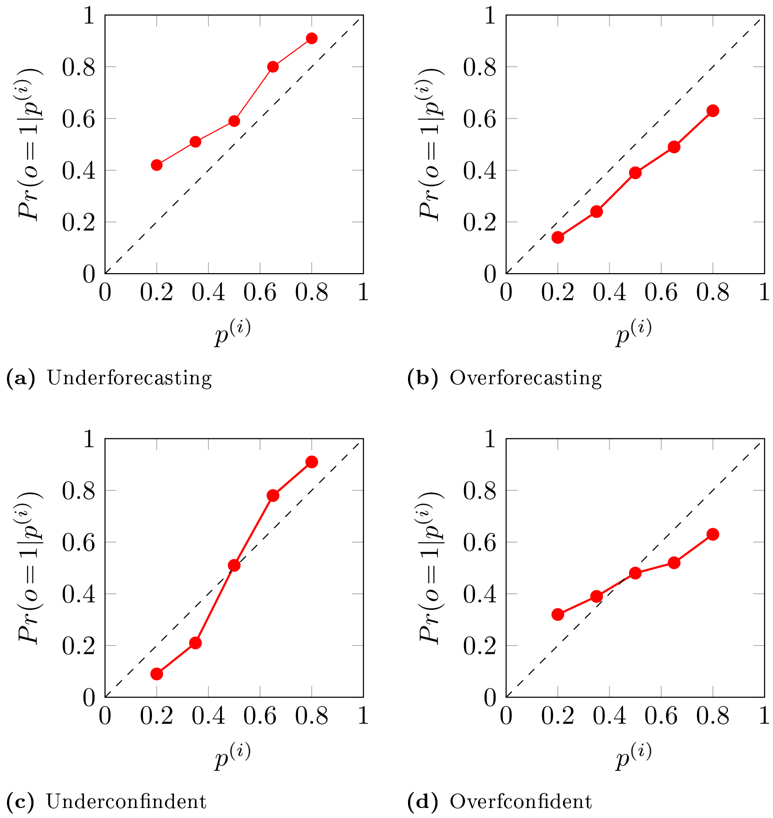
Unlike the scores previously presented, which provided a summary of the performances through a single value, the reliability diagram shows the full joint distribution of the forecast and the verifying reanalyses. It refers to a binary predictand, so different diagrams have to be created for the two quantiles.

The structure of such plots can be divided in two main parts, the first of which is the calibration function. In Figure 3.2 we show some hypotetical examples, with four curves corresponding to different problems that can affect the forecasts.

The first step in making such plots is the division of the possible outcomes of our forecast (considering the two terciles threshold separately) in I probability intervals. Each category corresponds to possible output value, that we will generally call $p^{(i)}$. From the binary verifying oservation o, we compute the probability of $o = 1$ given the forecast outcome $p^{(i)}$. They consitute the conditional probabilities $Pr(o = 1|p^{(i)})$, and plotting all the values $i = 1, ..., I$ we finally obtain the calibration curve. As can be seen in Figure 3.2, this curves allow an immediate visualization of some kind of errors. We start by noticing that a dashed line connect the lower-left corner to the upper-right one. An hypotetical calibrtaion curve lying on this line would represent the best case scenario: the output probabilities are equal to the frequency of o beeing equal to one given $p^{(i)}$. Dispacements from this ideal cases can result in various kinds of biases. In **(a)** we see an example of undeforecasting, that is the forecast probabilities beeing regularly lower than the frequencies $Pr(o|p^{(i)})$. On the other hand, **(b)** shows the opposit problem, called overforecasting and consisting in the probabilities being higher than the frequencies. These kind of problems are often referred as unconditional biases.

A different kind of biases are the conditional ones, in a certain sense they are the ones in which the bias depends on the forecast itself. They are shown in the bottom panels of Figure 3.2. In **(c)** we see an example of an underconfident or poor resolution forecast. In this situation, the frequencies depends only weakly on the forecast, and they are closer to the climatological distribution. Panel **(d)** shows the opposit situation, called overconfidence or good resolution, consisting in $Pr(o|p^{(i)})$ depending strongly on $p^{(i)}$.

The second part of the reliability diagram is the refinement distribution. It shows the frequency $Pr(p^{(i)})$ with which each of the I categories appears in the forecasts. Again, we use some example plot in Figure 3.3 for describing how the curves can be interpreted.



(a) Underforecasting

(b) Overforecasting

(c) Underconfindent

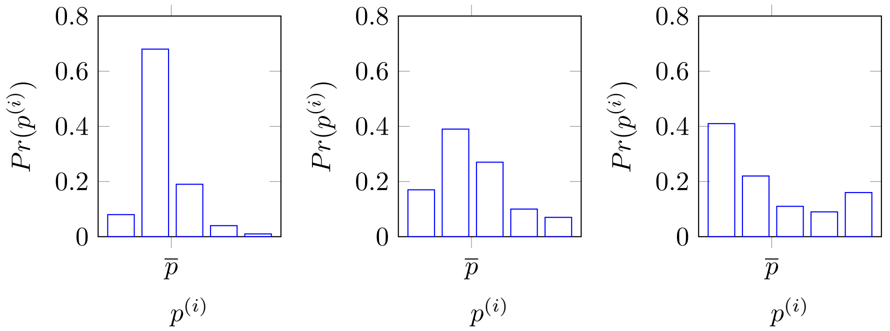
(d) Overfconfident

Figure 3.2: Four examples of calibration functions. Each of them represents an hypothetical forecast affected by the problem described below the plot. With the symbol $p^{(i)}$ we denote the forecast probability, while $Pr(o\!=\!1|p^{(i)})$ represents the conditional probabilities of the veryfing observation (or reanalysis) for each value of the forecast. Together, the conditional probabilities consititute the calibration function, shown in red. Note that underconfident and overconfident forecasts are sometimes referred as good relolution and poor resolution, respectively. The values shown in red are chosen manually for giving the correct shape for each cure. The plot is inspired by an analogous one in [Wilks, 2011, p. 335].

These kind of plot are used for evaluating the confidence of the algorithm. If the values are ralely far from the mean value, as in panel **(a)**, then the forecast is underconfident. The opposit case can be seen in **(c)**, where the forecast is overconfident and often outputs extreme values. An intermediate confidence situation is shown in panel **(b)**.

In order to apply this verificantion technique to our result, we need to adapt the notation. Naturally, we start from the binary verification tensors $\boldsymbol{B}$ and the output probabilities $\boldsymbol{P}_{\text{inf}}$ and $\boldsymbol{P}_{\text{sup}}$. We first defined the tresholds for the output probabilities. Altought the reliability diagrams for the single models are not shown in this analiysis, we decided to keep the verification fully compatible with them and so we contrained the number of probability categories to five.

Obviously, the choice of this number need to be discussed. A DMO technique



(a) Low confidence **(b)** Intermediate confidence **(c)** High confidence

Figure 3.3: Three examples of refinement distribution. With the letter y we denote again the forecast probability, while $Pr(p^{(i)})$ represent the frequency with which each of the categories $p^{(i)}$ appears. On the x-axis, $\overline{p}$ represent the mean value of the probability forecasts. The values shown in red are chosen manually for giving the correct shape for each curve. Also this plot is inspired by an analogou one in [Wilks, 2011, p. 335].

applied to 5-memmber ensembles (such as the ECMWF-IFS and the CNR-ISAC re-forecast used), can output only a set of 6 values, which is the number of possible position of the tercile respect to the members. However, dividing the interval between 0 and 1 (the domain of the calibration function) in six parts gives a lenght of $0.1\overline{6}$, that is not very comfortable to use, so we reduced the number to five. The resulting probability intervals $p^{(c)}$ are: [0.0, 0.2], (0.2, 0.4], (0.4, 0.6], (0.6, 0.8] and (0.8, 1.0], note that only the first is closed on both sides. DMO techniques applied to a 25 memeber ensemble like the multi-mdoel one, produces 26 probability categories. So, four of the previous intervals contains 5 of these values while the remaining one contains six of them. For example, when using DV, six of the possible outcomes are inside the interval [0.0, 0.2], due to DV assigning probability 0 to a forecast entirely above (below) the lower (upper) tercile. However, this problem does not arise in the folloeing chapter, beacuse the regression as a contiuous output.

For simplyfing the notation, we refer to each of the intervals using its central value, so we define:

$$\overline{p}^{(c)} = 0.1 + 0.2(c-1),$$

with $c = 1, ..., 5$ beeing the category index.

So, once the intervals have been defined, we counted how many elements from $\boldsymbol{P}_{\text{inf}}$ and $\boldsymbol{P}_{\text{sup}}$ fall in each of them. Dividing this quantity by the number of dates of the validation set gives us the frequancy of each of the forecast categories:

$$\boldsymbol{Pr}_{\text{inf}}(p^{(c)})(w,i,j) = \frac{1}{m_V}\sum_{d=1}^{m_V} \boldsymbol{CTRP}_{inf}^{(c)}(w,i,j,d),$$

$$\boldsymbol{Pr}_{\text{sup}}(p^{(c)})(w,i,j) = \frac{1}{m_V}\sum_{d=1}^{m_V} \boldsymbol{CTRP}_{sup}^{(c)}(w,i,j,d).$$

The quantity inside the sum acts as a counter for each category and tercile, and it is defined by:

$$\boldsymbol{CTRP}^{(c)}_{inf}(w,i,j,d) = \begin{cases} 1 & \text{if } \boldsymbol{P}_{\text{inf}}(w,i,j,d) \in p^{(c)} \\ 0 & \text{otehrwise,} \end{cases}$$

$$\boldsymbol{CTRP}^{(c)}_{sup}(w,i,j,d) = \begin{cases} 1 & \text{if } \boldsymbol{P}_{\text{sup}}(w,i,j,d) \in p^{(c)} \\ 0 & \text{otehrwise.} \end{cases}$$

The conditional probabilities are obtained from the tensors $\boldsymbol{B}(w,i,j,d,0)$ and $\boldsymbol{B}(w,i,j,d,2)$ counting how many times their elements are equal to one when the corresponding prediction falls in each interval and dividing the results by the total number of times $\boldsymbol{P}_{\text{inf}}$ and $\boldsymbol{P}_{\text{sup}}$ are inside that interval. The formula used for their computation is:

$$\boldsymbol{Pr}_{\text{inf}}(o|p^{(c)})(w,i,j) = \frac{1}{\sum d = 1^{m_v} \boldsymbol{CTRP}(w,i,j,d)} \sum_{d=1}^{m_V} \boldsymbol{CTRB}^{(c)}_{inf}(w,i,j,d),$$

$$\boldsymbol{Pr}_{\text{sup}}(o|p^{(c)})(w,i,j) = \frac{1}{\sum d = 1^{m_v} \boldsymbol{CTRP}(w,i,j,d)} \sum_{d=1}^{m_V} \boldsymbol{CTRB}^{(c)}_{sup}(w,i,j,d).$$

This times there are two counters. The one inside the sum at the denominator is the same as in the previous formula, while the other is given by:

$$\boldsymbol{CTRB}^{(c)}_{inf}(w,i,j,d) = \begin{cases} 1 & \text{if } \boldsymbol{P}_{\text{inf}}(w,i,j,d) \in p^{(c)} \text{ and } \boldsymbol{B}(w,i,j,d,0) = 1 \\ 0 & \text{otehrwise,} \end{cases}$$

$$\boldsymbol{CTRB}^{(c)}_{sup}(w,i,j,d) = \begin{cases} 1 & \text{if } \boldsymbol{P}_{\text{sup}}(w,i,j,d) \in p^{(c)} \text{ and } \boldsymbol{B}(w,i,j,d,2) = 1 \\ 0 & \text{otehrwise.} \end{cases}$$

Note that the quantities described above are computed for each of the validation sets. So we first perform an average on all the 18 winters, the resulting values are $\overline{\boldsymbol{Pr}}_{\text{inf}}(p^{(c)})(w,i,j)$, $\overline{\boldsymbol{Pr}}_{\text{sup}}(p^{(c)})(w,i,j)$, $\overline{\boldsymbol{Pr}}_{\text{inf}}(o|p^{(c)})(w,i,j)$ and $\overline{\boldsymbol{Pr}}_{\text{sup}}(o|p^{(c)})(w,i,j)$.

3.3.3 Spatial averages

As in the previous chapter, all the scores are computed on a lat-lon grid and the spatial average has to be performed using some weights.

We reintroduce the weights matrix:

$$\boldsymbol{W}(i,j) = \cos\big(\phi(i)\big),$$

from the previous chapter. For each of the scores we perform the spatial average on five spatial regions: Northern Hemisphere (NH), Southern Hemisphere (SH), Equatorial Belt (EB) and Europe (EU)[6] and the whole Globe (ALL).

The results of these operation retain the same name assigned previously, at which we add the superscript provided in parenthesis near each name.

[6] The definition of each of these region in terms of lat-lon boundaries can be found in the Result section of the previous chapter.

3.4 Results

In the following tables we expose the scores for the two DMO algorithms tested in this chapter.

First of all, we look at the dependence of the scores from the forecast week. Like for the non-probabilistic scores of the previous chapter, we see a degradation of the performances with time. Focusing on the DV method, we see a significant difference between the multi-model and the single model, expecially for the $\overline{RPSS}$, shown in Table 3.1. The skill score for the multi-model is always more than double the other two, and this is particularly noticeable in the Equatorial Belt. If we look at the $\overline{RPS}$ in that region (not shown here, for avoiding redundancy due to the similarity with $\boldsymbol{RPSS}$), it may seem strange to discover that the value is not particularly high if compared to the ones relative to the remaining regions. We conclude that this score particularly low is due to $\overline{RPS}^{(clim)}$ beeing particularly good in comparison to the same value over the other regions. So, the same ranked probability score for the forecast result in different skill scores depending on how good is the prediction based only on the climatology. Note that, using $\boldsymbol{DRPSS}$ instead of $\boldsymbol{RPSS}$, the gap between the multi-model and the two single model is reduced. This is probably due to their difference in the ensemble dimension. The multi-model contains 25 members, so its correction term D is approximately equal to 0.018, while the other two contain 5 member each and, for them, D is five times greater, that is $D \simeq 0.09$. The consequence is that the discrete skill score for the smallest ensembles increase (respect to the non-discrete one) more than for the largest ensemble. Another interesting features is the almost compleye lack of predicting skill for the third and fourth weeks. On the contrary, for both DV and TPP, we observe particularly high values for the first week, often more than double te ones for the second week. For the multi-model, expecially for TPP and when using $\boldsymbol{DRPSS}$, we can see values above 0.65 and sometimes close to 0.7. About this split between the first two weeks and the remaining two we discussed in the previous chapter, so we will not repeat the same set of reasons.

Tukey Plotting Position seems more promising: its values are generally better than the DV counterparts. One of the most desirable features is that negative values for $\boldsymbol{RPSS}$ and $\boldsymbol{DRPSS}$ are less common. The simplicity of the algorithm, together with its capability of providing better results than DV, make TPP the optimal candidate for comparison with the more complex method tested in the following chapters.

Finally, we briefly discuss the dependence of the score from the region over which it is computed. The northern and the southern hemispheres seems to follow a common trend, with the first two weeks beeing rather predictable and the second one carachterized by values smaller but, for $\boldsymbol{RPSS}$ and $\boldsymbol{DRPSS}$, almost always positive. On the equatorial belt the behaviour is rather different. While the first week results significantly less predictable with respect to the other regions (expecially for the single models), the decrease in the third and fourth weeks is less marked and, for the multi-model, the scores remain particularly high. In Europe we see the opposite trend. Except for the first week, the values are often lower than their counterpars on the other regions. The last two weeks are almost totally unpredictable, with an $\boldsymbol{RPSS}$ always lower than zero, which means that the forecast has less skill than climatology. The $\boldsymbol{DRPSS}$ confirms

w	$\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{MM}}$	$\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{E}}$	$\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{C}}$
1	0.64	0.33	0.23
2	0.27	0.11	-0.02
3	0.09	-0.02	-0.09
4	0.05	-0.03	-0.08

(a) Northern Hemisphere

w	$\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{MM}}$	$\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{E}}$	$\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{C}}$
1	0.63	0.33	0.24
2	0.30	0.11	-0.02
3	0.15	0.00	-0.09
4	0.08	-0.06	-0.14

(b) Southern Hemishpere

w	$\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{MM}}$	$\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{E}}$	$\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{C}}$
1	0.47	0.09	-0.06
2	0.29	-0.01	-0.13
3	0.22	-0.05	-0.12
4	0.19	-0.06	-0.11

(c) Equatorial Belt

w	$\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{MM}}$	$\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{E}}$	$\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{C}}$
1	0.66	0.31	0.22
2	0.19	0.03	-0.10
3	-0.06	-0.10	-0.18
4	-0.07	-0.10	-0.14

(d) Europe

Table 3.1: Ranked probability skill score for the Democratic Voting (DV) method, averaged over the 18 validation winters. The four table present the spatial average over the four different regions defined in the previous chapter and the first column shows, in blue, the week. The remaining column refer to the three models: in the first one the are the values for the multi-model ($\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{MM}}$), in the second the ECMWF-IFS ones ($\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{E}}$) and in the third the CNR-ISAC ones ($\boldsymbol{RPSS}^{(\mathrm{DV})}_{\mathrm{C}}$). The value corresponding to the best performances (the highest ones) for each row is highlighted in red.

w	$\boldsymbol{RPSS}_{\mathrm{MM}}^{(\mathrm{TPP})}$	$\boldsymbol{RPSS}_{\mathrm{E}}^{(\mathrm{TPP})}$	$\boldsymbol{RPSS}_{\mathrm{C}}^{(\mathrm{TPP})}$
1	0.65	0.39	0.32
2	0.29	0.17	0.08
3	0.11	0.05	0.00
4	0.07	0.03	0.00

(a) Northern Hemisphere

w	$\boldsymbol{RPSS}_{\mathrm{MM}}^{(\mathrm{TPP})}$	$\boldsymbol{RPSS}_{\mathrm{E}}^{(\mathrm{TPP})}$	$\boldsymbol{RPSS}_{\mathrm{C}}^{(\mathrm{TPP})}$
1	0.64	0.40	0.33
2	0.32	0.18	0.09
3	0.17	0.07	0.01
4	0.11	0.02	-0.03

(b) Southern Hemishpere

w	$\boldsymbol{RPSS}_{\mathrm{MM}}^{(\mathrm{TPP})}$	$\boldsymbol{RPSS}_{\mathrm{E}}^{(\mathrm{TPP})}$	$\boldsymbol{RPSS}_{\mathrm{C}}^{(\mathrm{TPP})}$
1	0.49	0.19	0.09
2	0.31	0.09	0.01
3	0.24	0.04	0.00
4	0.21	0.03	0.00

(c) Equatorial Belt

w	$\boldsymbol{RPSS}_{\mathrm{MM}}^{(\mathrm{TPP})}$	$\boldsymbol{RPSS}_{\mathrm{E}}^{(\mathrm{TPP})}$	$\boldsymbol{RPSS}_{\mathrm{C}}^{(\mathrm{TPP})}$
1	0.67	0.38	0.31
2	0.21	0.10	0.02
3	-0.02	-0.02	-0.08
4	-0.04	-0.03	-0.05

(d) Europe

Table 3.2: As in Table 3.1, but for the Tukey Plotting Position.

w	$\boldsymbol{DRPSS}_{\text{MM}}^{(\text{DV})}$	$\boldsymbol{DRPSS}_{\text{E}}^{(\text{DV})}$	$\boldsymbol{DRPSS}_{\text{C}}^{(\text{DV})}$
1	0.68	0.39	0.30
2	0.34	0.19	0.07
3	0.17	0.07	0.01
4	0.14	0.06	0.02

(a) Northern Hemisphere

w	$\boldsymbol{DRPSS}_{\text{MM}}^{(\text{DV})}$	$\boldsymbol{DRPSS}_{\text{E}}^{(\text{DV})}$	$\boldsymbol{DRPSS}_{\text{C}}^{(\text{DV})}$
1	0.67	0.39	0.31
2	0.37	0.20	0.07
3	0.23	0.09	0.01
4	0.17	0.04	-0.03

(b) Southern Hemishpere

w	$\boldsymbol{DRPSS}_{\text{MM}}^{(\text{DV})}$	$\boldsymbol{DRPSS}_{\text{E}}^{(\text{DV})}$	$\boldsymbol{DRPSS}_{\text{C}}^{(\text{DV})}$
1	0.52	0.17	0.04
2	0.36	0.08	-0.03
3	0.29	0.04	-0.02
4	0.27	0.04	-0.01

(c) Equatorial Belt

w	$\boldsymbol{DRPSS}_{\text{MM}}^{(\text{DV})}$	$\boldsymbol{DRPSS}_{\text{E}}^{(\text{DV})}$	$\boldsymbol{DRPSS}_{\text{C}}^{(\text{DV})}$
1	0.69	0.37	0.29
2	0.27	0.11	0.00
3	0.04	0.00	-0.08
4	0.03	0.00	-0.04

(d) Europe

Table 3.3: Discrete ranked probability skill score for the Democratic Voting (DV) method, averaged over the 18 validation winters. The divion in the four spatial region and in weeks is done as in Table 3.1, as well as the color highlighting. As for RPSS, the best values are the highest ones. Each of the remaining column refer to a different model: in the first one the are the values for the multi-model ($\boldsymbol{DRPSS}_{\text{MM}}^{(\text{DV})}$), in the second the ECMWF-IFS ones ($\boldsymbol{DRPSS}_{\text{E}}^{(\text{DV})}$) and in the third the CNR-ISAC ones ($\boldsymbol{DRPSS}_{\text{C}}^{(\text{DV})}$). The value corresponding to the best performances (the lowest ones).

w	$\boldsymbol{DRPSS}_{\mathrm{MM}}^{(\mathrm{TPP})}$	$\boldsymbol{DRPSS}_{\mathrm{E}}^{(\mathrm{TPP})}$	$\boldsymbol{DRPSS}_{\mathrm{C}}^{(\mathrm{TPP})}$
1	0.69	0.45	0.38
2	0.36	0.25	0.17
3	0.19	0.14	0.10
4	0.16	0.12	0.09

(a) Northern Hemisphere

w	$\boldsymbol{DRPSS}_{\mathrm{MM}}^{(\mathrm{TPP})}$	$\boldsymbol{DRPSS}_{\mathrm{E}}^{(\mathrm{TPP})}$	$\boldsymbol{DRPSS}_{\mathrm{C}}^{(\mathrm{TPP})}$
1	0.68	0.45	0.39
2	0.38	0.26	0.17
3	0.25	0.16	0.10
4	0.19	0.11	0.06

(b) Southern Hemishpere

w	$\boldsymbol{DRPSS}_{\mathrm{MM}}^{(\mathrm{TPP})}$	$\boldsymbol{DRPSS}_{\mathrm{E}}^{(\mathrm{TPP})}$	$\boldsymbol{DRPSS}_{\mathrm{C}}^{(\mathrm{TPP})}$
1	0.53	0.27	0.18
2	0.38	0.18	0.10
3	0.31	0.13	0.09
4	0.29	0.12	0.09

(c) Equatorial Belt

w	$\boldsymbol{DRPSS}_{\mathrm{MM}}^{(\mathrm{TPP})}$	$\boldsymbol{DRPSS}_{\mathrm{E}}^{(\mathrm{TPP})}$	$\boldsymbol{DRPSS}_{\mathrm{C}}^{(\mathrm{TPP})}$
1	0.70	0.43	0.38
2	0.29	0.18	0.11
3	0.07	0.07	0.02
4	0.06	0.07	0.04

(d) Europe

Table 3.4: As for Table 3.3, but for the Tukey Plotting Position.

this lack of skill, altought for themulti-model the score is slightly positive (but very close to zero).

3.4.1 Reliability Diagram: DMO results

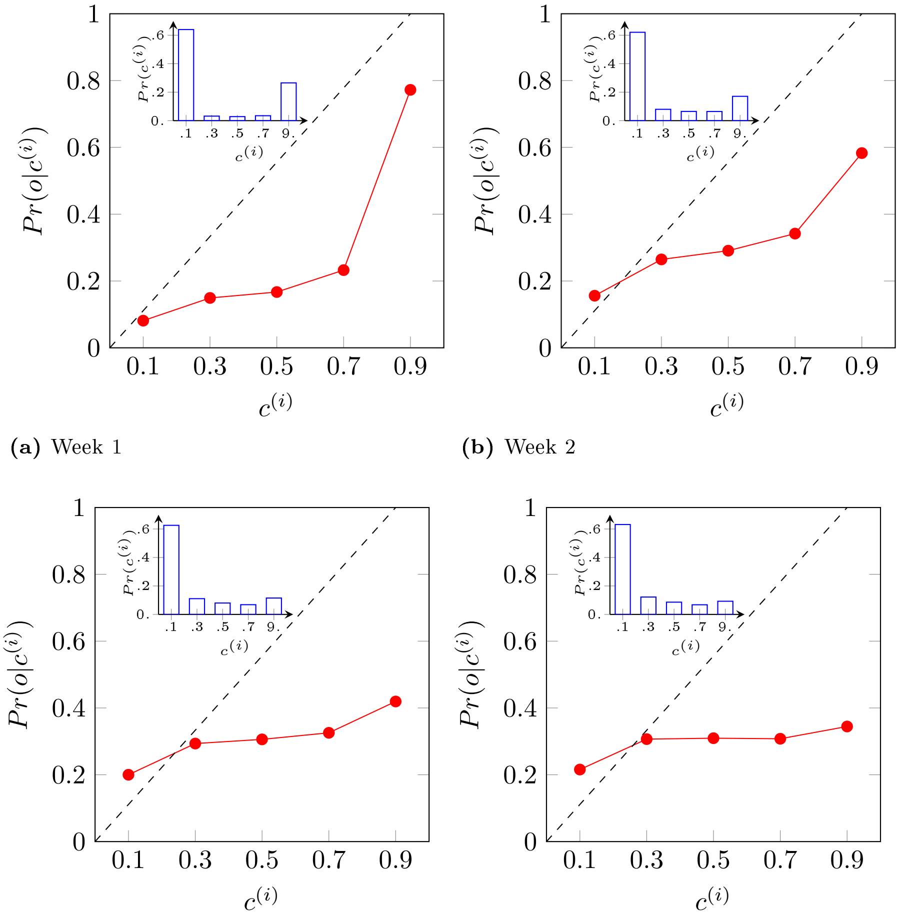
In this section, we provide the reliability diagrams obtained for the DMO tecniques applied to the multi-model.

Unlike for the other three scores, DV and TPP obtain rather similar results: most of the time the difference between the correspondig values in different plot show changes only starting from the second decimal place. So, we can present a general discussion that can be considered valid for both algorithms. Recall that the reliability diagram can be used only for binary outcomes, so we need to treat separately the two terciles. However, the differences between the two sets of plot are again very limited, like for the comparison between different algorithms. Therefore we further reduce the description of the results, analyzing the common trends.

As expected, the first week presents the calibration funcition (the red one) closer to the ideal case (the dashed line), in all the four plots (Figures 3.4-3.5-3.6-3.7). While the values for the first and last intervals are closer to the bisector of the first quadrant, the performances in the other are poorer. In general, all the curve falls below the dashed line, and as described previously when discussinf Figure 3.2-(b), we are in presence of an overforecasting problem. Our algorithms produces probabilities higher than the verifying frequencies, expecially in the middle of the output range. In addition, an analysis of the refinement distribution reveals an high confidence problem, with the two extremes of the distribution having a aprticularly high frequenct respect to the central values.

In the following week the slope of the curve decreases, with the final week presenting an almost horizontal calibration function. This means that the algorithm becomes increasingly similar to the climatology, with the five categories becoming approximately equiprobable. The high confidence seen in the refinement distribution persists during the second week of the forecast, but the column corresponding to $p^{(5)}$ decreases in height. In the following weeks all categories except the first converge to a common value.

These results will serve as reference values for the regression algorithms, from which we expect some improvements.



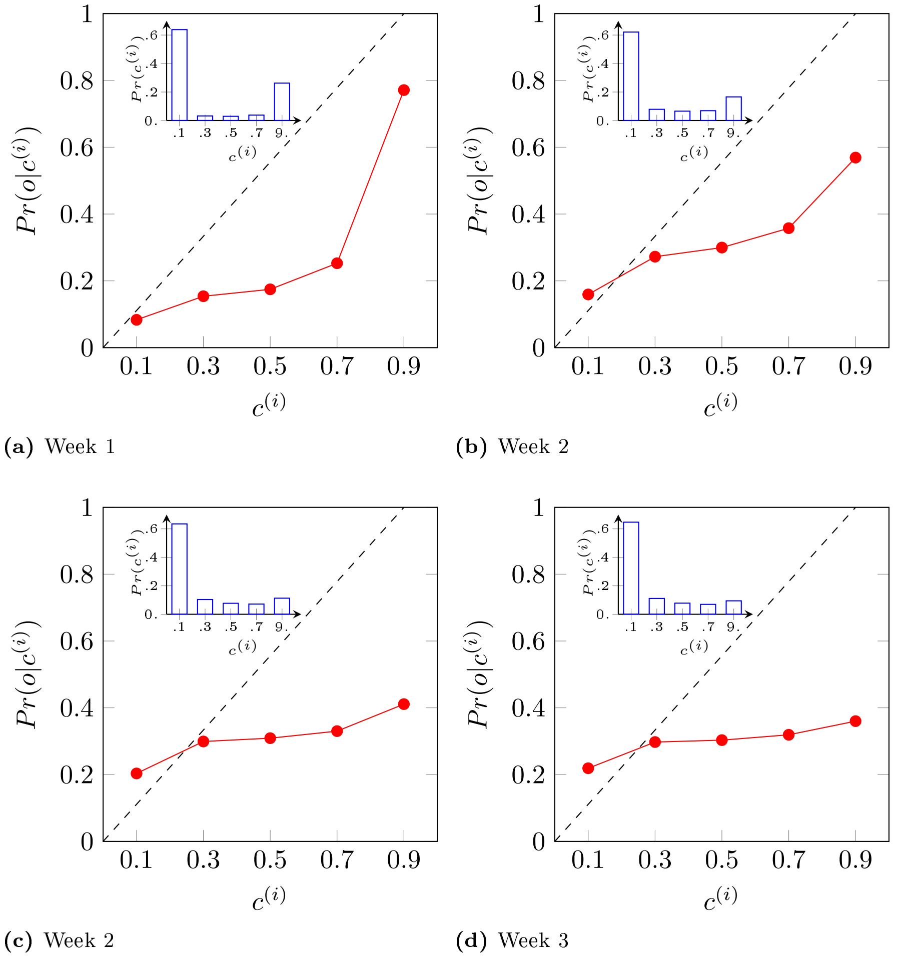
(a) Week 1

(b) Week 2

(c) Week 3

(d) Week 4

Figure 3.4: Reliability diagrams for the democratic voting algorithm, computed for teh lower tercile tresholds. The four panel refer to the forecast weeks, as suggested by the label below them. Each of the panel has the same structure: the plot outside shows the calibration function (in red), while the smoller plot in the corner contains the refinement distribution (in blue).



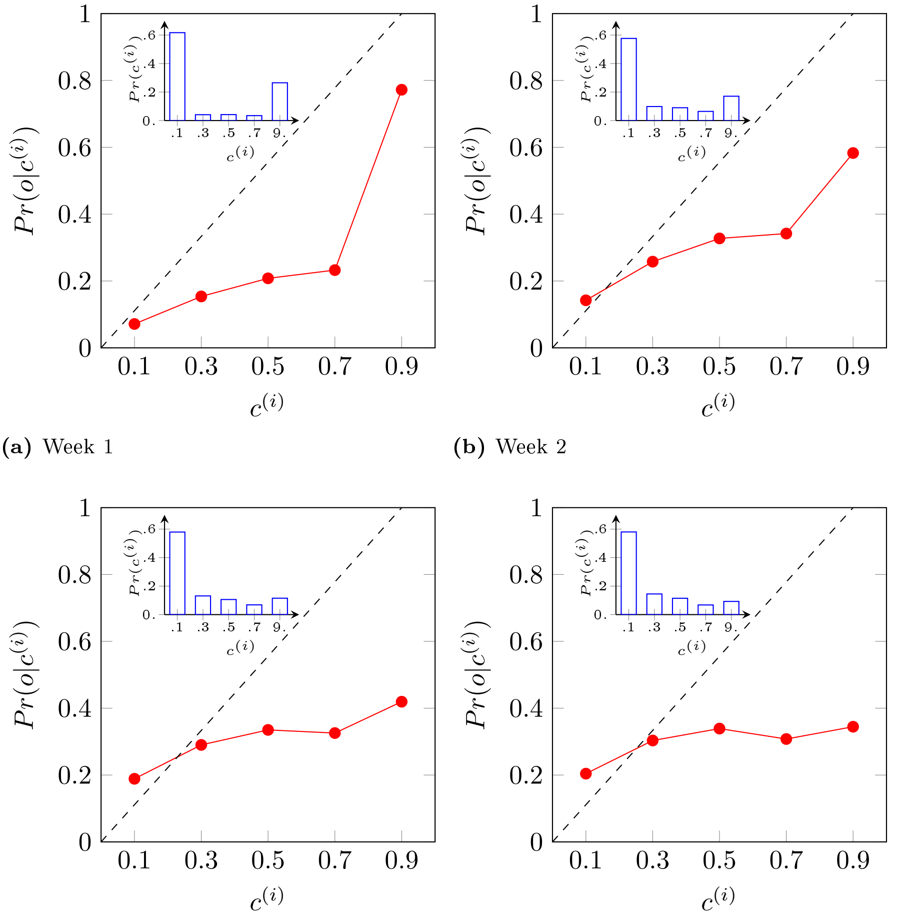
(a) Week 1

(b) Week 2

(c) Week 2

(d) Week 3

Figure 3.5: As in Figure 3.4, but for the upper tercile. Note the similarities between the two plots: most of the values differ only on the second decimal place.



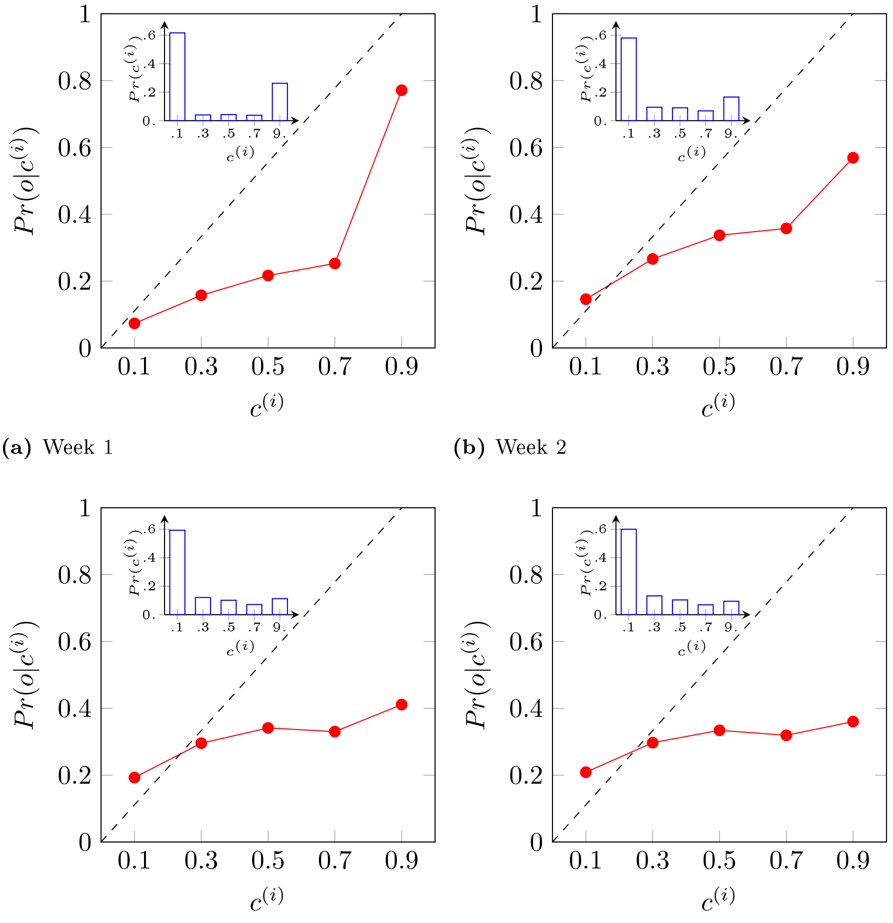
(a) Week 1

(b) Week 2

(c) Week 2

(d) Week 3

Figure 3.6: As in Figure 3.4, but for the Tukey plotting position method applied to the lower tercile and TPP method.



(a) Week 1

(b) Week 2

(c) Week 2

(d) Week 3

Figure 3.7: As in Figure 3.6, but for the upper tercile.

Chapter 4

Logistic Regression

Due to the presence of bias and dispersion errors, DMO methods can lead to unreliable resuslts. In our case, the 2-metre temperature provided by the model can be, on average, higher or lower than the real one, or the dispersion of the ensemble members cannot truly represent the real forecast uncertainty. Using a regression technique can help to minimize the impact of these imperfections on the results. [Wilks, 2011]

Being one of the most widely used algrtihms in literature (some examples are Hamill et al. [2004], Whitaker et al. [2006], Wilks [2006], Wilks and Hamill [2007], Wilks [2009]) we decided to use LR as the first MOS technique for forecasting tercile probabilities. As mentioned before, LR can be implemented in slightly different ways, depending on the choice of the input features. However, before discussing these details, it can be useful to give a small overview on the general functioning of the algorithm.

4.1 The basic algorithm

Logistic regression is one of the basic machine learning techniques[1]. In this analysis, it is used to predict the probability of a binary event (the temperature anomaly being above or below each tercile) given a set of features. In order to describe how it works, we can use a simple theoretical example: predicting the probability of the (dependent) variable a being above the threshold a^*, given the values of the indipendent variable (the predictor or feature) b.

[1] In this and in the following sections, we describe logistic regresson in its basic form and then we provide the methodology for applying it to our specific task. All the information contained in this description derive from the online course (often referred as Massively Open Online Course, or MOOC) of *Machine Learning* from Coursera, provided by Standford University and taught by Professor Andrew Ng:
https://www.coursera.org/learn/machine-learning.
Logistic regression is a rather old algorithm and it is used for a wide range of application, so this specialized course on machine learning gives to the student the knowledge for implementing such techniques in an efficient way. In order to manage wisely our computational resources, we applied LR following the suggestions from the online class, from which we also took the notation of the chapter, when possible. So, if not stated otherwise, the sources of the description is always the online course. Naturally, a teoretical explanation of the algorithm can be also found in Wilks [2011], where additional references to the scientific literature (such as Applequist et al. [2002], Watson and Colucci [2002], Lehmiller et al. [1997] and Wilks [2009]) regarding some application to the atmospheric forecasts are given.

For the task, we have a training dataset of m examples, in which we know both a and b. The value of each variable at the i-th example is given respectively by $a^{(i)}$ and $b^{(i)}$, with $i = 1, ..., m$.

First of all, we define a binary verification vector $\boldsymbol{y}$ following the rule:

$$\boldsymbol{y} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(i)} \\ \vdots \\ y^{(m)} \end{bmatrix}, \quad \text{with} \quad y^{(i)} = \begin{cases} 1 & \text{if } a^{(i)} > a^* \\ 0 & \text{otherwise.} \end{cases}$$

Then, we create two feature vectors. The first is $\boldsymbol{f}_0$, its entries are all equal to one and it is often called the "bias unit", while the other is $\boldsymbol{f}_1$ and contains the values of b:

$$\boldsymbol{f_0} = \begin{bmatrix} f_0^{(1)} \\ \vdots \\ f_0^{(i)} \\ \vdots \\ f_0^{(m)} \end{bmatrix} = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \qquad \boldsymbol{f_1} = \begin{bmatrix} f_1^{(1)} \\ \vdots \\ f_1^{(i)} \\ \vdots \\ f_1^{(m)} \end{bmatrix} = \begin{bmatrix} b^{(1)} \\ \vdots \\ b^{(i)} \\ \vdots \\ b^{(m)} \end{bmatrix}.$$

From these vectors we obtain the feature matrix:

$$\boldsymbol{X} = \begin{bmatrix} — & (f_0)^{\mathrm{T}} & — \\ — & (f_1)^{\mathrm{T}} & — \end{bmatrix} = \begin{bmatrix} 1 & \dots & 1 & \dots & 1 \\ b^{(1)} & \dots & b^{(i)} & \dots & b^{(m)} \end{bmatrix}.$$

Each column in $\boldsymbol{X}$ represents the value of all the features for a given example (i) and constitute a vector that, in order to simplify the notation in the following discussion, we call $\boldsymbol{f}^{(i)}$:

$$\boldsymbol{f}^{(i)} = \begin{bmatrix} f_0^{(i)} \\ f_1^{(i)} \end{bmatrix} = \begin{bmatrix} 1 \\ b^{(i)} \end{bmatrix}$$

In addition, for each of the feature we introduduce a coefficient: θ_0 and θ_1. Similarly to the procedure just described, these two values are gathered in a single vector:

$$\boldsymbol{\theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}.$$

Obviously, the coefficients do not vary across the dataset, therefore there is not the superscript (i). Note that, unlike all the matrices and vectors previously described, $\boldsymbol{\theta}$ contains unknown values.

Finally, for a given example, the probability of $a^{(i)}$ being above a^* is modeled using the logistic function, which acts as the "hypothesis function" for this particular alogirithm:

$$h_\theta(\boldsymbol{f}^{(i)}) = \frac{1}{1 + \exp(-\boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{f}^{(i)})}$$

As a mean for keeping the example more concrete, we assign actual values to $a^{(i)}$, $b^{(i)}$ and a^*, creating a dataset with $m = 15$ entries. I Figure 4.1 we show on

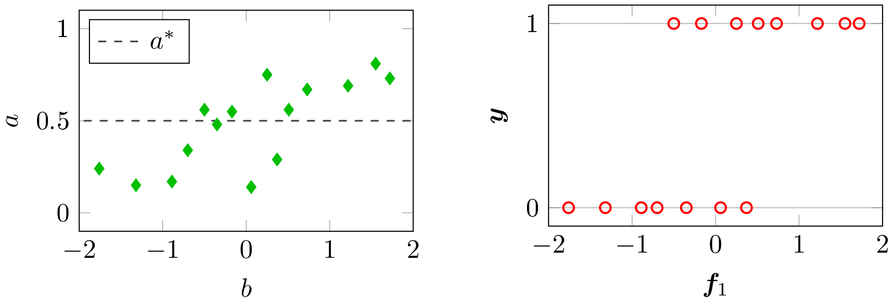


Figure 4.1: Dataset for the example of the basic version of logistic regression. On the left, the scatter plot showing on the x-axis the independent variable b (the predictor) and on the y-axis the dependent one, b. The dashed line represent the threshold a^*. On the right we show the binary verification vector $\boldsymbol{y}$ as a function of the feature vector $\boldsymbol{f}_1$, obtained respectively from a and b.

the left a scatter plot of $a^{(i)}$ and $b^{(i)}$, with the threshold a^* represented by the black dashed line. On the right there are the vectors $\boldsymbol{f}_1$ and $\boldsymbol{y}$ obtained from these values. The pattern is rather clear: higher values of f_1^i often correspond to an higher probability of a^i being greater than a^*. This is the rule that, in the current exampole, LR has to learn.

For completing this task, we start with some random initialized coefficients $\boldsymbol{\theta}_{\text{ini}}$. It is likely that the hypothesis function cannot model the desired probability using this random vector. So, it is necessary a way to determine the optimal values for $\boldsymbol{\theta}$: the aloìgorithm needs to be trained.

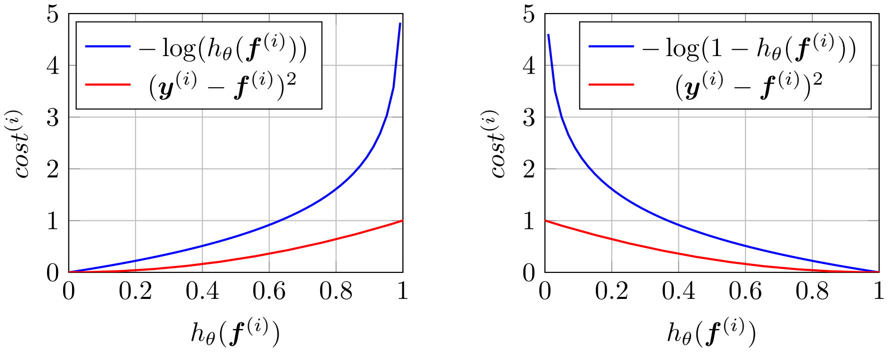
The first step is the decision of a "cost function". Its role is to estimate how much $h_\theta(\boldsymbol{f}^{(i)})$ differs from $\boldsymbol{y}^{(i)}$. The general form of such a function is the following:

$$J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i=1}^{m} cost^{(i)},$$

where $cost^{(i)}$ is simply "cost" for the i-th example. It is the measure of how much we want to penalize the algorithm if the outcome is $h_\theta(\boldsymbol{f}^{(i)})$ while the desired value is $\boldsymbol{y}^{(i)}$. A possible candidate is the squared difference between these two values, that happens to be the same function minimized in the linear regression of the previous chapter:

$$J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(\boldsymbol{f}^{(i)}) - \boldsymbol{y}^{(i)}\right)^2.$$

However, while in the previous chapter the hypothesis function was linear, in the current algorithm it is not. This choice for $J(\boldsymbol{\theta})$ is a non-convex function and this leads to some possible problems during the optimization. If we use the non-linear h_θ in this J, we may find that there are multiple local optima, therefore not a desirable quality for a cost function. The problem arises when we want to perform a numerical minimization: the optimization algorithm could find one of those local minima and stay in it, without converging to the global minimum.



(a) $\boldsymbol{y}^{(i)} = 0$

(b) $\boldsymbol{y}^{(i)} = 1$

Figure 4.2: Behaviour of the two possible choices for the term $cost^{(i)}$ in $J(\boldsymbol{\theta})$. We show the cost as a function of the feature vector $\boldsymbol{f}$, for a fixed example (i). The verification vector is also constant in each panel: in **(a)** its value is 0, while it is equal to 1 in **(b)**. As the legend suggests, the blue curve represent the logarithmic cost function while the red one shows the other possibility, based on the squared difference.

To avoid this problem, we decide an alternative (convex) cost function:

$$cost^{(i)} = \begin{cases} -\log\big(h_\theta(\boldsymbol{f}^{(i)})\big) & \text{if } \boldsymbol{y}^{(i)} = 1 \\ -\log\big(1 - h_\theta(\boldsymbol{f}^{(i)})\big) & \text{if } \boldsymbol{y}^{(i)} = 0. \end{cases}$$

With some simple algebra, it can be shown that this is equivalent to:

$$J(\boldsymbol{\theta}) = \frac{1}{m} \sum_{i=1}^{m} \Big(\boldsymbol{y}^{(i)} \log\big(h_\theta(\boldsymbol{f}^{(i)})\big) + (1 - \boldsymbol{y}^{(i)}) \log\big(1 - h_\theta(\boldsymbol{f}^{(i)})\big) \Big). \tag{4.1}$$

To better understand how the cost function works, in Figure 4.2 we show the value of $cost^{(i)}$ for a single example in the dataset. The two cases $\boldsymbol{y}^{(i)} = 0$ and $\boldsymbol{y}^{(i)} = 1$ are treated respectively in **(a)** and **(b)**. The similarity between the two possible choices for $cost^{(i)}$ is evident: both of them increase whith the rise in the difference between the hypothesis function and $\boldsymbol{y}^{(i)}$. The blue curve (the one using the logarithms) has a sharper increase (in **(a)**) and decrease (in **(b)**) than the red one. This results in a slightly contrasting behaviour when treating the two extreme cases: in the logarithmic case, h_θ is more penalized when it is particularly far from the real value $\boldsymbol{y}^{(i)}$, while the contrary happens when these two values are closer. Note that what matters in deciding how much h_θ is penalized is the relative value of $cost^{(i)}$ at the two extremes of the codomain of h_θ, not its absolute value. This is a consequence of the freedom in the choice of the "cost" and, of course, of the consistent use of the same cost function troughout the regression, without switching between various possibilities.

Due to its nature of convex function, we chose for the following analysis to use always the "logarithmic" cost function (4.1). Of this function, we also know the gradient:

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \theta_j} = \frac{1}{m} \sum_{i=1}^{m} \big(h_\theta(\boldsymbol{f}^{(i)}) - \boldsymbol{y}^{(i)}\big) \boldsymbol{f}_j^{(i)},$$

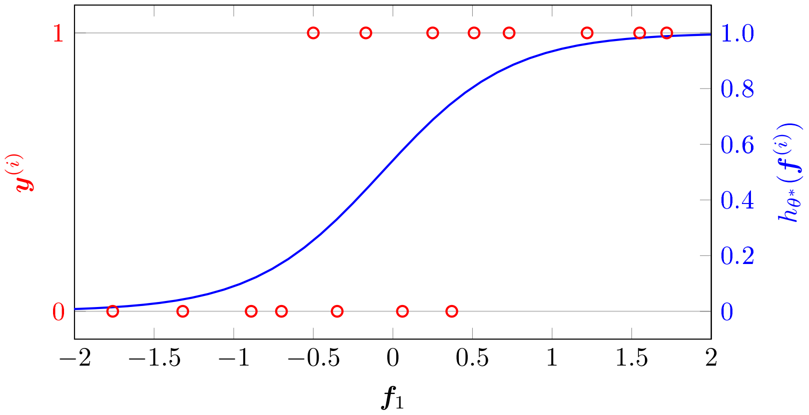


Figure 4.3: Hypothesis function (in blue) for the simple example described in the first section of the chapter, together with the binary verification vector (in red). The scale of the two axis is the same, however a larger number of ticks are shown on the right one due to h_θ being a continuous function while $\boldsymbol{y}$ can assume only the values 0 and 1. On the x-axis there is the feature vector $\boldsymbol{f}_1$, the whole dataset is the same as in Figure 4.1. The value θ^* used for the hypothesis function is obtained through a real minimization of the cost function over the examples, using BFGS.

with $j = 0, 1$ in our example. The gradient plays an important role in the minimization algorithm. Normally, dfferent methods can be used to find the optimal $\boldsymbol{\theta}$: all of them require the knowledge of the cost function, but only a subset also needs $\partial J(\boldsymbol{\theta})/\partial\theta_j$ (and there are algorithms with higher degree of complexity that make use of the Hessian matrix or other additional informations).

We, after some performance test using the real dataset (the 2-metre temperature anomaly fields) for computing the tercile probabilities, decided to use as minimization method the quasi-Newton method of Broyden, Fletcher, Goldfarb, and Shanno (BFGS) [Jeorge Nocedal, 2006] as provided by the function *minimize* from the *optimize* module of the *scipy* package (Python 2.7)[2].

Going back to our simple example, we now possess all the ingredients for applying LR. We give to the minimization routine the cost function $J(\boldsymbol{\theta})$, its gradient $\partial J(\boldsymbol{\theta})/\partial\theta_j$ and the initial guess for the coefficient $\boldsymbol{\theta}_{ini}$ previously defined. The output is the set of coefficients $\boldsymbol{\theta}^*$ that achieve the lowest cost. Using them, we obtain the plot shown in Figure 4.3, in which the curve in blue represents h_{θ^*} as a continuous function of $\boldsymbol{f}$ while the binary verification data are reported in red.

As expected, LR has learnt to predict an high probability of $\boldsymbol{y}$ being equal to one (or, in other terms, of a being above the threshold a^*) when f_1 (or analogously b) is sufficently greater than zero. This example can be seen as the basic implementation of this machine learning algorithm, useful for understanding how the fundamental components work and what roles they play. However, before apllying the procedure to our real dataset, there are some additional terms that need to be introduced and discussed, in order to achieve adequate performances.

[2] More information about the function can be found on the scipy website: *http://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html* .

4.2 More advanced concepts

In this section we summarize all the modifications to the basic algorithm introduced before. The description has been split in subsections, to keep the discussion clear and understendable.

4.2.1 Feature space with higher dimensionality

The first modification introduced is the possibility of applying LR using more than a single feature. The vector notation already introduced can help with this task, as we show in the following example. Instead of having only one predictor b, suppose that there are n values, $b_1, ..., b_n$. Starting from them, we create the features vectors:

$$\boldsymbol{f_0} = \begin{bmatrix} f_0^{(1)} \\ \vdots \\ f_0^{(i)} \\ \vdots \\ f_0^{(m)} \end{bmatrix} = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \ \boldsymbol{f_1} = \begin{bmatrix} f_1^{(1)} \\ \vdots \\ f_1^{(i)} \\ \vdots \\ f_1^{(m)} \end{bmatrix} = \begin{bmatrix} b_1^{(1)} \\ \vdots \\ b_1^{(i)} \\ \vdots \\ b_1^{(m)} \end{bmatrix}, \ldots, \boldsymbol{f_n} = \begin{bmatrix} f_n^{(1)} \\ \vdots \\ f_n^{(i)} \\ \vdots \\ f_n^{(m)} \end{bmatrix} = \begin{bmatrix} b_n^{(1)} \\ \vdots \\ b_n^{(i)} \\ \vdots \\ b_n^{(m)} \end{bmatrix}.$$

Analogously to the previous case, from these vectors we create the feature matrix:

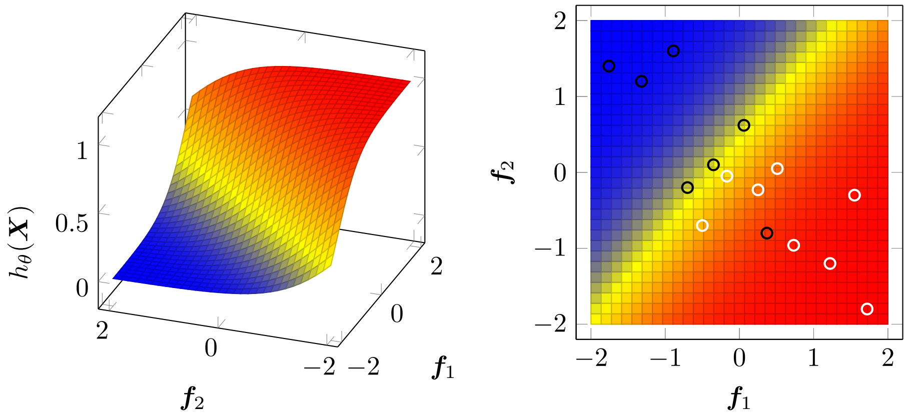
$$\boldsymbol{X} = \begin{bmatrix} — & (f_0)^{\mathrm{T}} & — \\ — & (f_1)^{\mathrm{T}} & — \\ \vdots & \vdots & \vdots \\ — & (f_n)^{\mathrm{T}} & — \end{bmatrix} = \begin{bmatrix} 1 & \ldots & 1 & \ldots & 1 \\ b_{(1)}^{(1)} & \ldots & b_{(1)}^{(i)} & \ldots & b_{(1)}^{(m)} \\ \vdots & & \vdots & & \vdots \\ b_{(n)}^{(1)} & \ldots & b_{(n)}^{(i)} & \ldots & b_{(n)}^{(m)} \end{bmatrix}.$$

A larger number of features also requires an adequate number of coefficients:

$$\boldsymbol{\theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{bmatrix}.$$

Naturally, the binary verification vector $\boldsymbol{y}$ remains the same.

Now that all the basic matices have been redefined, the vector notation demonstrates its usefulness: the hypothesis function, the cost function and its gradient preserves exactly the shape described in the previous section. In order to actually see what logistic regression does when used with more than one feature, we modify the previous example introducing a second feature b_2, renaming b as b_1. Without repeating all the steps, we show directly the results in Figure 4.4. The plot is three dimensional, with the two vector features on the x and y axis, while on the z axis h_θ is reported. Note that the projection of the surface representing the hypothesis function on a vertical plane has the same shape seen in the one dimensional case, which is exactly what we would expect from a generalization on more dimensions. The view "from above" **(b)** is particularly intersting: it shows us that on the feature space (that is the plane



(a) View from the side

(b) View from above

Figure 4.4: Hypothesis function for the 2-dimensional case presented as an exstension of the example from the previous chapter. The panel on the left shows a 3-dimensional view of the surface corresponding to h_θ, whose values are on the z-axis, while the x and y axis shows the two features $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$. On the right, the same plot but seen from above, therefore looking at the plane generated by $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$. The black circles correspond to $\boldsymbol{y}^{(i)} = 0$, while the white one represent the examples for which $\boldsymbol{y}^{(i)} = 1$. Note that as $\boldsymbol{f}_1$ we chose the same vector as in Figure 4.3, while $\boldsymbol{f}_2$ contains some new values. Again, the coefficient vector $\boldsymbol{\theta}^*$ used in h_θ has been obtained through numerical minimization of the cost function over the dataset.

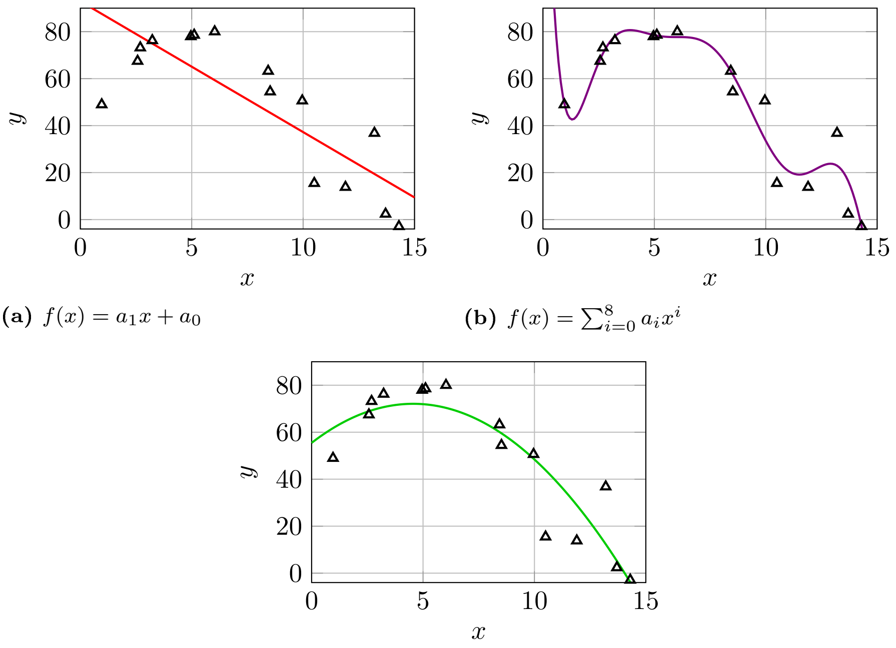
defined by the two features f_1 and f_2), the curves of equiprobability (the couples $(\boldsymbol{f}_1, \boldsymbol{f}_2)$ sharing the same value of $h_\theta(\boldsymbol{X})$) are straight lines. This generalizes to an arbitrary number of dimension in the following way: in a three dimensional feature space we find planes of equiprobability and, in the most general form, for an n-dimensional space there are $(n-1)$-dimensonal hyper-planes of equiprobability.

This is a fundamental feature of logistic regression and it imposes some constraints on which rule the algorithm can learn. Without modifying the features, LR cannot represent non-linear equiprobability curves. Of course, we can always add new feature that are non-linear modification of the original ones (like $(\boldsymbol{f}_1^{(i)})^2$, for example), but it is an operation performed manually and it is not always clear which non-linear function is the optimal representation of the function we want to learn.

4.2.2 Undefitting and overfitting

Logistic regression, like most of the regression methods, can suffer from undefitting or overfitting. These problems arise when the algorithm learns a function that does not fit the data correctly. In particular, they represent the two possible extreme cases:

- underfitting occur when the regression is not able to represent the complexity underlying the data;
- overfitting happens when a function with a too high degree of complexity



(a) $f(x) = a_1x + a_0$

(b) $f(x) = \sum_{i=0}^{8} a_ix^i$

(c) $f(x) = a_2x^2 + a_1x + a_0$

Figure 4.5: Three different fit of the same dataset (the black triangles), obtained through random perturbation on both x and y coordinates of a parabola. Panel **(a)** shows a linear regression of the points, that we use as an example of underfit. In **(b)** we use a octic fit (polynomial of degree 8), a clear example of overfit. Finally, in **(c)** we use a quadratic curve that, as expected, follows approximately the shape of the parabola from which the points have been generated. The determination of the coefficient has been performed using the *polyfit* function from Matlab. (Documentation: *http://www.mathworks.com/help/matlab/ref/polyfit.html* .)

is used for the regression, therfore it is not able to generalize well on data outside the training dataset.

They can be more easily understood through a simple example, like the one in Figure 4.5. We generated some points (the training set), first using the rule $y = b_2x^2+b_1x+b_0$ and then adding some random perturbations on both x and y, so that they follow what approximately seems a parabola. We test three possible fits: $f(x) = a_1x+a_0$ in **(a)**, $f(x) = \sum_{i=0}^{8} a_ix^i$ in **(b)** and $f(x) = a_2x^2+a_1x+a_0$ in **(c)**. Ideally, we want these function to learn from the data the initial rule, so that, if we extract some extra data (the test set) from the original function, they would be near the curve obtained from the fit.

In **(a)**, nearly all the points at the two extremes of the domain are below the red line, while the ones in the middle are above it. This is an hint that probably a linear fit is not the best one for this set of data, because of the excessive simplicity of the functions that it can generate. In addition, knowing the initial rule, **(a)** tells us that in an hypothetical test set, elements with x particularly small or high will probabily be far from the red curve. In **(b)** we see the opposite situation. At a firts glance, the fit seems almost perfect: the

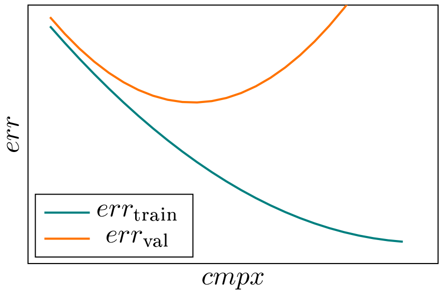


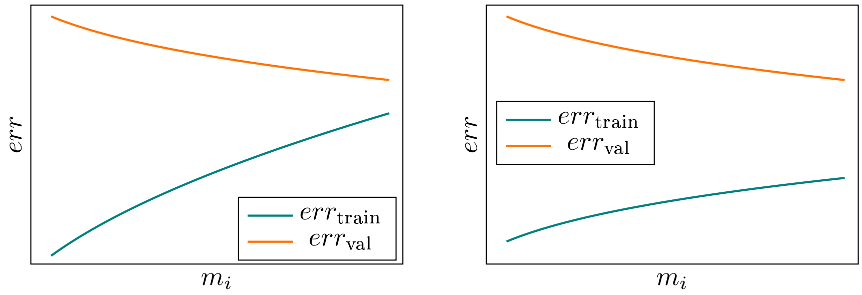
Figure 4.6: Ideal example of the behaviour of the error over the training (teal curve) and the validation (orange curve) sets, as a function of the complexity, *cmpx*, of the algorithm. All the quantity have not been clearly defined because we are discussing a general trend, valid for differen algorithms and their specific choice of the error function or their measure of complexity. Naturally, such a plot is useful as a conceptual tool more than as a concrete one, due to the difficulty (or impossibility) of varying *cmpx*.

violet curve is very close to nearly all the training data. However, looking again at the extremes of the x-axis, this function will not perform well on the test set. The function generated by the fit tries to pass through all the points, reaching an high degree of complexity that is far from the original function. We can imagine that on the test set, expecially near $x = 0$ (where the violet curve has extremely high values) or $x = 15$ (where the function decreases sharply), this fit will have low performance. Obviously, the right fit is the one shown in **(c)**: it is nearer to the training points than the red curve and less close than the violet one, but it has the right shape to perform relatively well also on the test set. Naturally, this is only a qualitative discussion, we did not introduce error bars to quantify the performances. The aim of these plots is only to give an intuitive idea of the two terms just introduced.

Now that we have an intuitive idea of these two problems, we can ask ourself: “how can we identify such problems in our logistic regression? And how can we deal with them?”

To answer the first question, we normally have to look at the preformance of the algorithm on both a training and a validation set of data. In fact, it is not always possible to look at the hypothesis function (the analogous of the $f(x)$ of the previous example), because the feature space can have an high dimensionality. So, a visual analysis like the one just presented is a very limited approach, useful only when we have one or two features and a small dataset.

A more general approach is looking at the error (the specific function chosen for this role can vary depending on the problem, so we name it *err*) on the two datasets. Imagine that we can summarize the complexity of the algorithm using a single variable, that we will call *cmpx*. Ideally, plotting *err* as a function of *cmpx* would result in a plot similar to the one in Figure 4.6. There we can see err_{train} in teal and err_{val} in orange: the first one in theory falls with the increase in complexity, while the second has an initial decline followed by a rise. In reality, a pattern like that is never so well defined, due to the presence of spikes and random fluctuations. In addition, changing the model complexity is often a difficult operation that requires redifying the features set and, consequently, writing entirely new programs, which need to run again on the whole dataset.



(a) **(b)**

Figure 4.7: Learning curves for two hypotetical cases in which the error over the validation set (orange curve) is considered too high. In panel **(a)** we see an underfitting example, with the errors over the training set (teal curve) and the validation one being rather close for high values of m (the dimension of the training set). In **(b)** we see the opposit scenario, overfitting, where the gap between the curves is significantly wider.

So, normally we do not possess such a plot, but only the values corresponding to the actual level of complexity of the algorithm used in the analysis. Nevertheless, it is an useful conceptual model for understanding how we can diagnose underfitting or overfitting. The first quantity that we usually evaluate is err_{val}, on which is often based our level of satisfaction with the method. However, an high value of this estimator cannot decrete by itself of which problem suffers our regression. So, it can be useful to compute also the error on the validation set. Normally, when the degree of complexity is excessively low, the algorithm performs relatively bad in both sets (with probably slightly better performances on the training one), this means that we are not using a tool capable of extracting all the information contained in our data. On the other hand, an algorithm capable of representing function too elaborate (and kept free to do so) can try to fit perfectly the training data, resulting in err_{train} particularly low. Nonetheless, when the same function is tested on another set, its performance are disappointing, like in the simple example shown before. The method is “too powerful” and its atempt to modelize also the random fluctuation (that inevitably affect the data) leads to an output rule that overlay with the real one only in some specific points, that are indeed the elements of the training set.

Another useful tool for checking eventual problems in out algoritm is the learning curve. Unlike the previous plot, this curve can be obtained rather simply. It shows the error, again on both training and validation set, as a function of the dimesnion of the training set, m. Normally, m is a fixed quantity that depends on how much data we possess, so increasing it is non always straightforward. Decreasing it, on the other hand, is often effortless. Therefore, we can choose some values $m_i < m$, for which we repeat the calculation, which hopefully needs less computational resources that the complete one, due to the lower amount of examples involved. Then, we plot the results and, ideally, the two extreme cases that we can obtain are the ones shown in Figure 4.7. In both of them, the validation error is too high (it is the same curve in **(a)** and **(b)**) and we want to understand what is the cause, so we compare it to err_{train} for all the m_i. In a realistic scenario, we would not have two continuous curves, but only some points, nevertheless in this theoretical discussion we wanted to keep the plots as simple as possible. At a first glance the two plot seems to show the same trend: err_{train} increases as a larger number of examples is used for the training (when using few data, even a simple algorithm can output a curve close to all of them, when the number rises the task become more dificult), while err_{val} decreases (due to the better training of the method, which had a larger amount data for extracting information on the underlying function). However, a closer look reveals a fundamental difference for high values of m_i. In **(a)**, the two errors are rather close and this is a classical example of underfitting. Recalling the first example, this behaviour can be easily understood imagining what would happen when we fit a straight line through some data generated by a quadratic funcrtion. In this case, adding more example would not improve the fit, because the function simply cannot represent the complexity of the original one. In **(b)** there is a more significant gap between the teal and the orange curve. This means that the algorithm is fitting quite well the training examples, while it is not generalizing well on the validation ones. This means that we are in presence of an overfitting problem.

Note that these learning curves are not only a duplicate of what we can discover just comparing the errors on the two set. First of all, the information that they brings is more complete than what we can extrapolate from a single couple of values. In addition, they give us useful information on how to solve these problem. This bring us back to the original questions, in particular to the second one. Normally, there are different options for dealing with these unwanted low performances. A small (and not exaustive) list include:

- obtaining more example to expand the training set,
- choosing a smaller set of features to use as input of the algorithm,
- increasing the number of features by adding different predictors or by creating new features starting from the original ones (like adding polinomial terms).

The first solution can help with overfitting. An intuitive idea for why this happens came from the learning curve. If we imagine to expand the x-axis, probably err_{val} will continue to decrease, until it arrives near the other curve, which in turn will rise. This kind of solution, however, does not work well with underfitting. Again, the learning curve helps with the intuition. Normally, err_{val} cannot be lower than err_{train} (except for some random fluctuation). In Figure 4.7 **(a)** the two values are already close together, the gap can decrease only by a small amount by increasing m.

Reducing the number of features can also help with overfitting but not with underfitting. This is rather evident looking at their definition and at the examples shown in Figure 4.5. The opposit operation can solve undefitting, as we can reasonably expect. In both operations, is implied that the choosing of the features is done evaluating each time which predictand brings more information and which, instead, is redundant.

Finally, the last solution that we propose is regularization. This introduces some significant changes in the algorithm, and deserves a separate subsection.

4.2.3 Regularization

Regularization starts as a method for adressing overfitting without having to decrease the number of features. It works by reducing the value of the parameters in the regression. Instead of focusing on a teoretical description, like in the previous section, we prefer to show directly its implementation on logistic regression. This is due to the notable modification it brings to the algorithm, in particular to the cost function.

When we initally discussed the basics of LR, we introduced the feature matrix $\boldsymbol{X}$ and the coefficent vector $\boldsymbol{\theta}$. Each member in the latter refer to one specific fetaure, which in turn constitutes a row in $\boldsymbol{X}$. With regularization, we want to keep the value of each coefficient relatively low. So, we introduce the vector $\boldsymbol{\lambda}$:

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda_0 \\ \vdots \\ \lambda_i \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ \lambda \\ \vdots \\ \lambda \end{bmatrix},$$

where n is the number of features (we are treating now the general case of a n-dimensional feature space). Note that $\boldsymbol{\lambda}$ has the same shape of $\boldsymbol{\theta}$, with $n+1$ entries, of which the first has the subscript 0 and refers to the bias unit $\boldsymbol{f}_0$. As a general rule, we do not apply regularization to $\boldsymbol{f}_0$, even if there is nothing that forbid such an operation. This is a standard practice in machine learning. Also, in all the cases we used the same value λ for all the features.

Using this notation, the cost function become:

$$J(\boldsymbol{\theta}) = \left(\frac{1}{m}\sum_{i=1}^{m}\Big(\boldsymbol{y}^{(i)}\log\big(h_\theta(\boldsymbol{f}^{(i)})\big) + (1-\boldsymbol{y}^{(i)})\log\big(1-h_\theta(\boldsymbol{f}^{(i)})\big)\Big)\right) + \\ + \frac{1}{2m}\sum_{j=0}^{n}\lambda_j\theta_j^2, \quad (4.2)$$

and the new gradient is given by:

$$\frac{\partial J(\boldsymbol{\theta})}{\partial\theta_j} = \left(\frac{1}{m}\sum_{i=1}^{m}\big(h_\theta(\boldsymbol{f}^{(i)}) - \boldsymbol{y}^{(i)}\big)\boldsymbol{f}_j^{(i)}\right) + \frac{1}{m}\lambda_j\theta_j. \quad (4.3)$$

The hypothesis function is not affected by regularization and remain the same as shown in the previous sections.

Since we finished the exposition of all the math behind the procedure, we can briefly explain intuitively what role does $\boldsymbol{\lambda}$ play. Normally, when there is overfitting, the regression is trying to pass as close as possible to all the training example. For doing so, the function needs to be sensitive to small changes in the inoput variables, and this transaltes in high values for the coefficients. The regularization term

$$J_{\text{reg}} = \frac{1}{2m}\sum_{j=0}^{n}\lambda_j\theta_j^2$$

has the specific role of targeting these high values. Inserting J_{reg} in the cost function is a way to equiparate an excessively high coefficient to the error associated with a bad fit (that usually depends on how much the examples are far from the curve). So, the minimization algorithm has to find a compromise between these two kind of error and is less prone to output extremely high coefficients, with the exception of these θs being really necessary for a good fit.

Finally, we can discuss how the parameter λ is chosen. Normally, we use a cross-validation approach, trying different versions of the algorithm each with a differen value for the regularization term (included $\lambda = 0$, that is the same of no-regularization). Then, the one that gives the best output is chosen as the final value. Sometimes, regularization is not needed. In these cases, cross-validation will give the best performances for $\lambda = 0$ or for some extremely low values of the parameter. However, its uselfuness often cannot be determined a priori, so a test with a small set of parameter is useful in most cases and it is worth the computational time spent. Note that, if not used correctly, regularization can lead to underfitting, preventing the algorithm to represent compelx enough functions. In this case, the solution is obviously to reduce λ.

4.2.4 Multi-class logistic regression

Analogously to what we did with the expansion of the feature space to n dimension, we can do the same operation to the output space. Using the example at the beginning of the section, instead of learning to predict if a variable a is above a threshold a^*, we can apply the algorithm to an arbitrary number of variables $a_1, ..., a_k$ (each with its threshold a_i^*, but this is a detail not important to understand the concept).

The first modification introduced is in the binary verification vector. Now we have k vectors:

$$\boldsymbol{y}_1 = \begin{bmatrix} y_1^{(1)} \\ \vdots \\ y_1^{(i)} \\ \vdots \\ y_1^{(m)} \end{bmatrix}, \dots , \boldsymbol{y}_j = \begin{bmatrix} y_j^{(1)} \\ \vdots \\ y_j^{(i)} \\ \vdots \\ y_j^{(m)} \end{bmatrix}, \dots , \boldsymbol{y}_k = \begin{bmatrix} y_k^{(1)} \\ \vdots \\ y_k^{(i)} \\ \vdots \\ y_k^{(m)} \end{bmatrix},$$

each of them defined by:

$$y_j^{(i)} = \begin{cases} 1 & \text{if } a_j^{(i)} > a_j^* \\ 0 & \text{otherwise.} \end{cases}$$

with $j = 1, ..., k$ being the index for the actual output predicted and $i = 1, ..., m$ the one for counting the examples in the dataset.

The simplest approach is to split the problem in k differen regression, using each time one of the $\boldsymbol{y}_j$ as the verification vector and following the same procedure explained in the previous section. When LR is used as a classification technique, this method is often called the "one vs. all classification". Altough this can be used in our specific case, we will follow a slightly different approach. This, however, requires at least a little introduction about the specific task at which we will apply LR. Even if the same procedure can be extended to a wider variety of problems, we think that it will be more easily understandable applied directly to the computation of tercile probabilities. So, we postpone the descpription to one of the following sections.

With the extension to more than just one class, we end the exposition of the slightly less basic versions of LR. Therfore, we proceed with the application of the algorithm to our analysis.

4.3 Computing tercile probabilities

In this section, we give a brief overview of how logistic regression is used for computing the probabilities that 2-metre temperature anomalies are in each one of the three terciles of the reanalysis distribution (over the training period).

For this section only, in order to simplify the discussion, we introduce a new notation for some quantities concerning the ensembles. While the tensors introduced when treating DMO was useful when working with all the forecast weeks and the grid points, during a more teorethical discussion it can resutl distracting, due to the high number of indexes, subscripts and superscripts. So, imagining to apply LR for predicting the probabilities for a fixed region referring to a single forecast time, we define:

- $\overline{x}^{(i)}$, the ensemble mean of the i-th example,
- $\sigma^{(i)}$, the ensemble standard deviation of the i-th example,
- $q_{1/3}$ and $q_{2/3}$, respectively the value of the first and second tercile of the reanalysis distribution,
- $t_{\mathrm{2m}}^{(i)}$, the value of the 2-metre temperature anomaly of the i-th example.

As always, m is the number of example in the training set.

Using these quantities, we first define the feature vectors:

$$\boldsymbol{f}_0 = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \quad \boldsymbol{f}_{\overline{x}} = \begin{bmatrix} \overline{x}^{(1)} \\ \vdots \\ \overline{x}^{(i)} \\ \vdots \\ \overline{x}^{(m)} \end{bmatrix}, \quad \boldsymbol{f}_{\sigma} = \begin{bmatrix} \sigma^{(1)} \\ \vdots \\ \sigma^{(i)} \\ \vdots \\ \sigma^{(m)} \end{bmatrix}, \quad \boldsymbol{f}_{\overline{x}\sigma} = \begin{bmatrix} \overline{x}^{(1)}\sigma^{(1)} \\ \vdots \\ \overline{x}^{(i)}\sigma^{(i)} \\ \vdots \\ \overline{x}^{(m)}\sigma^{(m)} \end{bmatrix},$$

from which we create three different feature matrices:

$$\boldsymbol{X}_{(\alpha)} = \begin{bmatrix} — & (f_0)^{\mathrm{T}} & — \\ — & (f_{\overline{x}})^{\mathrm{T}} & — \end{bmatrix},$$

$$\boldsymbol{X}_{(\beta)} = \begin{bmatrix} — & (f_0)^{\mathrm{T}} & — \\ — & (f_{\overline{x}})^{\mathrm{T}} & — \\ — & (f_{\sigma})^{\mathrm{T}} & — \end{bmatrix},$$

$$\boldsymbol{X}_{(\gamma)} = \begin{bmatrix} — & (f_0)^{\mathrm{T}} & — \\ — & (f_{\overline{x}})^{\mathrm{T}} & — \\ — & (f_{\overline{x}\sigma})^{\mathrm{T}} & — \end{bmatrix}.$$

They give us the shape of the three coefficient vectors:

$$\boldsymbol{\theta}^{(\alpha)} = \begin{bmatrix} \theta_0^{(\alpha)} \\ \\ \theta_{\overline{x}}^{(\alpha)} \end{bmatrix}, \quad \boldsymbol{\theta}^{(\beta)} = \begin{bmatrix} \theta_0^{(\beta)} \\ \\ \theta_{\overline{x}}^{(\beta)} \\ \\ \theta_{\sigma}^{(\beta)} \end{bmatrix}, \quad \boldsymbol{\theta}^{(\gamma)} = \begin{bmatrix} \theta_0^{(\gamma)} \\ \\ \theta_{\overline{x}}^{(\gamma)} \\ \\ \theta_{\overline{x}\sigma}^{(\gamma)} \end{bmatrix}.$$

Finally, we define the verification vectors for the two quantiles:

$$\boldsymbol{y}_{1/3} = \begin{bmatrix} y_{1/3}^{(1)} \\ \vdots \\ y_{1/3}^{(i)} \\ \vdots \\ y_{1/3}^{(m)} \end{bmatrix} \quad \text{with} \quad y_{1/3}^{(i)} = \begin{cases} 1 & \text{if } t_{2\mathrm{m}}^{(i)} < q_{1/3} \\ 0 & \text{otherwise,} \end{cases}$$

$$\boldsymbol{y}_{2/3} = \begin{bmatrix} y_{2/3}^{(1)} \\ \vdots \\ y_{2/3}^{(i)} \\ \vdots \\ y_{2/3}^{(m)} \end{bmatrix} \quad \text{with} \quad y_{2/3}^{(i)} = \begin{cases} 1 & \text{if } t_{2\mathrm{m}}^{(i)} < q_{2/3} \\ 0 & \text{otherwise.} \end{cases}$$

Naturally, three sets of features and coefficients imply that each of them is used in a slightly different implementation of the hypothesis function:

$$\begin{aligned} h_\theta^{(\alpha)}\big(\boldsymbol{X}_{(\alpha)}\big) &= \frac{1}{1+\exp\big(-(\boldsymbol{\theta}^{(\alpha)})^{\mathrm{T}}\boldsymbol{X}_{(\alpha)}\big)} \\ h_\theta^{(\beta)}\big(\boldsymbol{X}_{(\beta)}\big) &= \frac{1}{1+\exp\big(-(\boldsymbol{\theta}^{(\beta)})^{\mathrm{T}}\boldsymbol{X}_{(\beta)}\big)} \\ h_\theta^{(\gamma)}\big(\boldsymbol{X}_{(\gamma)}\big) &= \frac{1}{1+\exp\big(-(\boldsymbol{\theta}^{(\gamma)})^{\mathrm{T}}\boldsymbol{X}_{(\gamma)}\big)} \end{aligned}$$

From these values we extract directly the probabilities. The coefficients obtained using $\boldsymbol{y}_{1/3}$ and $\boldsymbol{y}_{2/3}$ result in h_θ being the probability that the temperature anomaly is respectively below the first and second tercile ($Pr(t_{2\mathrm{m}}^{(i)} < q_{1/3})$ and $Pr(t_{2\mathrm{m}}^{(i)} < q_{2/3})$) for the i-th example.

Before proceeding, it can be useful a brief explanation of why we have decided to implement these three version of the algorithm. They differ only for the predictors, so we focus on the reason behind their choice. The first matrix, $\boldsymbol{X}_{(\alpha)}$, contains only the ensemble mean. This is the most basic set of features we tried: knowing the the terciles and value of the anomaly predicted (averaged on all the ensemble members), LR extrapolates the probability that the latter is above or below the threshold. The other two matrices, $\boldsymbol{X}_{(\beta)}$ and $\boldsymbol{X}_{(\gamma)}$, contain also the ensemble spread. Altough in some studies [Hamill et al., 2004] the introduction of this predictor did not improve the final outcome, we tried these two implementation with the hope that the standard deviation would bring some useful infomation. Intuitively, the same distance between $\overline{x}^{(i)}$ and $q_{1/3}$ or $q_{2/3}$ can result in different probabilities of $t_{2\mathrm{m}}^{(i)}$ being in one tercile or another depending on how scattered are the ensemble members. The reason for using the product $\overline{x}\sigma$ lies in the possibility of interpreting algorithm as a regression using only $\overline{x}$ as input feature, but with the coefficient associated to it depending linearly on σ [Wilks, 2011]:

$$\theta_0^{(\gamma)} + \theta_{\overline{x}}^{(\gamma)}\overline{x} + \theta_{\overline{x}\sigma}^{(\gamma)}\overline{x}\sigma = \theta_0^{(\gamma)} + \theta_{\overline{x}}^{*}(\sigma)\overline{x} \quad \text{with} \quad \theta_{\overline{x}}^{*}(\sigma) = \theta_{\overline{x}}^{(\gamma)} + \theta_{\overline{x}\sigma}^{(\gamma)}\sigma$$

However, in literature [Wilks, 2006] the use of only σ as the second predictor has lead to slightly better forecast, on some artificial dataset. This is the reason why we tested both versions on our dataset.

4.3.1 Unified Logistic Regression

Until now, we have simply defined some sets of predictors and applied the procedure described in the previous sections. From this point, we can consider the two verification vectors separately and repeat for each choice of features the entire procedure, one for each tercile. A closer look to the hypothesis function, however, reveals a specific problem in this methodology.

We temporarly focus on the first variant of the algorithm, because it is the simplest one. First, we write $h_\theta^{(\alpha)}$ explicitly as a function of $\boldsymbol{f}_{\overline{x}}$ and then, using some simple algebra, we obtain the following relationship, linear in $\overline{x}$:

$$\log\left(\frac{h_\theta^{(\alpha)}\big(\boldsymbol{X}_{(\alpha)}\big)}{\big(1 - h_\theta^{(\alpha)}(\boldsymbol{X}_{(\alpha)})\big)}\right) = \theta_0^{(\alpha)} + \theta_1^{(\alpha)}\overline{x}.$$

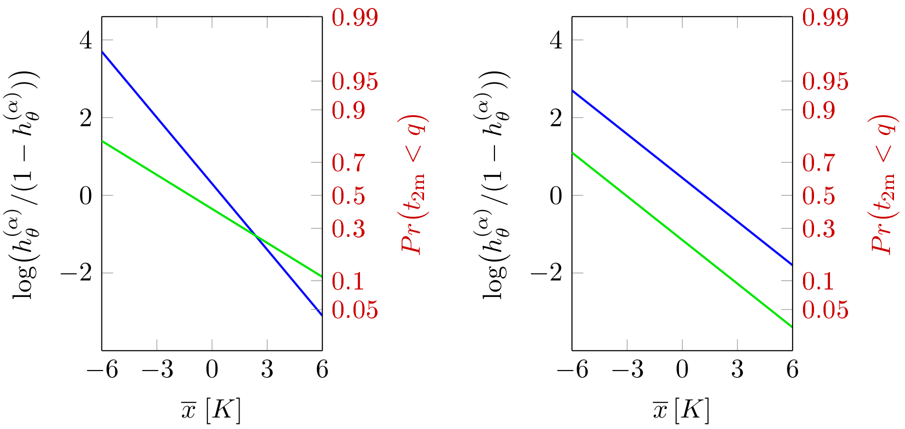
Naturally, a different set of coefficients is obtained for the first and second tercile. So, if we plot in a single graph the two straight lines corresponding to those sets of coefficients we obtain something similar to the example shown in Figure 4.8-**(a)**.

The problem arises when we want to derive probabilities from the hypothesis function. In fact, the coefficients obtained using $\boldsymbol{y}_{1/3}$ and $\boldsymbol{y}_{2/3}$ are totally indipendent from each other, so the two straigh lines can cross. This means can result in $Pr\big(t_{2\mathrm{m}} < q_{2/3}\big)$ being lower than $Pr\big(t_{2\mathrm{m}} < q_{1/3}\big)$, for some values of $\overline{x}$. This is an inconsistency: for such a relationship to be possible, we need either $Pr\big(q_{1/3} < t_{2\mathrm{m}} < q_{2/3}\big)$ to be negative (that is against the basic rules of probability) or $q_{1/3} > q_{2/3}$ (that is against their definition).

To solve this kind of problem, we follow the methodology explained in Wilks [2009]. So, we fit all the quantiles contemporarely using a procedure normally called “unified logistic regression”. We introduece a function $g(q)$ as an additional term into the hypothesis function, that become:

$$h_\theta^{(\alpha)}\big(\boldsymbol{X}_{(\alpha)}\big) = \frac{1}{1 + \exp\big(-(\boldsymbol{\theta}^{(\alpha)})^{\mathrm{T}}\boldsymbol{X}_{(\alpha)} - g(q)\big)}.$$

Naturally, this methodology can be extended for the other two variants of the



(a) Normal LR

(b) Unified LR

Figure 4.8: Comparison between the probabilities obtained for the two terciles using the standard logistic regression (panel **(a)**) and the unified version (panel **(b)**). The left y-axis of each panel shows the quantity $\log\big(h_\theta^{(\alpha)}/(1-h_\theta^{(\alpha)})\big)$, which has a linear relationship with the feature (in this case the ensemble mean $\overline{x}$, measured in Kelvin). The right y-axis, instead, shows the probabilities that the anomaly lies below the tercile, $Pr\big(t_{2\mathrm{m}} < q\big)$, obtained directly from the hypothesis function. The figure is inspired by the analogous plots from [Wilks, 2011, p. 288] and Wilks [2009].

algorithm:

$$h_\theta^{(\beta)}\big(\boldsymbol{X}_{(\beta)}\big) = \frac{1}{1+\exp\big(-(\boldsymbol{\theta}^{(\beta)})^{\mathrm{T}}\boldsymbol{X}_{(\beta)} - g(q)\big)},$$
$$h_\theta^{(\gamma)}\big(\boldsymbol{X}_{(\gamma)}\big) = \frac{1}{1+\exp\big(-(\boldsymbol{\theta}^{(\gamma)})^{\mathrm{T}}\boldsymbol{X}_{(\gamma)} - g(q)\big)}.$$

As $g(q)$ we use:

$$g(q) = \theta_q q,$$

one of the three functions proposed in Wilks [2009]. To see how this new variant solves the inconistency problem in the result, we return briefly to the simple example in Figure 4.8. Now, because all the training examples are fit simultaneously, the coefficient $\theta_{\overline{x}}^{(\alpha)}$ (that represents the slope of the curve) is the same for both the terciles. The intercept is different, because it is given by the sum of $\theta_0^{(\alpha)}$ (constant in the two cases) and $g(q)$ (that obviously varies dependig on the tercile). Therfore, the two lines are parallel and cannot cross, so there is no inconsistency in the probability for any value of $\overline{x}$. This behaviour can be seen in Figure 4.8-**(b)**. Naturally, all this discussion can be extended to the other two choices of predictands, but in those cases the feature space is two dimensional and the visualization can be a little less clear.

Finally, we can discuss how this change affects the cost function and its numerical minimization. Altough this new term has a different origin than the other components of the algorithm, we can incorporate q into the feature matrix $\boldsymbol{X}$, while the new coefficient θ_q becomes an element of the vector $\boldsymbol{\theta}$. In this way, all the formulas become formally identical to the basic LR and we can follow the same procedure explained in the previous sections.

So, we first redefine the verification vector $\boldsymbol{y}$. Note that, because the fit is done at the same time for the two terciles, we now have one single vector. Differently from the previous case it contains $2m$ elements instead of m:

$$\boldsymbol{y} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(i)} \\ \vdots \\ y^{(m)} \\ y^{(m+1)} \\ \vdots \\ y^{(m+i)} \\ \vdots \\ y^{(2m)} \end{bmatrix} \tag{4.4}$$

with:

$$y^{(i)} = \begin{cases} 1 & \text{if } t_{\text{2m}}^{(i)} < q_{1/3} \\ 0 & \text{otherwise,} \end{cases} \qquad \text{for } i = 1, ..., m \tag{4.5}$$

$$y^{(i)} = \begin{cases} 1 & \text{if } t_{\text{2m}}^{(i)} < q_{2/3} \\ 0 & \text{otherwise,} \end{cases} \qquad \text{for } i = m+1, ..., 2m \tag{4.6}$$

Then, we redefine $\boldsymbol{f}_0$, $\boldsymbol{f}_{\overline{x}}$, $\boldsymbol{f}_\sigma$ and $\boldsymbol{f}_{\overline{x}\sigma}$ duplicating their elements in order to be compatible with $\boldsymbol{y}$.

$$\boldsymbol{f}_0 = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \ \boldsymbol{f}_{\overline{x}} = \begin{bmatrix} \overline{x}^{(1)} \\ \vdots \\ \overline{x}^{(m)} \\ \overline{x}^{(1)} \\ \vdots \\ \overline{x}^{(m)} \end{bmatrix}, \ \boldsymbol{f}_\sigma = \begin{bmatrix} \sigma^{(1)} \\ \vdots \\ \sigma^{(m)} \\ \sigma^{(1)} \\ \vdots \\ \sigma^{(m)} \end{bmatrix}, \ \boldsymbol{f}_{\overline{x}\sigma} = \begin{bmatrix} \overline{x}^{(1)}\sigma^{(1)} \\ \vdots \\ \overline{x}^{(m)}\sigma^{(m)} \\ \overline{x}^{(1)}\sigma^{(1)} \\ \vdots \\ \overline{x}^{(m)}\sigma^{(m)} \end{bmatrix}. \tag{4.7}$$

In addition, we introduce the new feature vector:

$$\boldsymbol{f}_q = \left.\begin{bmatrix} q_{1/3} \\ \vdots \\ q_{1/3} \\ q_{2/3} \\ \vdots \\ q_{2/3} \end{bmatrix}\right\} 2m \text{ elements.} \tag{4.8}$$

These vectors are combined in the final three feature matrices:

$$\boldsymbol{X}_{(\alpha)} = \begin{bmatrix} — & (f_0)^{\mathrm{T}} & — \\ — & (f_{\overline{x}})^{\mathrm{T}} & — \\ — & (f_q)^{\mathrm{T}} & — \end{bmatrix}, \tag{4.9}$$

$$\boldsymbol{X}_{(\beta)} = \begin{bmatrix} — & (f_0)^{\mathrm{T}} & — \\ — & (f_{\overline{x}})^{\mathrm{T}} & — \\ — & (f_\sigma)^{\mathrm{T}} & — \\ — & (f_q)^{\mathrm{T}} & — \end{bmatrix}, \tag{4.10}$$

$$\boldsymbol{X}_{(\gamma)} = \begin{bmatrix} — & (f_0)^{\mathrm{T}} & — \\ — & (f_{\overline{x}})^{\mathrm{T}} & — \\ — & (f_{\overline{x}\sigma})^{\mathrm{T}} & — \\ — & (f_q)^{\mathrm{T}} & — \end{bmatrix}. \tag{4.11}$$

The modified coefficients vectors are:

$$\boldsymbol{\theta}^{(\alpha)} = \begin{bmatrix} \theta_0^{(\alpha)} \\ \\ \theta_{\overline{x}}^{(\alpha)} \\ \\ \theta_q^{(\alpha)} \end{bmatrix}, \quad \boldsymbol{\theta}^{(\beta)} = \begin{bmatrix} \theta_0^{(\beta)} \\ \\ \theta_{\overline{x}}^{(\beta)} \\ \\ \theta_\sigma^{(\beta)} \\ \\ \theta_q^{(\beta)} \end{bmatrix}, \quad \boldsymbol{\theta}^{(\gamma)} = \begin{bmatrix} \theta_0^{(\gamma)} \\ \\ \theta_{\overline{x}}^{(\gamma)} \\ \\ \theta_{\overline{x}\sigma}^{(\gamma)} \\ \\ \theta_q^{(\gamma)} \end{bmatrix}. \tag{4.12}$$

With this new notation, the hypothesis function is always in the form:

$$h_\theta(\boldsymbol{X}) = \frac{1}{1 + exp\big(-(\boldsymbol{\theta})^T \boldsymbol{X}\big)},$$

where $\boldsymbol{X}$ and $\boldsymbol{\theta}$ are each time substituted with one of the nine possibilities. The same is true for the cost function and its gradient, which remain as in Equation 4.2-4.3. For the regularization term we try the following values: $\lambda = 0.0, 0.001, 0.01, 0.1, 1$. The minimization algorithm is BFGS, as in the first example of this chapter.

4.3.2 Application to the re-forecasts

As we mentioned in the previous section, the notation in which the different versions of LR have been presented imply that the analysis is performed on a single grid point and forecast time. This is not our case: the dataset that we want to analyze is on a lat-lon grid of 1°resolution covering the whole globe and it is split in four forecast weeks. For this reason, we re-introduce the tensor notation from the previous chapter. In addition, in order to compare the multi-model performance with the ones obtained from its two components, we repeat the computation three times, for the multi-model and for each model. It can be useful to briefly recall all the symbols assigned to the various quantities. First of all we have $\boldsymbol{E}_{\mathrm{E}}(w,i,j,d,l)$, $\boldsymbol{E}_{\mathrm{C}}(w,i,j,d,l)$ and $\boldsymbol{E}_{\mathrm{MM}}(w,i,j,d,l)$, containing all the members of each ensemble. The ensemble mean is in $\boldsymbol{X}_{\mathrm{MM}}(w,i,j,d)$, $\boldsymbol{X}_{\mathrm{E}}(w,i,j,d)$ and $\boldsymbol{X}_{\mathrm{C}}(w,i,j,d)$. Then, there are the reanalysis, all in one tensor $\boldsymbol{Y}(w,i,j,d)$. From them we obtained first the terciles on the training set, $\boldsymbol{Y}_{1/3}(w,i,j)$ and $\boldsymbol{Y}_{2/3}(w,i,j)$, followed by the binary verification tensors $\boldsymbol{B}(w,i,j,d,0)$, $\boldsymbol{B}(w,i,j,d,1)$ and $\boldsymbol{B}(w,i,j,d,2)$, respectively for the first, second and third interval in which the distribution is split by the terciles. Recall that their element are equal to 1 when the correspondig reanalysis falls in the tercile considered, 0 otherwise.

Finally, we introduce three new tensors, containing the ensemble standard deviation. For the ECMWF-IFS and the CNR-ISAC models the procedure is rather straightforward. Starting from $\boldsymbol{E}_{\mathrm{E}}$ and $\boldsymbol{E}_{\mathrm{C}}$, the standard deviation is computed along the last dimension of these tensors, following the formulas below:

$$\boldsymbol{S}_{\mathrm{E}}(w,i,j,d) = \sqrt{\frac{1}{5}\sum_{l=0}^{4}\big(\boldsymbol{E}_{\mathrm{E}}(w,i,j,d,l) - \boldsymbol{X}_{\mathrm{E}}(w,i,j,d)\big)^2},$$

$$\boldsymbol{S}_{\mathrm{C}}(w,i,j,d) = \sqrt{\frac{1}{5}\sum_{l=0}^{4}\big(\boldsymbol{E}_{\mathrm{C}}(w,i,j,d,l) - \boldsymbol{X}_{\mathrm{C}}(w,i,j,d)\big)^2}.$$

On the contrary, for the multi-model there are different options. Starting from $\boldsymbol{E}_{\mathrm{MM}}$, we can use a formula analogous to the previous case, :

$$\boldsymbol{S}_{\mathrm{MM}}(w,i,j,d) = \sqrt{\frac{1}{25}\sum_{l=0}^{24}\big(\boldsymbol{E}_{\mathrm{MM}}(w,i,j,d,l) - \boldsymbol{X}_{\mathrm{MM}}(w,i,j,d)\big)^2}.$$

The problem is that $\boldsymbol{E}_{\mathrm{MM}}$ is an artifact created only for the DMO methods and not a real ensemble. In an operationa situation, if only LR is used for predicting probabilities, the computation of $\boldsymbol{E}_{\mathrm{MM}}$ requires additional resources. In fact, the linear regression of $\boldsymbol{X}_{\mathrm{E}}$ and $\boldsymbol{X}_{\mathrm{C}}$ gives directly $\boldsymbol{X}_{\mathrm{MM}}$, without the need of computing all the members of the multi-model.

These resources can maybe be spared using the alternative version proposed below. In this version, $\boldsymbol{S}_{\mathrm{E}}$ and $\boldsymbol{S}_{\mathrm{C}}$ are used as predictands in place of a single value representing the multi-model standard deviation. This is analogous to imposing that $\boldsymbol{S}_{\mathrm{MM}}$ is a linear combination of $\boldsymbol{S}_{\mathrm{E}}$ and $\boldsymbol{S}_{\mathrm{C}}$, with the logistic regression coefficients acting also as coefficients for this combination. Obviously, this is only a rough approximation. Its performances will be compared with the one obtained using $\boldsymbol{S}_{\mathrm{MM}}$ before deciding if it is reasonable.

4.3.3 Adjusting the notation

We use the notation presented in the previous section as basis. We introduce the dependence from the grid point (i,j) and from the week w. The first to be re-defined is the verification vector, that becomes a tensor and is given by:

$$\mathbf{Y}_{\mathrm{lr}}(w,i,j,:) = \begin{bmatrix} \boldsymbol{B}(w,i,j,0,0) \\ \vdots \\ \boldsymbol{B}(w,i,j,d,0) \\ \vdots \\ \boldsymbol{B}(w,i,j,m,0) \\ 1-\boldsymbol{B}(w,i,j,0,2) \\ \vdots \\ 1-\boldsymbol{B}(w,i,j,d,2) \\ \vdots \\ 1-\boldsymbol{B}(w,i,j,m,2) \end{bmatrix},$$

where the subscript "lr" stands for logistic regression (to distinguish it from the reanalysis tensor $\boldsymbol{Y}$). The tensor is four-dimensional, but for fixed (w,i,j) the remaining slice is a vector that behave exactly as in the algorithms already described.

The composition of $\mathbf{Y}_{\mathrm{lr}}(w,i,j,:)$ implies that we are using LR for predicting simultaneously $Pr(t_{2\mathrm{m}} < q_{1/3})$ and $Pr(t_{2\mathrm{m}} < q_{2/3})$. In fact, the first half of the vector contains the binary verification data for the first quantiles, already with the right values (equal to 1 when $\boldsymbol{Y}(w,i,j,d) < \mathbf{Y}_{1/3}(w,i,j)$). Instead, the elements of the second half are equal to 1 when $Y(w,i,j,d) < \mathbf{Y}_{2/3}(w,i,j)$. $\boldsymbol{B}(w,i,j,m,2) = 1$ for the opposit inequality, that is $Y(w,i,j,d) > \mathbf{Y}_{2/3}(w,i,j)$. So, in $\mathbf{Y}_{\mathrm{lr}}(w,i,j,d)$ $(d = m+1,...,2m)$, we insert the difference between 1 and $\boldsymbol{B}(w,i,j,m,2)$, which gives us the wanted values.

For the feature vectors, we have to distinguish between the two initial models and the multimodel. Three different versions of LR are tested for the ECMWF-IFS and the CNR-ISAC models. They are simply the application of Equation 4.4-4.12 substituing $\overline{x}$ with $\boldsymbol{X}_{\mathrm{E}}(w,i,j,d)$ or $\boldsymbol{X}_{\mathrm{C}}(w,i,j,d)$ and σ with $\boldsymbol{S}_{\mathrm{E}}$ or $\boldsymbol{S}_{\mathrm{C}}$, depending on the model considered. For the multimodel, in addition to these tests, we analize also the case in which $\boldsymbol{S}_{\mathrm{E}}$ and $\boldsymbol{S}_{\mathrm{C}}$ are used in place of $\boldsymbol{S}_{\mathrm{MM}}$.

Naturally, due to the spatial and temporal dependence, all these vectors become tensors, which in turn will be combined in tensors with an higher dimesnionality that will serve as input for the hypothesis function. To remove any ambiguity, we write them explicitly. Note that, as for $\mathbf{Y}_{\mathrm{lr}}$, for fixed (w,i,j) the remaining slice is a vector as in Equation 4.7. The following ones refer to the multi-model, but the ECMWF-IFS and the CNR-ISAC cases can be obtained simply by substituing the subscrit “MM” with “E” or “C”:

$$\boldsymbol{F}_{\mathrm{MM},\overline{x}}(w,i,j,:) = \begin{bmatrix} \boldsymbol{X}_{\mathrm{MM}}(w,i,j,1) \\ \vdots \\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,d) \\ \vdots \\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,m) \\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,1) \\ \vdots \\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,d) \\ \vdots \\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,m) \end{bmatrix}, \quad \boldsymbol{F}_{\mathrm{MM},\sigma}(w,i,j,:) = \begin{bmatrix} \boldsymbol{S}_{\mathrm{MM}}(w,i,j,1) \\ \vdots \\ \boldsymbol{S}_{\mathrm{MM}}(w,i,j,d) \\ \vdots \\ \boldsymbol{S}_{\mathrm{MM}}(w,i,j,m) \\ \boldsymbol{S}_{\mathrm{MM}}(w,i,j,1) \\ \vdots \\ \boldsymbol{S}_{\mathrm{MM}}(w,i,j,d) \\ \vdots \\ \boldsymbol{S}_{\mathrm{MM}}(w,i,j,m) \end{bmatrix},$$

$$\boldsymbol{F}_{\mathrm{MM},\overline{x}\sigma}(w,i,j,:) = \begin{bmatrix} \boldsymbol{X}_{\mathrm{MM}}(w,i,j,1)\boldsymbol{S}_{\mathrm{MM}}(w,i,j,1) \\ \vdots \\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,d)\boldsymbol{S}_{\mathrm{MM}}(w,i,j,d) \\ \vdots \\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,m)\boldsymbol{S}_{\mathrm{MM}}(w,i,j,m) \\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,1)\boldsymbol{S}_{\mathrm{MM}}(w,i,j,1) \\ \vdots \\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,d)\boldsymbol{S}_{\mathrm{MM}}(w,i,j,d) \\ \vdots \\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,m)\boldsymbol{S}_{\mathrm{MM}}(w,i,j,m) \end{bmatrix}.$$

The feature tensor containing the terciles is the same for the three models:

$$\boldsymbol{F}_{q}(w,i,j,:) = \begin{bmatrix} \boldsymbol{Y}_{1/3}(w,i,j) \\ \vdots \\ \boldsymbol{Y}_{1/3}(w,i,j) \\ \boldsymbol{Y}_{2/3}(w,i,j) \\ \vdots \\ \boldsymbol{Y}_{2/3}(w,i,j) \end{bmatrix}$$

Also the bias unit ($\boldsymbol{F}_0(w,i,j,:)$) remains constant and its elements are always equal to 1, as in Equation 4.7. Naturally the slices $\boldsymbol{F}_0(w,i,j,:)$ and $\boldsymbol{F}_a(w,i,j,:)$ are vectors of length $2m$. Finally, we introduce two tensor that combine the multimodel mean with the standard deviation of the single models, they are useful for the approximation previously described:

$$\boldsymbol{F}_{\mathrm{MM},\overline{x}\sigma-E}(w,i,j,:)=\begin{bmatrix}\boldsymbol{X}_{\mathrm{MM}}(w,i,j,1)\boldsymbol{S}_{\mathrm{E}}(w,i,j,1)\\ \vdots\\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,d)\boldsymbol{S}_{\mathrm{E}}(w,i,j,d)\\ \vdots\\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,m)\boldsymbol{S}_{\mathrm{E}}(w,i,j,m)\\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,1)\boldsymbol{S}_{\mathrm{E}}(w,i,j,1)\\ \vdots\\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,d)\boldsymbol{S}_{\mathrm{E}}(w,i,j,d)\\ \vdots\\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,m)\boldsymbol{S}_{\mathrm{E}}(w,i,j,m)\end{bmatrix}.$$

$$\boldsymbol{F}_{\mathrm{MM},\overline{x}\sigma-C}(w,i,j,:)=\begin{bmatrix}\boldsymbol{X}_{\mathrm{MM}}(w,i,j,1)\boldsymbol{S}_{\mathrm{C}}(w,i,j,1)\\ \vdots\\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,d)\boldsymbol{S}_{\mathrm{C}}(w,i,j,d)\\ \vdots\\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,m)\boldsymbol{S}_{\mathrm{C}}(w,i,j,m)\\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,1)\boldsymbol{S}_{\mathrm{C}}(w,i,j,1)\\ \vdots\\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,d)\boldsymbol{S}_{\mathrm{C}}(w,i,j,d)\\ \vdots\\ \boldsymbol{X}_{\mathrm{MM}}(w,i,j,m)\boldsymbol{S}_{\mathrm{C}}(w,i,j,m)\end{bmatrix}.$$

Using the newly defined tensors, the feature matrices that were presented in Equation 4.11 become tensors, which dimension is higher by one unity with respect to the ones just re-defined. This arises because we are now constructing sets of features, while the tensors above contained each a single feature. This is analogous to what happened in the standard algorithm, the only difference is the dimensionality of the dataset, not in the underlying concept. Note that, as before, each of the quantity below defines a variant of the algorithm. The three basic version, tested for both the initial model and the multi-model, are:

$$\boldsymbol{X}_{\mathrm{MM},(\alpha)}\left(w,i,j,:,:\right)=\begin{bmatrix}- & \left(\boldsymbol{F}_0\left(w,i,j,:\right)\right)^{\mathrm{T}} & -\\ - & \left(\boldsymbol{F}_{\mathrm{MM},\overline{x}}\left(w,i,j,:\right)\right)^{\mathrm{T}} & -\\ - & \left(\boldsymbol{F}_q\left(w,i,j,:\right)\right)^{\mathrm{T}} & -\end{bmatrix}, \tag{4.13}$$

$$\boldsymbol{X}_{\mathrm{MM},(\beta)}\left(w,i,j,:,:\right)=\begin{bmatrix}- & \left(\boldsymbol{F}_0\left(w,i,j,:\right)\right)^{\mathrm{T}} & -\\ - & \left(\boldsymbol{F}_{\mathrm{MM},\overline{x}}\left(w,i,j,:\right)\right)^{\mathrm{T}} & -\\ - & \left(\boldsymbol{F}_{\mathrm{MM},\sigma}\left(w,i,j,:\right)\right)^{\mathrm{T}} & -\\ - & \left(\boldsymbol{F}_q\left(w,i,j,:\right)\right)^{\mathrm{T}} & -\end{bmatrix}, \tag{4.14}$$

$$\boldsymbol{X}_{\mathrm{MM},(\gamma)}\left(w,i,j,:,:\right)=\begin{bmatrix}- & \left(\boldsymbol{F}_0\left(w,i,j,:\right)\right)^{\mathrm{T}} & -\\ - & \left(\boldsymbol{F}_{\mathrm{MM},\overline{x}}\left(w,i,j,:\right)\right)^{\mathrm{T}} & -\\ - & \left(\boldsymbol{F}_{\mathrm{MM},\overline{x}\sigma}\left(w,i,j,:\right)\right)^{\mathrm{T}} & -\\ - & \left(\boldsymbol{F}_q\left(w,i,j,:\right)\right)^{\mathrm{T}} & -\end{bmatrix}. \tag{4.15}$$

The ones here presented are the one referring to the multi-model, but, as always, substituing "MM" with "E" or "C" give us the version for the other two models.

The two additional variants, applied for the sole multi-model, are:

$$\boldsymbol{X}_{\mathrm{MM},(\beta-EC)}\left(w,i,j,:,:\right)=\begin{bmatrix} — & \left(\boldsymbol{F}_0\left(w,i,j,:\right)\right)^{\mathrm{T}} & — \\ — & \left(\boldsymbol{F}_{\mathrm{MM},\overline{x}}\left(w,i,j,:\right)\right)^{\mathrm{T}} & — \\ — & \left(\boldsymbol{F}_{E,\sigma}\left(w,i,j,:\right)\right)^{\mathrm{T}} & — \\ — & \left(\boldsymbol{F}_{C,\sigma}\left(w,i,j,:\right)\right)^{\mathrm{T}} & — \\ — & \left(\boldsymbol{F}_q\left(w,i,j,:\right)\right)^{\mathrm{T}} & — \end{bmatrix}, \tag{4.16}$$

$$\boldsymbol{X}_{\mathrm{MM},(\gamma-EC)}\left(w,i,j,:,:\right)=\begin{bmatrix} — & \left(\boldsymbol{F}_0\left(w,i,j,:\right)\right)^{\mathrm{T}} & — \\ — & \left(\boldsymbol{F}_{\mathrm{MM},\overline{x}}\left(w,i,j,:\right)\right)^{\mathrm{T}} & — \\ — & \left(\boldsymbol{F}_{E,\overline{x}\sigma}\left(w,i,j,:\right)\right)^{\mathrm{T}} & — \\ — & \left(\boldsymbol{F}_{C,\overline{x}\sigma}\left(w,i,j,:\right)\right)^{\mathrm{T}} & — \\ — & \left(\boldsymbol{F}_q\left(w,i,j,:\right)\right)^{\mathrm{T}} & — \end{bmatrix}, \tag{4.17}$$

To avoid confusion, we explicitally define the order of the index in the slices as the first referring to the rows of the matrix and the second to the columns. So, for example $\boldsymbol{X}_{\mathrm{MM},(\alpha)}\left(w,i,j,0,:\right)$ is a row vector referring to the feature having index 0:

$$\boldsymbol{X}_{\mathrm{MM},(\alpha)}\left(w,i,j,0,:\right)=\begin{bmatrix} — & \left(\boldsymbol{F}_0\left(w,i,j,:\right)\right)^{\mathrm{T}} & — \end{bmatrix},$$

while $\boldsymbol{X}_{\mathrm{MM},(\alpha)}\left(w,i,j,:,d\right)$ is a column vector, containing the example d-th for all the feature used in that specific version of the algorithm:

$$\boldsymbol{X}_{\mathrm{MM},(\alpha)}\left(w,i,j,:,d\right)=\begin{bmatrix} \left(\boldsymbol{F}_0\left(w,i,j,d\right)\right)^{\mathrm{T}} \\ \left(\boldsymbol{F}_{\mathrm{MM},\overline{x}}\left(w,i,j,d\right)\right)^{\mathrm{T}} \\ \left(\boldsymbol{F}_q\left(w,i,j,d\right)\right)^{\mathrm{T}} \end{bmatrix}.$$

The next step is the re-definition of the coefficients. For the three standard algorithms they are:

$$\boldsymbol{\Theta}_{\mathrm{MM},(\alpha)}\left(w,i,j,:\right)=\begin{bmatrix} \left(\boldsymbol{\Theta}^{(\alpha)}_{\mathrm{MM},0}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\alpha)}_{\mathrm{MM},\overline{x}}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\alpha)}_{\mathrm{MM},q}\left(w,i,j\right)\right) \end{bmatrix},$$

$$\boldsymbol{\Theta}_{\mathrm{MM},(\beta)}\left(w,i,j,:\right)=\begin{bmatrix} \left(\boldsymbol{\Theta}^{(\beta)}_{\mathrm{MM},0}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\beta)}_{\mathrm{MM},\overline{x}}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\beta)}_{\mathrm{MM},\sigma}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\beta)}_{\mathrm{MM},q}\left(w,i,j\right)\right) \end{bmatrix},$$

$$\boldsymbol{\Theta}_{\mathrm{MM},(\gamma)}\left(w,i,j,:\right)=\begin{bmatrix} \left(\boldsymbol{\Theta}^{(\gamma)}_{\mathrm{MM},0}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\gamma)}_{\mathrm{MM},\overline{x}}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\gamma)}_{\mathrm{MM},\overline{x}\sigma}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\gamma)}_{\mathrm{MM},q}\left(w,i,j\right)\right) \end{bmatrix}.$$

In addition, the two variants for the multi-model are given by:

$$\boldsymbol{\Theta}_{\mathrm{MM},(\beta-EC)}\left(w,i,j,:\right) = \begin{bmatrix} \left(\boldsymbol{\Theta}^{(\beta-EC)}_{\mathrm{MM},0}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\beta-EC)}_{\mathrm{MM},\overline{x}}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\beta-EC)}_{E,\sigma}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\beta-EC)}_{C,\sigma}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\beta-EC)}_{\mathrm{MM},q}\left(w,i,j\right)\right) \end{bmatrix},$$

$$\boldsymbol{\Theta}_{\mathrm{MM},(\gamma-EC)}\left(w,i,j,:\right) = \begin{bmatrix} \left(\boldsymbol{\Theta}^{(\gamma-EC)}_{\mathrm{MM},0}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\gamma-EC)}_{\mathrm{MM},\overline{x}}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\gamma-EC)}_{E,\overline{x}\sigma}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\gamma-EC)}_{C,\overline{x}\sigma}\left(w,i,j\right)\right) \\ \left(\boldsymbol{\Theta}^{(\gamma-EC)}_{\mathrm{MM},q}\left(w,i,j\right)\right) \end{bmatrix}.$$

From this point, we proceed with a loop over all tripletw (w,i,j), for each of them we follow the same steps exposed when treating the regularized version of LR. For simplicity, we will omit the subscripts referring to which algorithm is actually used. All the procedure described can be applied to each of the choice of features and coefficients presented just by adding the corresponding subscript. We first write explicitly the new hypothesis function:

$$h\big(\boldsymbol{X}(w,i,j,:,d);\boldsymbol{\Theta}(w,i,j,:)\big) = \frac{1}{1+\exp\big(-(\boldsymbol{\Theta}(w,i,j,:))^T\boldsymbol{X}(w,i,j,:,d)\big)}.$$

The decision of moving the parameters from the subscript into the parenthesis is for typographical reason (the multiple indexes in the subscript would not have been easily readable), the semicolon underlines the role of $\boldsymbol{\Theta}$ as a parameter and not as a variable. Then, given a vector $\boldsymbol{\lambda}$ whose lenght is equal to the number of featueres n_f (that is also the number of rows in the tensors from Equation 4.13-4.17), the cost function is given by:

$$\begin{aligned} J(\boldsymbol{\Theta}(w,i,j,:)) = \Bigg(\frac{1}{2m}\sum_{d=1}^{2m}\Big(&\boldsymbol{Y}_{\mathrm{lr}}(w,i,j,d)\log\big(h(\boldsymbol{X}(w,i,j,:,d);\boldsymbol{\Theta}(w,i,j,:))\big)+ \\ &+(1-\boldsymbol{Y}_{\mathrm{lr}}(w,i,j,d))\log\big(1-h(\boldsymbol{X}(w,i,j,:,d);\boldsymbol{\Theta}(w,i,j,:))\big)\Big)\Bigg)+ \\ &+\frac{1}{4m}\sum_{f=0}^{n_f}\boldsymbol{\lambda}(f)(\boldsymbol{\Theta}(w,i,j,f))^2, \end{aligned}$$

and the new gradient is given by:

$$\begin{aligned} &\frac{\partial J(\boldsymbol{\Theta}(w,i,j,:)))}{\partial\boldsymbol{\Theta}(w,i,j,f)} = \\ &= \Bigg(\frac{1}{2m}\sum_{i=1}^{2m}\big(h(\boldsymbol{X}(w,i,j,:,d);\boldsymbol{\Theta}(w,i,j,:)) - \boldsymbol{Y}_{\mathrm{lr}}(w,i,j,d))\big)\boldsymbol{X}(w,i,j,f,d)\Bigg)+ \\ &\qquad + \frac{1}{2m}\boldsymbol{\lambda}(f)\boldsymbol{\Theta}(w,i,j,f). \end{aligned}$$

Finally, using BFGS we find the values $\boldsymbol{\Theta}^*(w,i,j,f)$ that minimize the cost function.

These coefficient are used for computing probabilities. Using a notation similar to the DMO case, we define three new tensors. The first contain the probability that the 2-metre temperature anomaly is below the first tercile:

$$\boldsymbol{P}^{LR}_{inf}(w,i,j,d) = h(\boldsymbol{X}(w,i,j,:,d), \boldsymbol{\Theta}(w,i,j,:)),$$

with $\boldsymbol{X}(w,i,j,:,d)$ containing $\boldsymbol{F}_a(w,i,j,:) = \boldsymbol{Y}_{1/3}(w,i,j)$. This condition is due to our choice of unified variant of LR and correspond, on the training set, to $d = 1,...,m$.

The second contains the probability for the same variable being in the region between the two terciles:

$$\boldsymbol{P}^{LR}_{mid}(w,i,j,d) = h(\boldsymbol{X}(w,i,j,:,d_1), \boldsymbol{\Theta}(w,i,j,:)) - \boldsymbol{P}^{LR}_{inf}(w,i,j,d),$$

with $\boldsymbol{X}(w,i,j,:,d)$ containing $\boldsymbol{F}_a(w,i,j,:) = \boldsymbol{Y}_{2/3}(w,i,j)$. On the trainig set, this is the same as imposing $d = m+1,...,2m$.

Finally, the probability for the region above the secon tercile,

$$\boldsymbol{P}^{LR}_{sup}(w,i,j,d) = 1 - \boldsymbol{P}^{LR}_{mid}(w,i,j,d) - \boldsymbol{P}^{LR}_{inf}(w,i,j,d).$$

This definition assure that the sum of the probabilities on all the region of the distribution is equal to one. Naturally, $\boldsymbol{P}^{LR}_{sup}(w,i,j,d)$ can be computed without using the other two probabilities, starting directly from the hypothesis function. In this case,

$$\boldsymbol{P}^{LR}_{sup}(w,i,j,d) = 1 - h(\boldsymbol{X}(w,i,j,:,d_1), \boldsymbol{\Theta}(w,i,j,:)),$$

if $\boldsymbol{F}_a(w,i,j,:) = \boldsymbol{Y}_{2/3}(w,i,j)$ in the feature tensor.

4.4 Results

Many variant of LR were introduced in the previous section: there are 3 different choices of predictands for the two models and 5 for the multimodel. In addition, each of them has to be tested for all the five values of λ. If we account also for the cross-validation methodology, that imposes the application of the same procedure on $k = 18$ different training sets (if we split the dataset in winters, as in the previous chapters), the number of times we have to perform the computation is 990. If this number does not seem high enough, recall the structure of the dataset: to each of the 4 week and 65,160 grid points correspond an indipendent LR, with its set of coefficient and, therefore, a separate call of the minimization routine. If we count how many times this numerical minimization is performed, we obtain the number 258,033,600. It is difficult to provide a precise estimation of the time required by one of those operation, because it depends on various factors as the number of coefficients, how close to the final value is their random initial guess and the dimension of the dataset. However, repeating the procedure a certain number of time gives us the order of magnitude for this computational time: we obtain values close to 4ms with the simplest versions of LR (that we use as a lower boundary) while an higher degree of complexity result in more than double this time. Even using the most optimistic estimate, the total time required for performing all the minimization is approximately 12 days of uninterrupted computation[3]. In addition, this value does not account for all the preliminary operations, like loading the tensors, adjusting their shape, defining and filling of the temporary matrices. Therefore, we decide to reduce this time by introducing some modification to the procedure: for a quicker selection of the most promising variant, we perform the analysis following a slightly reduced version of cross-validation. We keep the division in winters introduced in the previous chapters but instead of trying all the 18 combinations, we use only a subset containing six of them. They have been chosen by extracting a random validation winter each three consecutive ones. The resulting validation sets are the ones correspinding to the winters: 1993-1994, 1997-1998, 1998-1999, 2002-2003, 2005-2006, 2008-2009. From this procedure we select simultaneously $\boldsymbol{\lambda}$ and the set of predictands. Then, on the most promising variant, the complete cross validation procedure is performed, from which we obtain our final estimate of the probablistic scores for logistic regression.

Naturally, using a smaller number of validation sets for the comparison can result in the output being more sensible to external fluctuations. For example, if a particular winter is exeptionally predictable, the resulting scores will be particularly good. Because the final value is obtained through an average on all the validation sets, the effect of this outlayer grows as the number of winters decreases. However, in our case the reduced dataset contains a considerable number of dates: the six winters randomly selected contain a total of 88 dates[4]. Nevertheless, we will compare, at the end of the analysis, the perfomances of the most promising algorithm on the whole set of winters with the scores obtained on the reduced ones. In this way, if for some unlikely chances the selected dates are extremely predictable or unpredictable, we will probably notice significant differences between the two values. In conlcusion, performing a complete analysis on every one of the variants can be an unwise management of the available resources: excluding the least promising techniques on a smaller dataset and keeping some parameter fixed can spare us some resources, which can be later applyed to broaden the set of algorithms tested.

4.4.1 Algorithm selection

In this subsection we focus on the selection of the most promising variant of logistic regression. We choose contemporarly the regularization parameter and the set of features, by comparing the ranked probability skill score[5] (RPSS) averaged over the whole globe. Values for each of the four regions previously described (Northern Emisphere, Southern Emisphere, Equatorial Belt and Europe) have also been computed, but the trend is very similar to the global mean and the information was therefore redundant. Naturally, other scores can be used for the comaparison. We computed both RPS and DRPSS, but again they did not provide a significant amount of additional information, and we exluded them from the analysis.

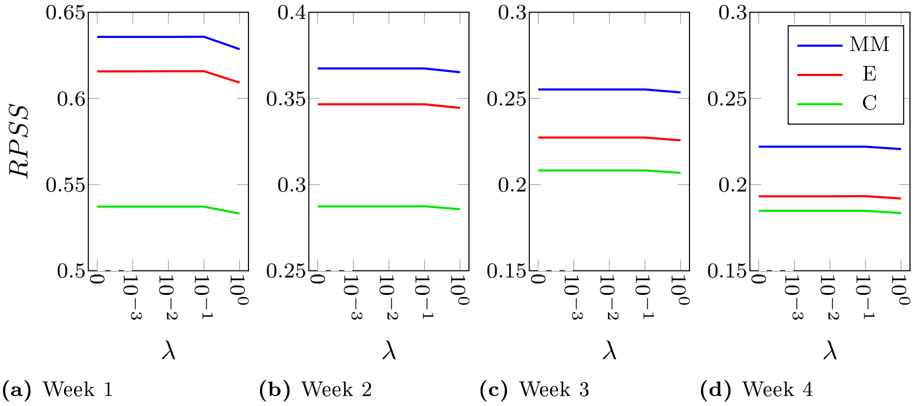
Due to the high number of values that have to be compared, we first show the

[3] On a "Intel(R) Xeon(R) CPU E5-2643 0 3.30GHz" CPU.

[4] The exact number of dates in each winter can be found in the second chapter.

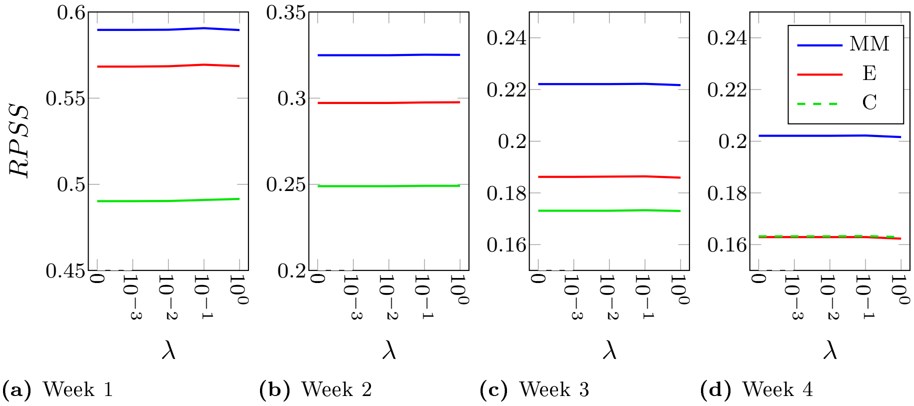
[5] The defintion together with a brief description of the score can be found at the end of the previous chapter.

selection of λ for each variant, through some plots of the RPSS as a function of λ. Then, we compare the different sets of features, each with the chosen regularization. Naturally, this division is only a way to simplify the exposition of the results. In reality, the choice of predictors and λ are done simultaneously, but presenting in a single plot an excessively high number of curves makes the figure unreadable, therefore useless.



(a) Week 1 **(b)** Week 2 **(c)** Week 3 **(d)** Week 4

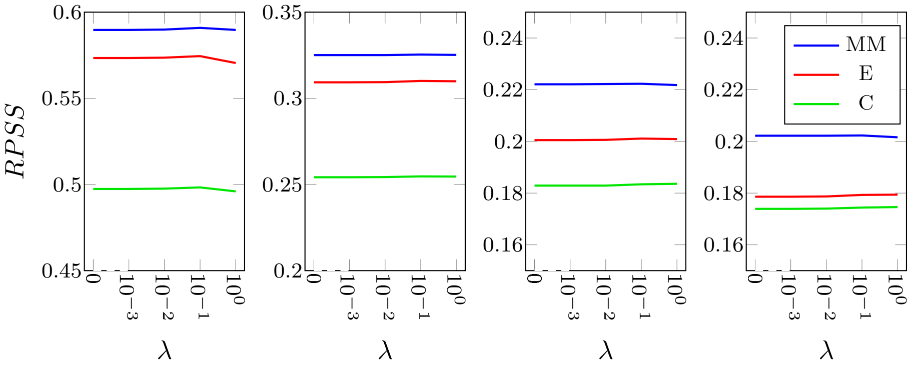
Figure 4.9: Ranked probability skill score as a function of the regularization parameter λ. The variant considered is denoted by α and uses only the ensemble mean as a predictor. Each panel refer to a different forecast week, as suggested by the label. The blue curve shows the values obtained by applying the algorithm to the multi-model, the red one refers to the ECMWF-IFS model and the green one to the CNR-ISAC model. The legend is shown only n the last plot, but it is valid for all the panels.



(a) Week 1 **(b)** Week 2 **(c)** Week 3 **(d)** Week 4

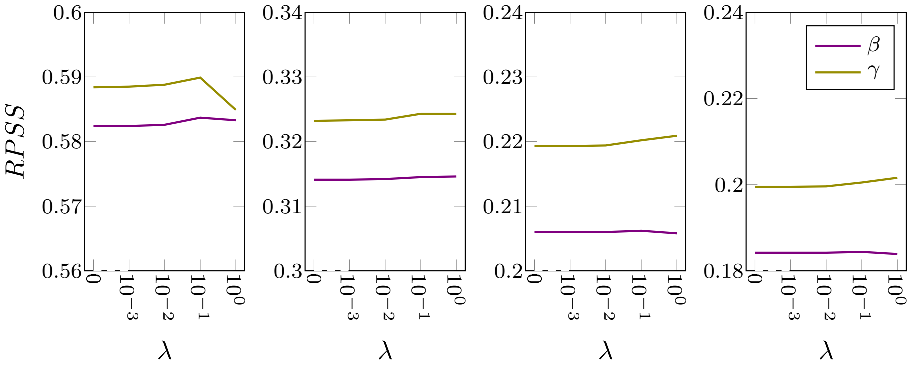
Figure 4.10: As in Figure 4.9, but for the variant β, that is the one using both ensemble mean and standard devation.

So, we start by analyzing the behaviour of the set of features $\boldsymbol{X}_{(\alpha)}$, shown in Figure 4.9, where the curve for each model is shown in a different color: blue for the multi-model, red for the ECMWF-IFS model and green for the CNR-ISAC one. Substantially, we see a nearly total absence of dipendence of the



(a) Week 1 **(b)** Week 2 **(c)** Week 3 **(d)** Week 4

Figure 4.11: As in Figure 4.9, but for the variant γ, that is the one using the ensemble mean as first predictor and the second is the product of the ensemble eman with the standard deviation.



(a) Week 1 **(b)** Week 2 **(c)** Week 3 **(d)** Week 4

Figure 4.12: Ranked probability skill score for the variants of LR using a linear combination of the standard deviation of the single models instead that the one computed directly from its members. In the legend, β is an abbreviation for $\beta - EC$, that is the subscript for the feature and coefficient tensors of the algorithms, while γ stands for $\gamma - EC$.

result from λ. Only for the largest choice of regularization we start to see a loss in performances, due to λ being too high. So, the algorithm is probably not affected by overfitting in its basic form and we can omit the regularization for this choice of predictors. Note that the choice of λ is dependant on the dimension of the set: so, before using the same algorithm on smaller set, a good practice would be to repeat this kind of analysis before deciding to avoid using the regularization term.

In Figure 4.10 we compare the RPSS for $\boldsymbol{X}_{(\beta)}$, again showing the different models simultaneously and keeping the choice of colors associated to each of them. Again, the plot reveals a lack of dependece from λ, so we decide to omit the regularization term when comparing the algorithms in the next analysis. The same behaviour can be seen also in Figure 4.11, where the set of features is $\boldsymbol{X}_{(\gamma)}$, and as a consequence we keep $\lambda = 0$ also in this case.

Finally, in Figure 4.12, where the two variants for the multi-model ensemble standard deviation are used as input features, we see a noiser curve, but the fluctuations are on the third decimal place and we do not consider this variation enough for chosing to use regularization.

In conclusion, in all the cases studied there was no need of using λ to avoid overfitting. Probably this is due to the simplicity of the algorithms, combined to the dimension of the dataseet that was enough for training adequately the logistic regression.

Before proceeding with the comparison of the variants, we compare separately the different choices possible for the multi-model standard deviation, in order to use a single value for the algorithms with the subscripts β and γ when considering the multi-model. This comparison is shown in Table 4.1. We see immediately that there is a marginal loss in performances in the algorithms using the linear combination of the satndard deviation of the single model instead of the one computed starting from the 25 multi-model memebers. However, we do not know if the small entity of this decrease is due to the two predictors being sufficiently similar in terms of useful information, or if this derives from the standard deviatio itself not being an useful predictor in our case. As can be deduced from the plot shown before, the algorithms using have worst performances compared to the ones using only the ensemble mean. We will discuss this behaviour when comparing the three variants, but for now we notice that in this situation we can not confidently assume that the approximate standard deviation in the $\beta - EC$ and $\gamma - EC$ algorithms is a good approximation. So, in comparing the algorithms we use the other version, that is the one computed starting from the multi-model members, denoted by the subscripts β and γ.

Finally, we show in Table 4.2 the comparison between the three choices of features, $\boldsymbol{X}_{(\alpha)}$, $\boldsymbol{X}_{(\beta)}$ and $\boldsymbol{X}_{(\gamma)}$. Clearly, the simplest version (α) outperforms the others all weeks. So, we deduce that in our case the standard deviation does not introduce useful information and, on the contrary, leads to a deterioration of the performances. Probably, the algorithm during the learning phase assigns some coefficients to the feature containing the standard-deviation, which does not really correspond to a real prediction rule and therefore does not generalize well on the validation set. We cannot deduce from this limited set of dates and test if the standard deviation really does not bring useful information to the prediction or if this lack of good performances is due to the dimension of the training set that is not enough for the algorithm.

From the comparison, we chose the tensor $X_{(\alpha)}$ as the definitive version of

w	$\boldsymbol{RPSS}^{(\mathrm{ALL})}_{\mathrm{MM},(\beta)}$	$\boldsymbol{RPSS}^{(\mathrm{ALL})}_{\mathrm{MM},(\beta-EC)}$	$\boldsymbol{RPSS}^{(\mathrm{ALL})}_{\mathrm{MM},(\gamma)}$	$\boldsymbol{RPSS}^{(\mathrm{ALL})}_{\mathrm{MM},(\gamma-EC)}$
1	0.59	0.58	0.59	0.59
2	0.32	0.31	0.33	0.32
3	0.22	0.21	0.22	0.22
4	0.20	0.18	0.20	0.20

Table 4.1: Ranked probability skill score averaged over the whole globe for the different choice of multi-model standard deviation. The first column shows, in blue, the week to which each row of the table refers. The remaining columns shows the values for the four variants: the first shows the results for the feature tensor $\boldsymbol{X}_{\mathrm{MM},(\beta)}$, the second $\boldsymbol{X}_{\mathrm{MM},(\beta-EC)}$, the third $\boldsymbol{X}_{\mathrm{MM},(\gamma)}$ and the last $\boldsymbol{X}_{\mathrm{MM},(\gamma-EC)}$.

LR for our analysis, and in the following section the detailed results obtained from the complete cross validation technique refer to this variant.

4.4.2 Complete cross validation analysis

In this section we perform an analysis similar to the one presented at the end of the pevious chapter. As decided during the alogorithm selection, the variant analyzed is the one using the sole ensemble mean as a predictor (denoted by the subscript α) and with the regression parameter set to zero. We begin by showing in Table 4.3 the values of Ranked probability skill scores averaged over the usual four regions, this is followed by the DRPSS in Table 4.4.

As for the DMO methods, the multi-model outperforms almost always the other two models. However, the difference is less marked than in the previous case, with the ECMWF-IFS model rather close to the multi-model performances.

Looking at the dependence of the score from the forecast time, we can again distinguish between the behaviour in the first couple of weeks from the remaining two. Expecially over the two hemishperes and over Europe, we see a sharp decline in performances from the first week to the second followed by another noticeable decrease in the third, while the scores remain almost constant between the last two. Over the equatorial belt the first week is less predictable than in the other regions, but the values over the remaining weeks are particularly high, if compared to their counterpart for the hemispheres or Europe. Regarding Europe, we want to emphasize the extremely low skill that all the algorithms obtain over this region in the last two weeks. All the behaviours are similar to the ones observed for DMO techniques, altought the values are rather different, expecially for the longer time ranges.

As in the previous chapter, we also use reliability diagrams[6] for verifying LR outputs. Naturally, the two terciles need to be analyzed separately because this tool can deal only with binary predictions. We keep the same probability thresholds $p^{(c)}$ defined in the previous chapter, for comparison purpouses. The resulting plots are shown in Figure 4.13 (for the lower tercile) and Figure 4.14 (for the upper one).

The first week, as expected, is the one where the calibration fuction is nearer

[6] A detailed description of the tool is presented in the verification section of previous chapter.

w	$\boldsymbol{RPSS}_{\mathrm{MM},(\alpha)}$	$\boldsymbol{RPSS}_{\mathrm{MM},(\beta)}$	$\boldsymbol{RPSS}_{\mathrm{MM},(\gamma)}$
1	0.64	0.59	0.59
2	0.36	0.32	0.33
3	0.25	0.22	0.22
4	0.23	0.20	0.20

(a) Multi-model

w	$\boldsymbol{RPSS}_{\mathrm{E},(\alpha)}$	$\boldsymbol{RPSS}_{\mathrm{E},(\beta)}$	$\boldsymbol{RPSS}_{\mathrm{E},(\gamma)}$
1	0.62	0.57	0.57
2	0.34	0.30	0.31
3	0.23	0.19	0.20
4	0.20	0.16	0.18

(b) ECMWF-IFS

w	$\boldsymbol{RPSS}_{\mathrm{C},(\alpha)}$	$\boldsymbol{RPSS}_{\mathrm{C},(\beta)}$	$\boldsymbol{RPSS}_{\mathrm{C},(\gamma)}$
1	0.54	0.49	0.50
2	0.28	0.25	0.25
3	0.21	0.17	0.18
4	0.19	0.16	0.17

(c) CNR-ISAC

Table 4.2: Comparison of the ranked probability skill score for the three variants of LR, with the regularization parameter $\lambda = 0$. The three tables refer to the different models, as suggested by the label below. The first column, as always, show the week for each of the rows. The values corresponding to the best performances have been highlited in red.

w	$\boldsymbol{RPSS}_{\mathrm{MM},(\alpha)}$	$\boldsymbol{RPSS}_{\mathrm{E},(\alpha)}$	$\boldsymbol{RPSS}_{\mathrm{C},(\alpha)}$
1	0.68	0.67	0.59
2	0.34	0.32	0.26
3	0.20	0.17	0.15
4	0.17	0.14	0.14

(a) Northern Hemisphere

w	$\boldsymbol{RPSS}_{\mathrm{MM},(\alpha)}$	$\boldsymbol{RPSS}_{\mathrm{E},(\alpha)}$	$\boldsymbol{RPSS}_{\mathrm{C},(\alpha)}$
1	0.68	0.67	0.59
2	0.37	0.36	0.29
3	0.25	0.22	0.20
4	0.20	0.18	0.17

(b) Southern Hemishpere

w	$\boldsymbol{RPSS}_{\mathrm{MM},(\alpha)}$	$\boldsymbol{RPSS}_{\mathrm{E},(\alpha)}$	$\boldsymbol{RPSS}_{\mathrm{C},(\alpha)}$
1	0.55	0.52	0.44
2	0.39	0.36	0.31
3	0.32	0.29	0.27
4	0.30	0.26	0.25

(c) Equatorial Belt

w	$\boldsymbol{RPSS}_{\mathrm{MM},(\alpha)}$	$\boldsymbol{RPSS}_{\mathrm{E},(\alpha)}$	$\boldsymbol{RPSS}_{\mathrm{C},(\alpha)}$
1	0.69	0.68	0.60
2	0.27	0.25	0.20
3	0.11	0.10	0.09
4	0.10	0.08	0.08

(d) Europe

Table 4.3: Ranked Probability Skill Score (RPSS) for the variant α of non-regularized logistic regression (LR), averaged over the 18 validation winters. The four table present the spatial average over the four different regions defined in the second chapter. The first column always shows, in blue, the week. Each of the remaining column refer to a different model: in the first one there are the values for the multi-model ($\overline{\boldsymbol{RPSS}}_{\mathrm{MM},(\alpha)}$), in the second the ECMWF-IFS ones ($\overline{\boldsymbol{RPSS}}_{\mathrm{E},(\alpha)}$) and in the third the CNR-ISAC ones ($\overline{\boldsymbol{RPSS}}_{\mathrm{C},(\alpha)}$). The value corresponding to the best performances for each row is highlighted in red.

w	$\boldsymbol{DRPSS}_{\mathrm{MM},(\alpha)}$	$\boldsymbol{DRPSS}_{\mathrm{E},(\alpha)}$	$\boldsymbol{DRPSS}_{\mathrm{C},(\alpha)}$
1	0.71	0.70	0.62
2	0.40	0.38	0.33
3	0.27	0.25	0.23
4	0.25	0.22	0.22

(a) Northern Hemisphere

w	$\boldsymbol{DRPSS}_{\mathrm{MM},(\alpha)}$	$\boldsymbol{DRPSS}_{\mathrm{E},(\alpha)}$	$\boldsymbol{DRPSS}_{\mathrm{C},(\alpha)}$
1	0.71	0.70	0.63
2	0.43	0.42	0.36
3	0.32	0.30	0.28
4	0.27	0.25	0.24

(b) Southern Hemishpere

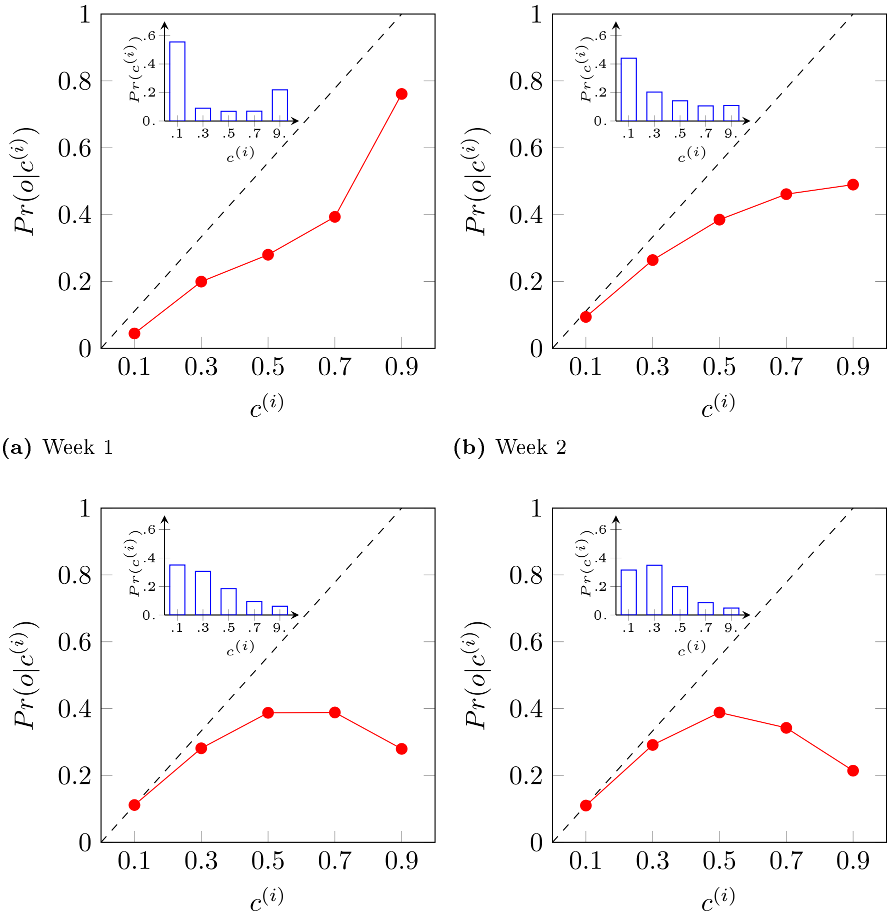
w	$\boldsymbol{DRPSS}_{\mathrm{MM},(\alpha)}$	$\boldsymbol{DRPSS}_{\mathrm{E},(\alpha)}$	$\boldsymbol{DRPSS}_{\mathrm{C},(\alpha)}$
1	0.59	0.57	0.50
2	0.45	0.42	0.38
3	0.39	0.36	0.33
4	0.36	0.33	0.32

(c) Equatorial Belt

w	$\boldsymbol{DRPSS}_{\mathrm{MM},(\alpha)}$	$\boldsymbol{DRPSS}_{\mathrm{E},(\alpha)}$	$\boldsymbol{DRPSS}_{\mathrm{C},(\alpha)}$
1	0.72	0.71	0.64
2	0.34	0.32	0.28
3	0.20	0.18	0.17
4	0.18	0.16	0.17

(d) Europe

Table 4.4: As for Table 4.3, but the score presented is the Discrete Ranked Probability Skill Score (DRPSS) instead of the RPSS.



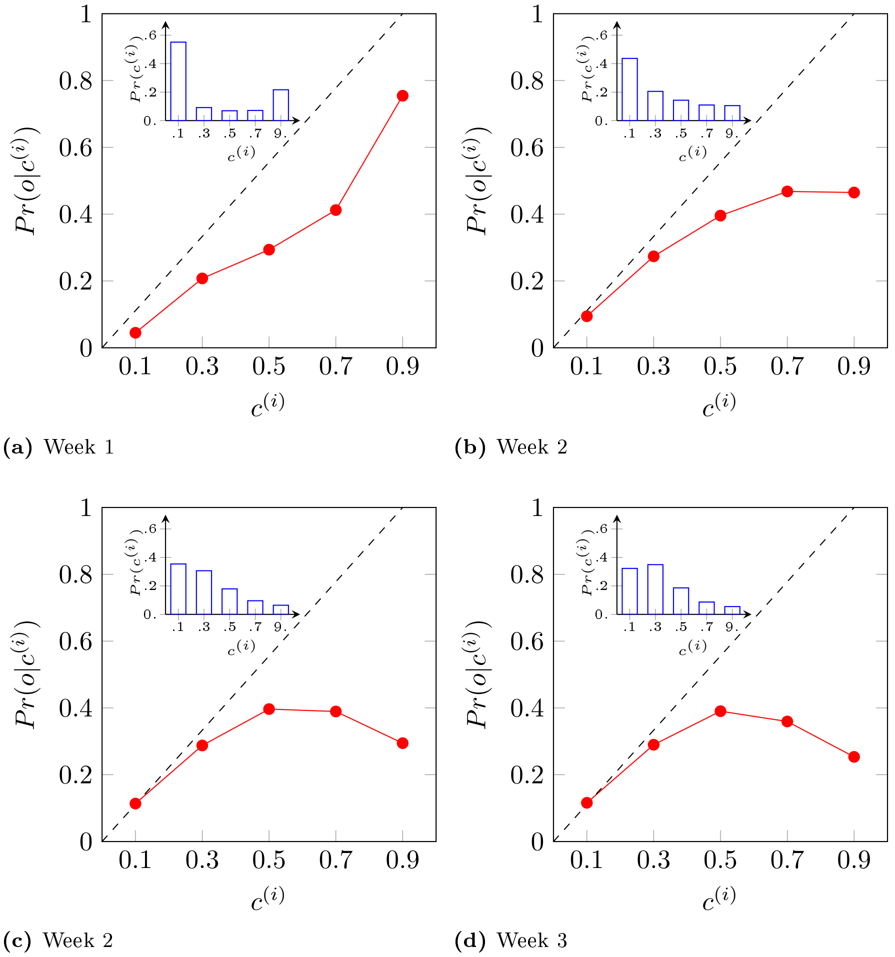
(a) Week 1

(b) Week 2

(c) Week 2

(d) Week 3

Figure 4.13: Reliability diagrams for the α variant of logistic regression using $\lambda = 0$ applied to the multi-model, computed for the lower tercile tresholds. The four panel refer to the forecast weeks, as suggested by the label below them. Naturally, all the values refer to an average over the whole globe and over the 18 validation winters. Each of the panel has the same structure: the plot outside shows the calibration function (in red), while the smoller plot in the corner contains the refinement distribution (in blue).



(a) Week 1

(b) Week 2

(c) Week 2

(d) Week 3

Figure 4.14: As in Figure 4.13, but for the upper tercile.

to the bisector of the quadrant. For the low probability intervals the red and dashed curves are closer, while the gap increases for higher values. The same behaviour was also present for the DMO techniques, but now the overall trend is rather constant, with the central categories representing better the corresponding frequency of the verifying renanalysis. The refinement distribution resembles the one seen for the DMO, with the extremes values being more frequent than the central one, a symptom of a confident forecast.

From the second week we start to see an interesting (and rather strange) behaviour. The bias becomes conditional and depends on the probability category. The red curve starts very close to the ideal case, and deviates more as the value on the x-axis increases. The trend is accentuated in the following weeks, where the curve actually changes slopes in the middle of the probability range. Also the refinement distribution changes, with higher values of probability becoming more unlikely while the distribution shifts towards the first two values of $p^{(c)}$. So, the cases in which the algorithm performs more badly ($\overline{p}^{(c)} = 0.7$ and $\overline{p}^{(c)} = 0.9$) are also the cases that appears more + rarely and so the impacts on the whole performances is limited. Nevertheless, we note that, expecially for $\overline{p}^{(c)} = 0.9$, DMO presents a calibration function closer to the bisector. For the other values, however, LR outperforms the previous methodologies, even considering the strange shape of the curve.

This analysis can be considered valid for both terciles, due to the similarity of their scores. Often, differences in corresponding values can be found only after the second decimal place, it is in fact difficult to notice such a small variation in the two figures.

In conlcusion, we consider logistic regression as a valid algorithm for improving the probabilistic forecast compared to the direct model techniques. In its simplest form (the one involving the sole ensemble mean) it brings significant improvements over the extended range, with improvements more evident for small enseble sizes, like the two single models.

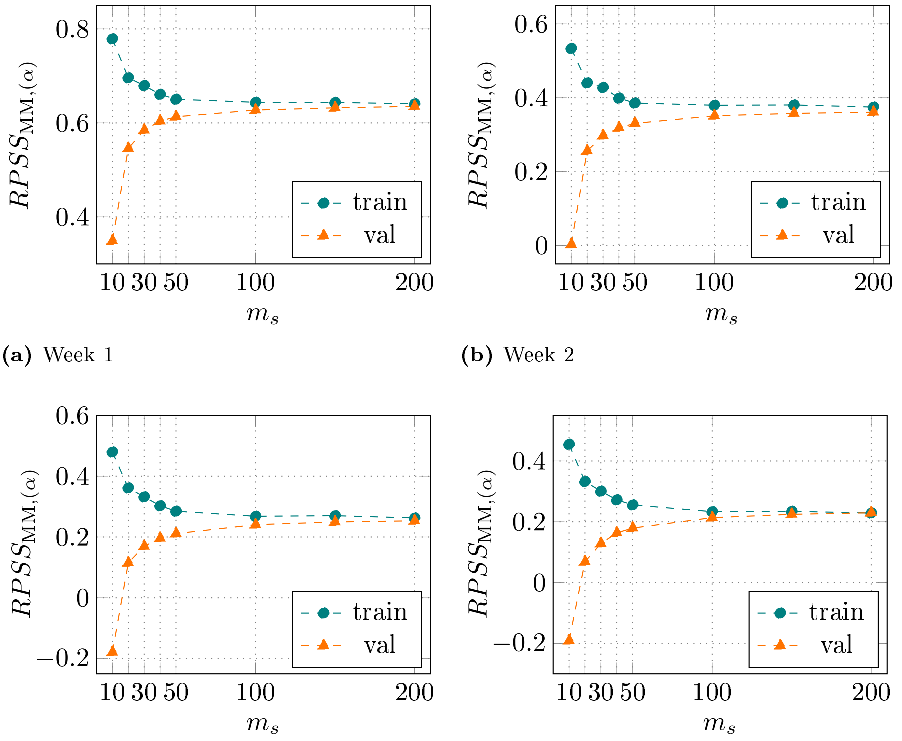
4.4.3 Learning curve

One of the most useful tools in diagnosing overfitting and underfitting is the learning curve. We previously described its implementation and how to understand the results, showing also a theoretical example in Figure 4.7. We now apply the technique to the chosen version of LR, limited to the multi-model case. For computational reason, we perform the analysis again on the reduced validation sets, composed of the same six winters used in the algorithm selection.

First of all, we decide eight values $m_\mathrm{s} < m$, with m representing the dimension of the full training set, as the dimensions of the reduced training sets. Then, we repeat the full logistic regression for all the values m_s, using each time a subset of the full training set, containing only m_s elements.

The procedure is obviously applied for all the weeks and grid points. Using the resulting coefficients, we compute the RPSS, wich is therefore averaged over all the globe. We repeat the operation for the six choice of the validation winters, and the six values are finally averaged ending up in the scores shown in Figure 4.15.

In the four panels of Figure 4.15 we see a similar trend: for small training sets the algorithm performs exceptionally well in the training phase, while in the



(a) Week 1

(b) Week 2

(c) Week 3

(d) Week 4

Figure 4.15: Learning curves for the variant α of LR applied to the multi-model. Each of the four panels show the results for one of the forecast weeks. The score used for the procedure is the ranked probability skill score averaged over the whole globe and the 6 validation winters, and it is shown on the y-axis. On the x-axis there are the eight values chosen for the reduced training sets. The scores obtained on the training sets are shown in teal, while the ones for the validation set are in orange. The dashed lines between the markers have been added for aiding the visualization. Note that, due to the interpretation of the score, the relative position of the training and vaidation curves is the opposite respect to the one presented in Figure 4.7.

validation one shows poor results. the two values becomes closer for increasing m_s, and above the thershold $m_s = 100$ the scores remain nearly constant.[7]

There are two different interpretation for this kind of behaviour. One possibility is that LR has extracted all the available information from the data and therefore no further improvements are possible in this case. The second interpretation is more interesting: there is some additional predictabilty inside our dataset, but the algorithm is not capable of catching it. This means that our LR is affected by underfitting: adding training examples do not improve the result (as can be seen for $m_s > 100$), and different solution needs to be adopted, like adding more features or combining the existent ones by creating polynomial terms.

However, two of the most widely used choices for additional features (σ and $\overline{x}\sigma$, following the previous notation) did not result in iprovements in our case. On the contrary, we saw a deterioration of the performances, as described in the algorithm selection section (4.4.1).

[7] Note that we are keping $\lambda = 0$ during the analysis. The values shown for small m_s are therefore not representative of the optimal performances of LR on such reduced datset. In fact, when a small number of examples is used for training, overfit can be reduced (or avoided) using the reguilarization parameter. This imply that a complete analysis for each of the values of m_s would require the selection of the optimal λ for each of them. However, this is not the scope of our analysis: we are studying the behaviour of the selected version of LR on reduced datasets for identifying overfitting or underfitting problems, and for consistency puropouses we keep all the parameters constant and equal to their definitive values.

Chapter 5

Nonhomogeneous Gaussian Regression

The last regression techique adopted in the analysis is the Nonhomogeneous Gaussian Regression (NGR). This MOS method is an extension of linear regression, first introduced by Gneiting et al. [2005]. The distinctive trait of the algorithm is the possibility for the residual variance to depend (linearly) from the ensemble variance. This result in the forecast having an higher uncetainty when the ensemble members are more dispersed, and a lower one in the opposite situation.

The algorithm is commonly used for predicting probabilities in the scientific literature, examples can be found in Wilks [2006], Wilks and Hamill [2007], Kann et al. [2009] and Hagedorn et al. [2008] (in the last two NGR results particularly promising when used on surface temperature). As always, a complete exposition of the algorithm can be found in [Wilks, 2011, Chapter 8], which we use here as the basis for the description presented in the following section.

5.1 Methodology

The basic implementation of NGR is rather simple and we begin the exposition directly by applying the technique to the computation of tercile probabilities. We introduce again the same simplified notation used in the previous chapter when we first applied LR to the same task. So, we use $\overline{x}^{(i)}$ and $\sigma^{(i)}$ respectively for the ensemble mean and standard deviation of the i-th example, $t_{\text{2m}}^{(i)}$ for the corresponding verifying 2-metre temperature, $q_{1/3}$ and $q_{2/3}$ for the two terciles, q for a generic quantile and m for the dimension of the dataset.

The basic idea behind the algorithm is to perform a simple linear regression:

$$t_{\text{2m}}^{(i)} = \theta_0 + \theta_{\overline{x}}\,\overline{x}^{(i)} + \epsilon^{(i)},$$

where $\epsilon^{(i)}$ represent the residual for the i-th example. We assume that these residuals follow a Gaussian distribution, whose variance is given by:

$$\left(\sigma_{\epsilon}^{(i)}\right)^2 = \theta_2 + \theta_{\sigma}\left(\sigma^{(i)}\right)^2.$$

The probability that the temperature anomaly is below q is then given by:

$$Pr\big(t_{2\text{m}}^{(i)} < q\big) = \Phi\bigg(\frac{q - (\theta_0 + \theta_{\overline{x}}\,\overline{x}^{(i)})}{\sigma_\epsilon^{(i)}}\bigg), \tag{5.1}$$

where Φ is the cumulative distribution function (CDF) of the standard Gaussian distribution. Naturally, the coefficients θ_0, $\theta_{\overline{x}}$, θ_2 and θ_σ are computed by minimizing the chosen cost function. Before proceeding with the exposition of the procedure and its details, we vectorize the notation in order to explain exactly what happens when NGR is actually applied.

We start by redefining a unique binary verification vector for the two quantiles, as for the unified LR:

$$\boldsymbol{y} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(i)} \\ \vdots \\ y^{(m)} \\ y^{(m+1)} \\ \vdots \\ y^{(m+i)} \\ \vdots \\ y^{(2m)} \end{bmatrix},$$

with:

$$y^{(i)} = \begin{cases} 1 & \text{if } t_{2\text{m}}^{(i)} < q_{1/3} \\ 0 & \text{otherwise,} \end{cases} \quad \text{for } i = 1, ..., m$$

$$y^{(i)} = \begin{cases} 1 & \text{if } t_{2\text{m}}^{(i)} < q_{2/3} \\ 0 & \text{otherwise,} \end{cases} \quad \text{for } i = m+1, ..., 2m$$

The feature vectors are:

$$\left.\boldsymbol{f}_0 = \boldsymbol{f}_2 \begin{bmatrix} 1 \\ \vdots \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \ \boldsymbol{f}_{\overline{x}} = \begin{bmatrix} \overline{x}^{(1)} \\ \vdots \\ \overline{x}^{(m)} \\ \overline{x}^{(1)} \\ \vdots \\ \overline{x}^{(m)} \end{bmatrix}, \ \boldsymbol{f}_\sigma = \begin{bmatrix} \sigma^{(1)} \\ \vdots \\ \sigma^{(m)} \\ \sigma^{(1)} \\ \vdots \\ \sigma^{(m)} \end{bmatrix}, \ \boldsymbol{f}_q = \begin{bmatrix} q_{1/3} \\ \vdots \\ q_{1/3} \\ q_{2/3} \\ \vdots \\ q_{2/3} \end{bmatrix} \right\} 2m \text{ elements.}$$

Unlike LR, in NGR there are two "bias units": $\boldsymbol{f}_0$ and $\boldsymbol{f}_2$, both containings all elements equal to one. They have been given different subscripts in order to distinguish between the correspoding coefficients. The first four vector are combined in two separate matrices:

$$\boldsymbol{X}_{\text{num}} = \begin{bmatrix} \text{—} & (\boldsymbol{f}_0)^{\text{T}} & \text{—} \\ \text{—} & (\boldsymbol{f}_{\overline{x}})^{\text{T}} & \text{—} \end{bmatrix}, \quad \boldsymbol{X}_{\text{den}} = \begin{bmatrix} \text{—} & (\boldsymbol{f}_2)^{\text{T}} & \text{—} \\ \text{—} & (\boldsymbol{f}_\sigma)^{\text{T}} & \text{—} \end{bmatrix},$$

note that $\boldsymbol{f}_q$ is not part of them. The remaining vector necessary for the analysis are the ones containing the coefficients, whose shape is determined by the choice of the features and therfore are given by:

$$\boldsymbol{\theta}_{\text{num}} = \begin{bmatrix} \theta_0 \\ \theta_{\overline{x}} \end{bmatrix}, \qquad \boldsymbol{\theta}_{\text{den}} = \begin{bmatrix} \theta_2 \\ \theta_{\sigma} \end{bmatrix}.$$

We can now define the hypothesis function, obviously based on Equation 5.1:

$$h\big(\boldsymbol{X}_{\text{num}}, \boldsymbol{X}_{\text{den}};\, \boldsymbol{f}_q, \boldsymbol{\theta}_{\text{num}}, \boldsymbol{\theta}_{\text{den}}\big) = \Phi\left(\frac{\boldsymbol{f}_q - (\boldsymbol{\theta}_{num})^T \boldsymbol{X}_{num}}{(\boldsymbol{\theta}_{den})^T \boldsymbol{X}_{den}}\right),$$

where the fraction inside the parenthesis is an elementwise operation. As seen previously, the semicolon is used to distinguish between variables (the feature matrcies) and parameters of the regression.

Finally, we have different possible choices for the cost function, some of which are rather interesting. For example Gneiting et al. [2005] proposed to minimize the continuous ranked probability score, that in our case is:

$$\boldsymbol{CRPS} = \big((\boldsymbol{\theta}_{den})^T \boldsymbol{X}_{den}\big)\Big(\boldsymbol{z}\big(2\Phi(\boldsymbol{z}) - 1\big) + 2\phi(\boldsymbol{z}) - \frac{1}{\sqrt{\pi}}\Big),$$

where ϕ is the Gaussian probability distribution function (PDF) and:

$$\boldsymbol{z} = \begin{bmatrix} z^{(1)} \\ \vdots \\ z^{(m)} \\ z^{(1)} \\ \vdots \\ z^{(m)} \end{bmatrix} . \quad \text{with: } z^{(i)} = \frac{t_{\text{2m}}^{(i)} - \theta_0 - \theta_{\overline{x}}\,\overline{x}^{(i)}}{\theta_2 + \theta_\sigma\, \sigma^{(i)}}.$$

However, we opted for a simpler choice, and we minimize numerically the average squared error, wich becames our cost function:

$$J(\boldsymbol{\theta}_{\text{num}}, \boldsymbol{\theta}_{\text{den}}) = \frac{1}{m}\sum_{i=1}^{m}\big(y^{(i)} - Pr(t_{\text{2m}}^{(i)} < q)\big)^2.$$

In order to perform the numerical minimization, we use again BFGS[1].

5.1.1 Application to the re-forecasts

The analysis described in the previous section refers to a single application of NGR, which in our case corresponds to keeping fixed both the forecast week and the grid point. So, before applying the procedure to the real dataset, we adjust the notation using the usual tensor notation.

To avoid beeing verbose, we do not repeat again the whole procedure from the beginning, but when possible we use some tensors already defined when describing LR (section “Application to the re-forecasts”). We start from the features tensors $\boldsymbol{F}_{\text{MM},\overline{x}}(w, i, j, d)$, $\boldsymbol{F}_{\text{MM},\sigma}(w, i, j, d)$ and $\boldsymbol{F}_q(w, i, j, d)$, containing

[1] In the previous chapter we provide information on the specific Python package used.

respectively the multi-model enseble mean, the standard deviation of the same model and the two quantiles. As always, changing the superscript with E or C give the corresponding value for the ECMWF-IFS or the CNR-ISAC model. The bias unit also became tensors: $\boldsymbol{F}_{\text{MM},0}(w,i,j,d)$ and $\boldsymbol{F}_{\text{MM},2}(w,i,j,d)$. As expected, their elements are all equal to one and they have the exact same shape as the other three tensor just introduced. Starting from $\boldsymbol{B}$ (introduced in the DMO chapter), we also redefine the verification tensor:

$$\boldsymbol{Y}_{\text{ngr}}(w,i,j,:)=\begin{bmatrix}\boldsymbol{B}(w,i,j,0,0)\\ \vdots\\ \boldsymbol{B}(w,i,j,d,0)\\ \vdots\\ \boldsymbol{B}(w,i,j,m,0)\\ 1-\boldsymbol{B}(w,i,j,0,2)\\ \vdots\\ 1-\boldsymbol{B}(w,i,j,d,2)\\ \vdots\\ 1-\boldsymbol{B}(w,i,j,m,2)\end{bmatrix},$$

where the subscript ngr stands for nonhomogeneous Gaussian regressiona and is necessary for distinguish this tensor from the one containing the reanalysis.

Now the notation begins to differ from the LR one. In fact, we distinguish between the features at the numerator in the hypothesis function and the ones at the denominator. So, the two feature matrices previously described become:

$$\boldsymbol{X}_{\text{MM,(num)}}(w,i,j,:,:)=\begin{bmatrix}— & \left(\boldsymbol{F}_{\text{MM},0}(w,i,j,:)\right)^{\text{T}} & —\\ — & \left(\boldsymbol{F}_{\text{MM},\overline{x}}(w,i,j,:)\right)^{\text{T}} & —\end{bmatrix},$$

$$\boldsymbol{X}_{\text{M,(den)}}(w,i,j,:,:)=\begin{bmatrix}— & \left(\boldsymbol{F}_{\text{MM},2}(w,i,j,:)\right)^{\text{T}} & —\\ — & \left(\boldsymbol{F}_{\text{MM},\sigma}(w,i,j,:)\right)^{\text{T}} & —\end{bmatrix}.$$

As for LR, we try two different possibilities for the ensemble standard deviation of the multi-model. In $\boldsymbol{X}_{\text{M,(den)}}$, we use $\boldsymbol{F}_{\text{MM},\sigma}$, based on $\boldsymbol{S}_{\text{MM}}$, which in turn is computed directly from the 25 ensemble members. We propose an alternative version in wich both the standard deviation from the ECMWF-IFS ($\boldsymbol{S}_{\text{E}}$) and CNR-ISAC ($\boldsymbol{S}_{\text{C}}$) models are used istead of $\boldsymbol{S}_{\text{MM}}$. The resulting denominator tensor is:

$$\boldsymbol{X}_{\text{M,(den - EC)}}(w,i,j,:,:)=\begin{bmatrix}— & \left(\boldsymbol{F}_{\text{MM},2}(w,i,j,:)\right)^{\text{T}} & —\\ — & \left(\boldsymbol{F}_{\text{E},\sigma}(w,i,j,:)\right)^{\text{T}} & —\\ — & \left(\boldsymbol{F}_{\text{C},\sigma}(w,i,j,:)\right)^{\text{T}} & —\end{bmatrix}.$$

Again, this is equivalent to assuming that the standard deviation of the multi-model is a linear combination of the same quantity computed for the two initial models. The goodness of the approximation is evaluated by comparing the scores for the two choice of features. Note that the numerator tensor is not modified in this alternative version.

The next step is the redefinition of the coefficient tensors:

$$\mathbf{\Theta}_{\mathrm{MM,(num)}}(w,i,j,:) = \begin{bmatrix} \left(\mathbf{\Theta}^{\mathrm{(num)}}_{\mathrm{MM},0}(w,i,j)\right) \\ \left(\mathbf{\Theta}^{\mathrm{(num)}}_{\mathrm{MM},\overline{x}}(w,i,j)\right) \end{bmatrix},$$

$$\mathbf{\Theta}_{\mathrm{MM,(den)}}(w,i,j,:) = \begin{bmatrix} \left(\mathbf{\Theta}^{\mathrm{(den)}}_{\mathrm{MM},0}(w,i,j)\right) \\ \left(\mathbf{\Theta}^{\mathrm{(den)}}_{\mathrm{MM},\sigma}(w,i,j)\right) \end{bmatrix},$$

$$\mathbf{\Theta}_{\mathrm{MM,(den\ -\ EC)}}(w,i,j,:) = \begin{bmatrix} \left(\mathbf{\Theta}^{\mathrm{(den\ -\ EC)}}_{\mathrm{MM},0}(w,i,j)\right) \\ \left(\mathbf{\Theta}^{\mathrm{(den\ -\ EC)}}_{\mathrm{E},\sigma}(w,i,j)\right) \\ \left(\mathbf{\Theta}^{\mathrm{(den\ -\ EC)}}_{\mathrm{C},\sigma}(w,i,j)\right) \end{bmatrix},$$

Finally, the hypothesis function is:

$$\begin{aligned} h\big(&\boldsymbol{X}_{\mathrm{(num)}}(w,i,j,:,:),\boldsymbol{X}_{\mathrm{(den)}}(w,i,j,:,:); \\ &\boldsymbol{F}_q(w,i,j,:),\mathbf{\Theta}_{\mathrm{(num)}}(w,i,j,:),\mathbf{\Theta}_{\mathrm{(den)}}(w,i,j,:)\big) = \\ &\Phi\left(\frac{\boldsymbol{F}_q(w,i,j)-(\boldsymbol{\theta}_{\mathrm{(num)}}(w,i,j,:))^T\boldsymbol{X}_{\mathrm{(num)}}(w,i,j,:,:)}{(\boldsymbol{\theta}_{\mathrm{(den)}}(w,i,j,:))^T\boldsymbol{X}_{\mathrm{(den)}}(w,i,j,:,:)}\right), \end{aligned}$$

where the divisaion in parenthesis is performed elementwise, the result is a vector (of lenght m) and Φ is computed for each of its element. An analogous vectorization is applied to the cost function, which becomes:

$$\begin{aligned} J\big(\mathbf{\Theta}_{\mathrm{(num)}}(w,i,j,:),\mathbf{\Theta}_{\mathrm{(den)}}(w,i,j,:)\big) &= \frac{1}{m}\sum_{d=1}^{m}\big(\boldsymbol{Y}_{\mathrm{ngr}}(w,i,j,d)- \\ &h\big(\boldsymbol{X}_{\mathrm{(num)}}(w,i,j,:,d),\boldsymbol{X}_{\mathrm{(den)}}(w,i,j,:,d); \\ &\boldsymbol{F}_q(w,i,j),\mathbf{\Theta}_{\mathrm{(num)}}(w,i,j,:),\mathbf{\Theta}_{\mathrm{(den)}}(w,i,j,:)\big)\big)^2. \end{aligned}$$

From its minimization over the training dataset we obtain the optimal coefficients $\boldsymbol{\theta}^*_{\mathrm{num}}$ and $\boldsymbol{\theta}^*_{\mathrm{den}}$. Once those values have been determined, we define the probabilities for the usual three categories as:

$$\begin{aligned} \boldsymbol{P}^{NGR}_{inf}(w,i,j,d) = h\big(&\boldsymbol{X}_{\mathrm{(num)}}(w,i,j,:,d),\boldsymbol{X}_{\mathrm{(den)}}(w,i,j,:,d); \\ &\boldsymbol{Y}_{1/3}(w,i,j),\mathbf{\Theta}_{\mathrm{(num)}}(w,i,j,:),\mathbf{\Theta}_{\mathrm{(den)}}(w,i,j,:)\big), \end{aligned}$$

$$\begin{aligned} \boldsymbol{P}^{NGR}_{mid}(w,i,j,d) = h\big(&\boldsymbol{X}_{\mathrm{(num)}}(w,i,j,:,d),\boldsymbol{X}_{\mathrm{(den)}}(w,i,j,:,d); \\ &\boldsymbol{Y}_{2/3}(w,i,j),\mathbf{\Theta}_{\mathrm{(num)}}(w,i,j,:),\mathbf{\Theta}_{\mathrm{(den)}}(w,i,j,:)\big), \end{aligned}$$

$$\boldsymbol{P}^{NGR}_{sup}(w,i,j,d) = 1-\boldsymbol{P}^{LR}_{mid}(w,i,j,d)-\boldsymbol{P}^{LR}_{inf}(w,i,j,d).$$

Naturally, these quantities are computed over all the validation sets and are used in the following as the basis for the probabilistic scores.

5.2 Results

Unlike the previous chapter, we can directly expose the NGR results, instead of performing first the algorithm selection. We begin, as usual, with the probabilistic scores: RPSS and DRPSS, averaged over the four regions and over the 18 validation winters (from the cross validation procedure). The results are shown in Table 5.1-5.2 , with the same structure as in the previous chapters.

In this case, different possibilities are tested for the sole multi-model, for which we tried two choices of predictand for the standard deviation. The procedure is analogous to the one seen fo LR and the underlying assumpion, togeteher with the potential benefit of one version over the other, are the same exposed in the previos chapter.

The results are similar to what we saw with the other algorithms. The multi-model shows the best performances nearly always, even if its value are often close to the ECMWF-IFS one, expecially in the first week. The similarity is even more evident when we look at the DRPSS. The score, in fact, is less sensitive to the ensemble dimension and the values for the two single models are closer to the multi-model ones. This behaviour is not surprising: if we recall the coefficients used for producing the multi-model fields, we see that due to the enhanced resolution of the ECMWF-IFS model in the first perdiod, its weights for the first week are significantly higher than the CNR-ISAC ones. As a result, probably much of the skill of the multi-model over this periods comes indeed from the ECMWF-IFS foreasts.

The comparison between the two variants for the choice of the ensemble standard deviation is also interesting. Both of them result in sufficiently high scores, with the approximate version (the one denoted by EC in the subscript) resulting in generally better performances over the northern hemisphere and Europe, while the standard version ouperform the other over the two remaining regions. We therefore conclude that, for this specific algorithm, using a linear combination of the ensemble standard deviations of the two single models instead of the multi-model is a reasonable choice.

Another noticeable information that can be extracted from the tables is the difference in predictability between the four regions. We see another confirmation of what we noticed with the other techniques. In the northern and southern hemisphere we find high scores for the first and second week, followed by two nearly constant and rather low values over the remaining period. The equatorial belt has lower performances at the beginning of the forecast, but higher score over the final weeks. Finally, Europe results highly predictable in the first week, but the sharp decrease in the successive period leads, at the end of the forecast, to the lowest scores of all the regions. Naturally, the division between the firts two weeks and the extendend range is always evident, as for DMO and LR.

After this comparison, we show the reliability diagrams for the multi-model, using the standard version of the algorithm (the one where the standard deviation is computed from the 25 artificial members). Naturally, the fist week is the one where the calibration function is closer to the black dashed line. The curve remains rather straight also in the second week, altought the slope decreases. Over the final two weeks we see again the peculiar behaviour noticed for LR: the last probability cathegory shows frequecies lower than its neighbour on the left. This is a conditional bias (because it depends strongly on the category $c^{(i)}$) that, strangely, indicates low skill when the algorithm is more confident in its

w	$\boldsymbol{RPSS}_{\text{MM}}$	$\boldsymbol{RPSS}_{\text{MM, (EC)}}$	$\boldsymbol{RPSS}_{\text{E}}$	$\boldsymbol{RPSS}_{\text{C}}$
1	0.62	0.60	0.61	0.54
2	0.31	0.31	0.29	0.24
3	0.16	0.17	0.15	0.14
4	0.14	0.15	0.13	0.12

(a) Northern Hemisphere

w	$\boldsymbol{RPSS}_{\text{MM}}$	$\boldsymbol{RPSS}_{\text{MM, (EC)}}$	$\boldsymbol{RPSS}_{\text{E}}$	$\boldsymbol{RPSS}_{\text{C}}$
1	0.60	0.55	0.59	0.53
2	0.34	0.32	0.31	0.26
3	0.22	0.20	0.19	0.17
4	0.18	0.16	0.15	0.14

(b) Southern Hemishpere

w	$\boldsymbol{RPSS}_{\text{MM}}$	$\boldsymbol{RPSS}_{\text{MM, (EC)}}$	$\boldsymbol{RPSS}_{\text{E}}$	$\boldsymbol{RPSS}_{\text{C}}$
1	0.44	0.39	0.41	0.34
2	0.31	0.28	0.27	0.24
3	0.25	0.21	0.21	0.19
4	0.23	0.18	0.18	0.17

(c) Equatorial Belt

w	$\boldsymbol{RPSS}_{\text{MM}}$	$\boldsymbol{RPSS}_{\text{MM, (EC)}}$	$\boldsymbol{RPSS}_{\text{E}}$	$\boldsymbol{RPSS}_{\text{C}}$
1	0.63	0.61	0.62	0.55
2	0.23	0.25	0.23	0.19
3	0.09	0.10	0.08	0.08
4	0.08	0.09	0.06	0.07

(d) Europe

Table 5.1: Ranked Probability Skill Score (RPSS) for the nonhomogeneous Gaussian regression technique, averaged over the 18 validation winters. The four table present the spatial average over the four different regions defined in the second chapter. The first column always shows, in blue, the week. The second and third columns contains the scores for the two variants applied to the multi-model: the subscript MM refers to the algorithm using $\boldsymbol{S}_{\text{MM}}$ as the standard deviation, while in MM - EC we use a linear combination of $\boldsymbol{S}_{\text{E}}$ and $\boldsymbol{S}_{\text{C}}$. Finally, the last two columns refers respectively to the ECMWF-IFS and CNR-ISAC models.The value corresponding to the best performances (the lowest ones) for each row is highlighted in red.

w	$\boldsymbol{DRPSS}_{\text{MM}}$	$\boldsymbol{DRPSS}_{\text{MM, (EC)}}$	$\boldsymbol{DRPSS}_{\text{E}}$	$\boldsymbol{DRPSS}_{\text{C}}$
1	0.66	0.64	0.64	0.58
2	0.37	0.37	0.36	0.31
3	0.24	0.25	0.23	0.22
4	0.22	0.23	0.21	0.20

(a) Northern Hemisphere

w	$\boldsymbol{DRPSS}_{\text{MM}}$	$\boldsymbol{DRPSS}_{\text{MM, (EC)}}$	$\boldsymbol{DRPSS}_{\text{E}}$	$\boldsymbol{DRPSS}_{\text{C}}$
1	0.64	0.60	0.63	0.57
2	0.40	0.38	0.38	0.33
3	0.29	0.27	0.27	0.25
4	0.26	0.24	0.23	0.22

(b) Southern Hemishpere

w	$\boldsymbol{DRPSS}_{\text{MM}}$	$\boldsymbol{DRPSS}_{\text{MM, (EC)}}$	$\boldsymbol{DRPSS}_{\text{E}}$	$\boldsymbol{DRPSS}_{\text{C}}$
1	0.49	0.45	0.47	0.40
2	0.38	0.35	0.34	0.31
3	0.32	0.28	0.28	0.27
4	0.30	0.26	0.25	0.25

(c) Equatorial Belt

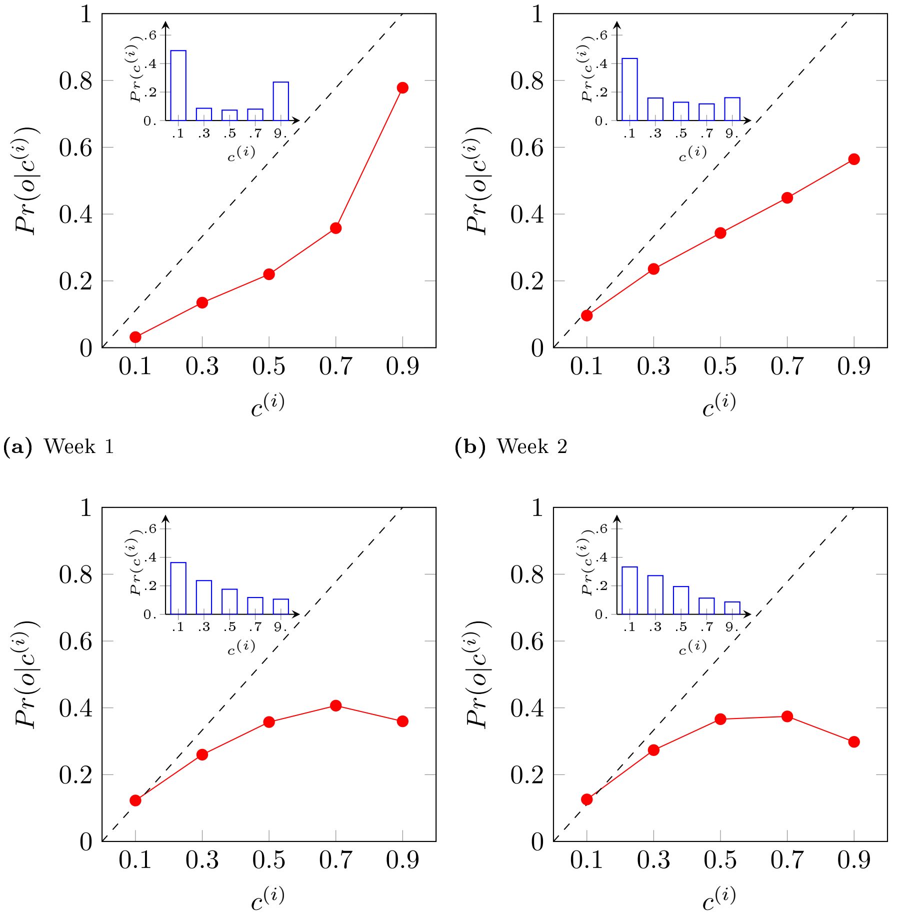
w	$\boldsymbol{DRPSS}_{\text{MM}}$	$\boldsymbol{DRPSS}_{\text{MM, (EC)}}$	$\boldsymbol{DRPSS}_{\text{E}}$	$\boldsymbol{DRPSS}_{\text{C}}$
1	0.66	0.65	0.66	0.59
2	0.31	0.32	0.30	0.26
3	0.17	0.18	0.17	0.16
4	0.16	0.17	0.15	0.16

(d) Europe

Table 5.2: As for Table 5.1, but the score presented is the Discrete Ranked Probability Skill Score (DRPSS).

prediction. Looking at the distribution in the upper left corner of the plot, the first two weeks behave similarly to LR, presenting some hints of overconfidence. Some differences can be seen in the extended range, where the freqeuncies of all the categories are closer to one another respect to the LR case. Lower values of probability are more likely to appear and it is interesting to notice that, for the fourth week, we do not see $p(c^{(2)}) > p(c^{(1)})$ as for LR.

We conclude that nonhomogeneous Gaussian regression is a valid algorithm for forecasting probabilities regarding the relative values of the 2-metre temperature anomaly respect to the terciles of the reanalysis. Again, it produces significant improvements on the scores obtainable directly from the ensembles, for both the multi-model combination and the single models, particularly over the extended range of the forecast.



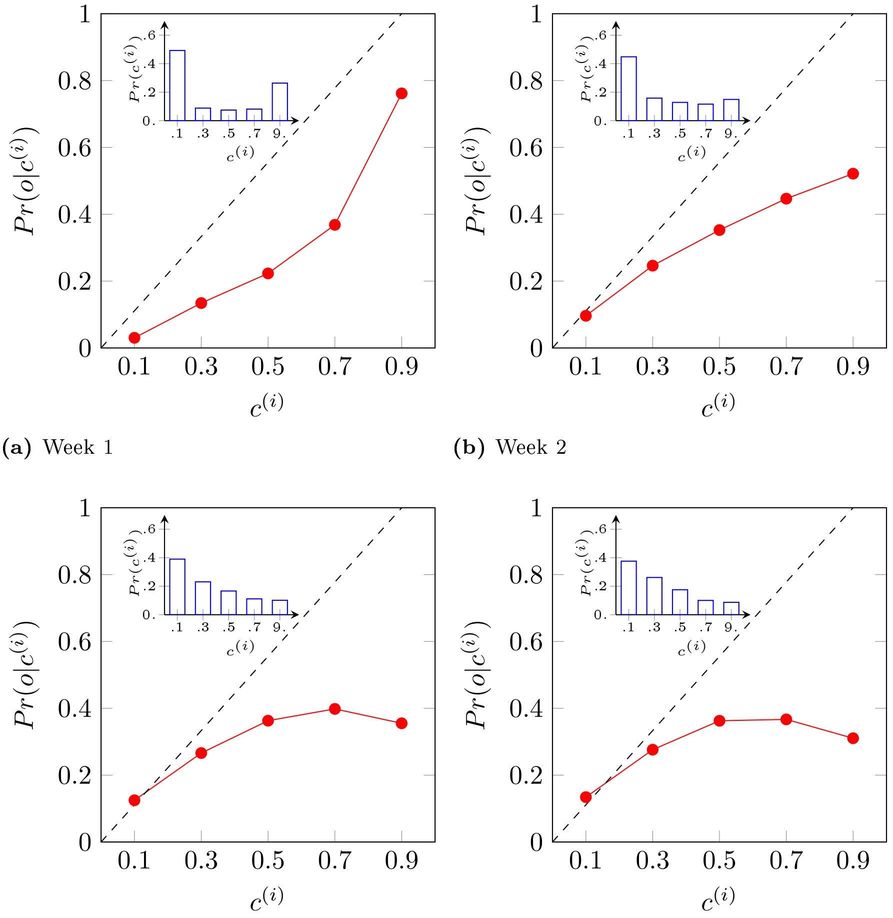
(a) Week 1

(b) Week 2

(c) Week 2

(d) Week 3

Figure 5.1: Reliability diagrams for the nonhomogeneous Gaussian regression applied to the multi-model, computed for the lower tercile treshold. As always, the four panel refer to the forecast weeks and all the values refer to an average over the whole globe and over the 18 validation winters. Each of the panel has the usual structure: the red curve represents the calibration function, while the smaller plot in the corner contains the refinement distribution (in blue).



(a) Week 1

(b) Week 2

(c) Week 2

(d) Week 3

Figure 5.2: As in Figure 5.1, but for the upper tercile.

Chapter 6

Summary and conclusions

The work done in this thesis is fundamentally the evaluation, from a probabilistic and non-probabilistic point of view, of a multi-model combination of the re-forecasts from the ECMWF-IFS and CNR-ISAC prediction systems. In this chapter we summarize the results, focusing on the multi-model and following the same order in which the various methods have been described in the previous chapters. The exposition is divided in two parts: we first focus on the multi-model implementation and the non-probabilistic scores directly obtainable from it, while in the second part we analyze the MOS results, comparing the different techniques tested.

6.1 Multi-model and non-probabilistic scores

In Chapter 2 we computed the multi-model fields for the 2-meter temperature, temperature at 850 hPa and geopotential height at 500 hPa anomalies, using a linear regression of the re-forecast of the two prediction systems against the ERA-Interim reanalysis. The first interesting results are the maps of the regression coefficients. For the first two weeks the ECMWF-IFS model has significantly higher weights than the CNR-ISAC one, probably due to its enhanced resolution over the first 10 days. However, the difference between the two models is less marked over the extended range. Some additional information can be obtained by looking at the sum of these coefficients: values close to 1 hints the presence of some predictability in the region, while where the sum is close to 0 the climatology represents the best possible prediction. In the first week, the maps are very homogeneous and the sum is close to one nearly everywere, while in the following weeks, the pattern depends on the variable considered. For the geopotential height at 500 hPa and the temperature at 850hPa, the equatorial belt seems particularly predictable over the extended range, while Europe and some areas over the oceans in the southern extratropics show considerably low values, sometimes even below zero. The 2-meter temperature shows, over Siberia and the Antarctic, values significantly greater than one, which suggests an underestimate of the anomalies over these regions by the single models.

We then computed the RMSE and anomaly correlation, following a cross validation procedure in which the dataset was split in 18 winters. Each winter was used, in turn, for validation, while the others acted as training set. We

$\overline{rmse}$

	(NH)		(SH)		(EB)		(EU)	
w	MM	BM	MM	BM	MM	BM	MM	BM
1	22.5	22.6	19.9	20.1	3.95	4.09	25.2	25.4
2	67.1	70.8	52.7	55.8	8.56	9.05	81.3	85.7
3	82.8	90.5	61.5	66.2	10.9	11.8	97.1	105
4	87.0	94.2	63.0	66.9	11.7	12.8	100	108

(a) Z500

$\overline{rmse}$

	(NH)		(SH)		(EB)		(EU)	
w	MM	BM	MM	BM	MM	BM	MM	BM
1	1.08	1.11	0.88	0.91	0.51	0.54	1.03	1.06
2	2.70	2.85	1.83	1.94	0.84	0.90	2.70	2.85
3	3.29	3.52	2.09	2.27	0.99	1.08	3.23	3.46
4	3.49	3.71	2.12	2.27	1.02	1.12	3.32	3.51

(b) T850

$\overline{rmse}$

	(NH)		(SH)		(EB)		(EU)	
w	MM	BM	MM	BM	MM	BM	MM	BM
1	1.35	1.45	0.59	0.65	0.41	0.48	1.25	1.32
2	2.79	2.90	1.08	1.16	0.62	0.69	2.78	2.89
3	3.35	3.59	1.25	1.39	0.72	0.80	3.35	3.54
4	3.52	3.77	1.31	1.47	0.75	0.81	3.40	3.58

(c) T2M

Table 6.1: Comparison of the root mean square error ($\overline{rmse}$) between the multi-model (MM) and the "best single model" (BM), chosen for each region and week as the one with the best performances between the ECMWF-IFS model and the CNR-ISAC one. The three panels show the values for week (w) 1 to 4 for the three variable, as indicated by their labels. The remaining columns are grouped in pairs, which show the values for the four spatial region over which the average has been performed. The best performances for each week and region are highlighted in red.

defined four spatial regions over which we performed the spatial average: the Northern Hemisphere (NH), the Southern Hemisphere (SH), the Equatorial Belt (EB) and Europe (EU). The multi model outperforms the two single models nearly always, especially when looking at the RMSE. The relevant results are summarized in the Table 6.1 (for the RMSE) and in the Table 6.2 (for the AC), where the multi-model is compared, for each week and region, with the best between the two single models. Clearly, the multi-model outperforms the other models nearly always.

The equatorial belt shows the lower values of RMSE for all the variables, due to the low variability over the region. The highest values can be nearly always found in the Northern Hemisphere or over Europe. The anomaly correlation shows a similar behavior, although the best performances in this case corre-

$\overline{\boldsymbol{ac}}$

	(NH)		(SH)		(EB)		(EU)	
w	MM	BM	MM	BM	MM	BM	MM	BM
1	0.97	0.97	0.95	0.95	0.95	0.95	0.96	0.96
2	0.65	0.63	0.58	0.56	0.75	0.74	0.55	0.53
3	0.35	0.32	0.31	0.27	0.56	0.54	0.23	0.23
4	0.23	0.21	0.25	0.21	0.48	0.45	0.09	0.09

(a) Z500

$\overline{\boldsymbol{ac}}$

	(NH)		(SH)		(EB)		(EU)	
w	MM	BM	MM	BM	MM	BM	MM	BM
1	0.95	0.95	0.92	0.92	0.90	0.89	0.95	0.94
2	0.65	0.63	0.63	0.60	0.68	0.67	0.56	0.54
3	0.40	0.35	0.49	0.42	0.55	0.51	0.26	0.23
4	0.32	0.27	0.49	0.42	0.53	0.48	0.19	0.17

(b) T850

$\overline{\boldsymbol{ac}}$

	(NH)		(SH)		(EB)		(EU)	
w	MM	BM	MM	BM	MM	BM	MM	BM
1	0.93	0.92	0.93	0.92	0.90	0.88	0.93	0.93
2	0.68	0.65	0.79	0.77	0.77	0.73	0.62	0.60
3	0.48	0.44	0.75	0.70	0.69	0.64	0.38	0.34
4	0.44	0.40	0.75	0.70	0.67	0.62	0.37	0.33

(c) T2M

Table 6.2: As for Table 6.1, but for the anomaly correlation ($\overline{\boldsymbol{ac}}$).

spond to the highest values. For the temperature at 2 meter, the performances over the Southern Hemisphere are noticeable, due to the extended range being characterized by exceptionally high values.

For both scores, the difference between the first two weeks and the last two is particularly evident. Such behavior is expected, given the different predictability sources for the two time ranges, as discussed in Chapter 1.

6.2 Probabilistic forecast

For both the multi-model and the single models we tested different techniques for predicting the probability that the 2-m temperature anomaly falls in each of the three intervals in which the reanalysis distribution is split by its terciles.

First of all, we extracted these probabilities directly from the ensembles using two different techniques: democratic voting (DV) and the Tukey plotting position (TPP). Then, we tested two different regression techniques: logistic regression (LR) and nonhomogeneous Gaussian regression (NGR). For both techniques we tried some variants of the algorithm, selecting the one with the best performances. For LR, the chosen version is the one denoted by α in Chapter 4, which uses only the ensemble mean as a predictor, while for NGR we show the score obtained using the multi-model ensemble standard deviation computed from the 25 members available in the multi-model.

All these methods have been verified using three probabilistic scores: ranked probability score (RPS), ranked probability skill score (RPSS) and discrete ranked probability skill score (DRPSS).

In Table 6.3 we compare the performances of the four methods, using the RPSS averaged over the usual four regions. This score has been chosen because it provides the skill relative to the climatology and therefore it has a more immediate interpretation than RPS. DRPSS is not used in this comparison because we show here the scores for the sole multi-model (the ensemble dimension is constant for all the elements of the table).

From Table 6.3 it can be clearly seen that logistic regression outperforms the other methods. Also NGR shows good performances, especially in the extended range over the two hemispheres and over Europe, where its scores are significantly higher than the DMO methods (DV and TPP). It is interesting to notice how Europe represents a difficult area for making predictions over the third and fourth weeks as evident from the scores of all the four methods. The Equatorial Belt, on the other hand, results particularly predictable over the same time range.

The second tool used for analyzing the performances is the reliability diagram, shown in Figure 6.1 and Figure 6.2, for the lower and upper terciles respectively. In the first week, the calibration function for LR (the red curve) is the closest to the bisector of the quadrant, i.e. the probability predicted are the closest to the verifying conditional frequencies. Even if the refinement distribution for all the algorithms shows particularly high values of $Pr(c^{(1)})$, for LR and NGR the difference between this frequency and the others $(Pr(c^{(2)}), ..., Pr(c^{(5)}))$ is slightly less marked. In the second week, LR is again the closest curve to the bisector for all the intervals except $c^{(5)} = 0.9$, where the slope of the red curve decreases significantly. The refinement distribution shows the same behavior seen for the previous week, but this time the difference between the first column and the others for LR and NGR is even less marked. In the last two weeks LR and NGR follow the same strange trend: the slope of the curve diminishes with the increasing of $c^{(i)}$, hinting to the presence of a conditional bias. This behavior is less marked for NGR. The DMO methods curves have a more constant slope, and particularly in the last week they are closer to a horizontal line, being therefore more similar to predictions based on climatology. It is interesting to notice that, while the refinement distribution for these DMO techniques keep approximately the same shape across the four forecasts weeks, for the two regressions we see a gradual shift of the maximum from $c^{(1)}$ to $c^{(2)}$. The behavior just described can be considered valid for both terciles.

In conclusion, we can consider logistic regression the most promising algorithm for extracting tercile probabilities for the 2-m temperature anomalies from our dataset. In Chapter 4 we showed how LR applied to the multi-model outperforms also the same algorithm used in the two single models separately. This suggests that by combining the two fields together we can obtain more skillful probabilistic forecast.

We can ask ourselves the last question: "are further improvements possible?" As shown with the learning curve (again in Chapter 4, Figure 4.15), the chosen

w	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{DV})}$	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{TPP})}$	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{LR})}$	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{NGR})}$
1	0.64	0.65	0.68	0.62
2	0.27	0.29	0.34	0.31
3	0.09	0.11	0.20	0.16
4	0.05	0.07	0.17	0.14

(a) Northern Hemisphere

w	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{DV})}$	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{TPP})}$	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{LR})}$	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{NGR})}$
1	0.63	0.64	0.68	0.60
2	0.30	0.32	0.37	0.34
3	0.15	0.17	0.25	0.22
4	0.08	0.11	0.20	0.18

(b) Southern Hemishpere

w	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{DV})}$	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{TPP})}$	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{LR})}$	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{NGR})}$
1	0.47	0.49	0.55	0.44
2	0.29	0.31	0.39	0.31
3	0.22	0.24	0.32	0.25
4	0.19	0.21	0.30	0.23

(c) Equatorial Belt

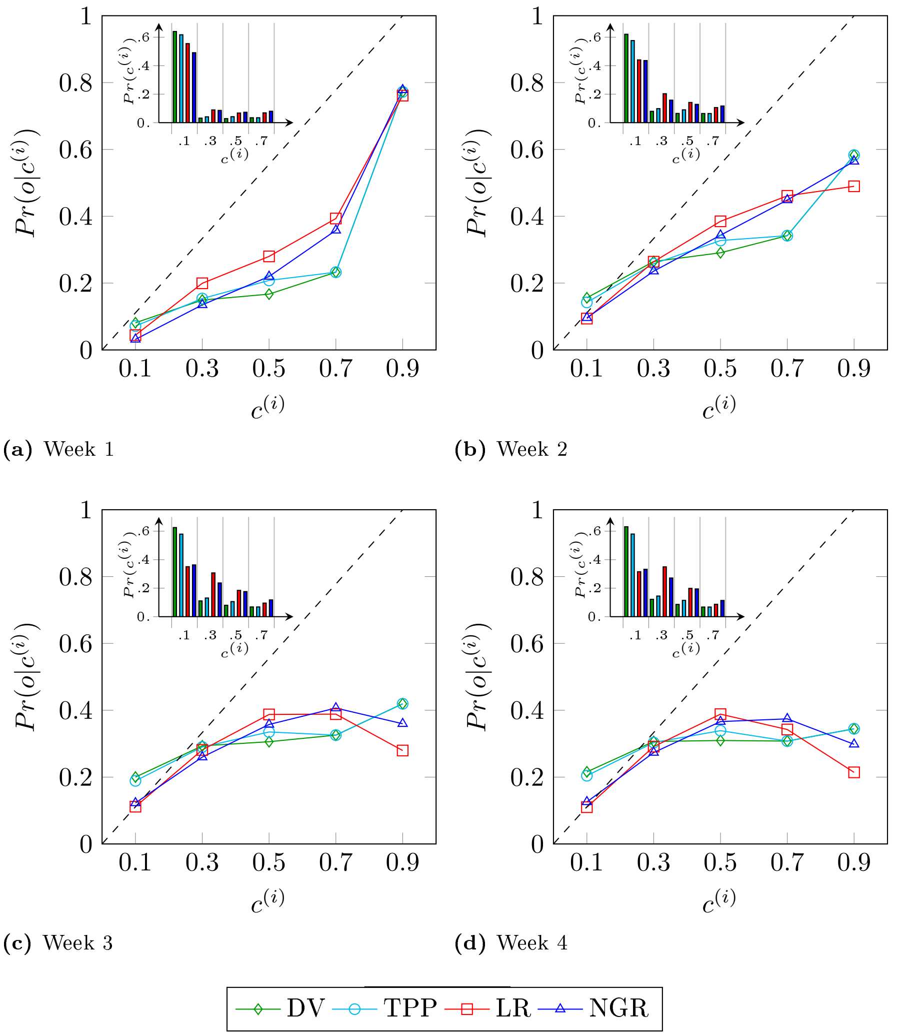
w	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{DV})}$	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{TPP})}$	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{LR})}$	$\boldsymbol{RPSS}_{\text{MM}}^{(\text{NGR})}$
1	0.66	0.67	0.69	0.63
2	0.19	0.21	0.27	0.23
3	-0.06	-0.02	0.11	0.09
4	-0.07	-0.04	0.10	0.08

(d) Europe

Table 6.3: Comparison of the ranked probability skill score for the four method tested, averaged over the 18 validation winters. Again, the four table present the spatial average over the four different regions and the first column shows, in blue, the week for the entire row. The value corresponding to the best performances (the highest ones) for each row is highlighted in red.

variant (α) has probably reached its maximum performances: the scores will not likely improve increasing the dimension of the dataset. The usage of additional features such as the ensemble standard deviation or its product with the ensemble mean does not solve the problem. Obviously, there are countless possibilities that can be tried as input features for LR, therefore we cannot confidently conclude that its results cannot be further improved. Nevertheless, the possibility we tested are the most widely used in the scientific literature and we believe that a reasonable choice for improving the scores is to use a more powerful algorithm, capable of representing non-linear equiprobability hyper-surfaces in the feature space. Such algorithm would require a careful choice of predictors and it is not unlikely that our dataset is too small for avoiding overfitting problems.

The choice of its implementation is not a trivial one. Although it will probably requires greater computational resources and a significant amount of work for identifying the optimal version, it is not certain that additional predictability exists in our data and, in the optimistic case, we do not know how much the forecast skill can potentially be improved.



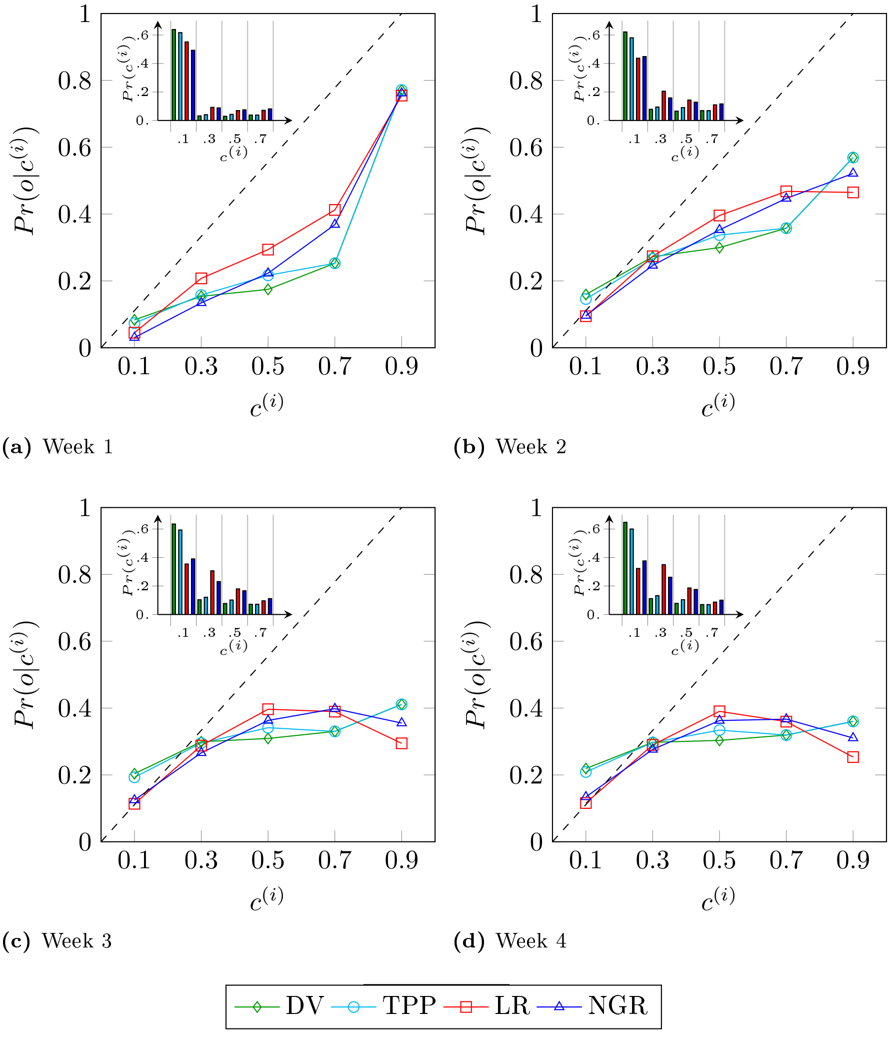
(a) Week 1

(b) Week 2

(c) Week 3

(d) Week 4

Figure 6.1: Reliability diagrams for the four methods, applied to the multi-model an computed for the lower tercile. The four panels refer to the forecast weeks. The values refer to an average over the whole globe and over the 18 validation winters. Each color refers to a different algorithm, both in the calibration function and in the refinement distribution: green for the democratic voting (DV), cyan for Tukey plotting position (TPP), red for logistic regression (LR), and blue for nonhomogeneous Gaussian regression.



(a) Week 1

(b) Week 2

(c) Week 3

(d) Week 4

Figure 6.2: As in Figure 6.1, but for the upper tercile.

Bibliography

Subseasonal to seasonal prediction research implementation plan. Technical report, 2013.

Scott Applequist, Gregory E. Gahrs, Richard L. Pfeffer, and Xu-Feng Niu. Comparison of methodologies for probabilistic quantitative precipitation forecasting, 2002.

Mark P. Baldwin and Timothy J. Dunkerton. Stratospheric harbingers of anomalous weather regimes. *Science*, 294(5542):581–584, 2001. ISSN 0036-8075.

Mark P. Baldwin, David B. Stephenson, David W. J. Thompson, Timothy J. Dunkerton, Andrew J. Charlton, and Alan O'Neill. Stratospheric memory and skill of extended-range weather forecasts. *Science*, 301(5633):636–640, 2003. ISSN 0036-8075. doi: 10.1126/science.1087143.

Anthony G Barnston and Robert E Livezey. Classification, seasonality and persistence of low-frequency atmospheric circulation patterns. *Monthly weather review*, 115(6):1083–1126, 1987.

Vilhelm Bjerknes. *Das Problem der Wettervorhersage: betrachtet vom Standpunkte der Mechanik und der Physik*. 1904.

R. Buizza and T. N. Palmer. Impact of ensemble size on ensemble prediction. *Monthly Weather Review*, 126(9):2503–2518, 1998.

Roberto Buizza, Jean-Raymond Bidlot, Nils Wedi, Manuel Fuentes, Mats Hamrud, Graham Holt, and Frederic Vitart. The new ecmwf vareps (variable resolution ensemble prediction system). *Quarterly Journal of the Royal Meteorological Society*, 133(624):681–695, 2007.

Kenneth S Butcher, Linda D Crown, and Elizabeth J Gentry. *The International System of Units (SI): Conversion Factors for General Use*. US Department of Commerce, Technology Administration, National Institute of Standards and Technology, 2006.

Mingyue Chen, Wanqiu Wang, Arun Kumar, Hui Wang, and Bhaskar Jha. Ocean surface impacts on the seasonal-mean precipitation over the tropical indian ocean. *Journal of Climate*, 25(10):3566–3582, 2012.

D. P. Dee, S. M. Uppala, A. J. Simmons, P. Berrisford, P. Poli, S. Kobayashi, U. Andrae, M. A. Balmaseda, G. Balsamo, P. Bauer, P. Bechtold, A. C. M.

Beljaars, L. van de Berg, J. Bidlot, N. Bormann, C. Delsol, R. Dragani, M. Fuentes, A. J. Geer, L. Haimberger, S. B. Healy, H. Hersbach, E. V. Hólm, L. Isaksen, P. Kållberg, M. Köhler, M. Matricardi, A. P. McNally, B. M. Monge-Sanz, J.-J. Morcrette, B.-K. Park, C. Peubey, P. de Rosnay, C. Tavolato, J.-N. Thépaut, and F. Vitart. The era-interim reanalysis: configuration and performance of the data assimilation system. *Quarterly Journal of the Royal Meteorological Society*, 137(656):553–597, 2011.

Bradley Efron and Gail Gong. A leisurely look at the bootstrap, the jackknife, and cross-validation. *The American Statistician*, 37(1):36–48, 1983.

J. B. Elsner and C. P. Schmertmann. Assessing forecast skill through cross validation. *Weather and Forecasting*, 9(4):619–624, 1994.

Xiouhua Fu, Bin Wang, Duane E. Waliser, and Li Tao. Impact of atmosphere–ocean coupling on the predictability of monsoon intraseasonal oscillations. *Journal of the Atmospheric Sciences*, 64(1):157–174, 2007.

Tilmann Gneiting, Adrian E. Raftery, Anton H. Westveld III, and Tom Goldman. Calibrated probabilistic forecasting using ensemble model output statistics and minimum crps estimation. *Monthly Weather Review*, 133(5):1098–1118, 2005.

Renate Hagedorn, Thomas M. Hamill, and Jeffrey S. Whitaker. Probabilistic forecast calibration using ecmwf and gfs ensemble reforecasts. part i: Two-meter temperatures. *Monthly Weather Review*, 136(7):2608–2619, 2008.

Thomas M. Hamill, Jeffrey S. Whitaker, and Xue Wei. Ensemble reforecasting: Improving medium-range forecast skill using retrospective forecasts. *Monthly Weather Review*, 132(6):1434–1447, 2004.

Harry H. Hendon, Brant Liebmann, Matthew Newman, John D. Glick, and J. E. Schemm. Medium-range forecast errors associated with active episodes of themadden–julian oscillation. *Monthly Weather Review*, 128(1):69–86, 2000.

Marika M. Holland, David A. Bailey, and Steve Vavrus. Inherent sea ice predictability in the rapidly changing arctic environment of the community climate system model, version 3. *Climate Dynamics*, 36(7):1239–1253, 2011. ISSN 1432-0894.

James R. Holton, , and Gregory J. Hakim, editors. *An Introduction to Dynamic Meteorology (Fifth Edition)*. Academic Press, Boston, fifth edition edition, 2013.

Stephen J. Wright Jeorge Nocedal. *Large-Scale Unconstrained Optimization*, pages 164–192. Springer New York, New York, NY, 2006. ISBN 978-0-387-40065-5.

T. Jung, M. J. Miller, and T. N. Palmer. Diagnosing the origin of extended-range forecast errors. *Monthly Weather Review*, 138(6):2434–2446, 2010.

T. Jung, F. Vitart, L. Ferranti, and J.-J. Morcrette. Origin and predictability of the extreme negative nao winter of 2009/10. *Geophysical Research Letters*, 38(7), 2011. ISSN 1944-8007. L07701.

Thomas Jung and Frederic Vitart. Short-range and medium-range weather forecasting in the extratropics during wintertime with and without an interactive ocean. *Monthly Weather Review*, 134(7):1972–1986, 2006.

Eugenia Kalnay. *Atmospheric modeling, data assimilation and predictability.* Cambridge University Press, 2003.

Alexander Kann, Christoph Wittmann, Yong Wang, and Xulin Ma. Calibrating 2-m temperature of limited-area ensemble forecasts using high-resolution analysis. *Monthly Weather Review*, 137(10):3373–3387, 2009.

Arun Kumar, Mingyue Chen, and Wanqiu Wang. An analysis of prediction skill of monthly mean climate variability. *Climate Dynamics*, 37(5):1119–1131, 2011. ISSN 1432-0894.

William K-M Lau and Duane E Waliser. *Intraseasonal variability in the atmosphere-ocean climate system.* Springer Science & Business Media, 2011.

G. S. Lehmiller, T. B. Kimberlain, and J. B. Elsner. Seasonal prediction models for north atlantic basin hurricane location. *Monthly Weather Review*, 125(8): 1780–1791, 1997.

Hai Lin and Gilbert Brunet. Impact of the north atlantic oscillation on the forecast skill of the madden-julian oscillation. *Geophysical Research Letters*, 38(2), 2011. ISSN 1944-8007. L02802.

Hai Lin and Zhiwei Wu. Contribution of the autumn tibetan plateau snow cover to seasonal prediction of north american winter temperature. *Journal of Climate*, 24(11):2801–2813, 2011.

Hai Lin, Gilbert Brunet, and Jacques Derome. Intraseasonal variability in a dry atmospheric model. *Journal of the Atmospheric Sciences*, 64(7):2422–2441, 2007.

Hai Lin, Gilbert Brunet, and Jacques Derome. An observed connection between the north atlantic oscillation and the madden–julian oscillation. *Journal of Climate*, 22(2):364–380, 2009.

Hai Lin, Gilbert Brunet, and Juan Sebastian Fontecilla. Impact of the madden-julian oscillation on the intraseasonal forecast skill of the north atlantic oscillation. *Geophysical Research Letters*, 37(19), 2010.

Edward N. Lorenz. Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20(2):130–141, 1963.

Edward N. Lorenz. Atmospheric predictability as revealed by naturally occurring analogues. *Journal of the Atmospheric Sciences*, 26(4):636–646, 1969.

Edward N Lorenz. The essence of chaos seattle, 1993.

K. Miyakoda, G. D. Hembree, and R. F. Strickler. Cumulative results of extended forecast experiments i: Model performance for winter cases. *Wea. Rev*, pages 10–1175, 1972.

W. A. Müller, C. Appenzeller, F. J. Doblas-Reyes, and M. A. Liniger. A debiased ranked probability skill score to evaluate probabilistic ensemble forecasts with small ensemble sizes. *Journal of Climate*, 18(10):1513–1523, 2005.

Jerome Namias. The annual course of month-to-month persistence in climatic anomalies. *Bull. Amer. Meteor. Soc*, 33(7):279–285, 1952.

Pallav Ray and Chidong Zhang. A case study of the mechanics of extratropical influence on the initiation of the madden–julian oscillation. *Journal of the Atmospheric Sciences*, 67(2):515–528, 2010.

Adam A. Scaife, Jeff R. Knight, Geoff K. Vallis, and Chris K. Folland. A stratospheric influence on the winter nao and north atlantic surface climate. *Geophysical Research Letters*, 32(18), 2005. ISSN 1944-8007. L18715.

A. J. Simmons, R. Mureau, and T. Petroliagis. Error growth and estimates of predictability from the ecmwf forecasting system. *Quarterly Journal of the Royal Meteorological Society*, 121(527):1739–1771, 1995. ISSN 1477-870X.

Stefan Sobolowski, Gavin Gong, and Mingfang Ting. Modeled climate state and dynamic responses to anomalous north american snow cover. *Journal of Climate*, 23(3):785–799, 2010.

Bart van den Hurk, Francisco Doblas-Reyes, Gianpaolo Balsamo, Randal D. Koster, Sonia I. Seneviratne, and Helio Camargo. Soil moisture effects on seasonal temperature and precipitation forecast scores in europe. *Climate Dynamics*, 38(1):349–362, 2012. ISSN 1432-0894.

Frédéric Vitart. Monthly forecasting at ecmwf. *Monthly Weather Review*, 132 (12):2761–2779, 2004.

Frédéric Vitart. Evolution of ecmwf sub-seasonal forecast skill scores. *Quarterly Journal of the Royal Meteorological Society*, 140(683):1889–1899, 2014.

Frédéric Vitart and Franco Molteni. Simulation of the madden– julian oscillation and its teleconnections in the ecmwf forecast system. *Quarterly Journal of the Royal Meteorological Society*, 136(649):842–855, 2010. ISSN 1477-870X.

Frédéric Vitart, Roberto Buizza, Magdalena Alonso Balmaseda, Gianpaolo Balsamo, Jean-Raymond Bidlot, Axel Bonet, Manuel Fuentes, Alfred Hofstadler, Franco Molteni, and Tim N Palmer. The new vareps-monthly forecasting system: A first step towards seamless prediction. *Quarterly Journal of the Royal Meteorological Society*, 134(636):1789–1799, 2008.

Joshua S. Watson and Stephen J. Colucci. Evaluation of ensemble predictions of blocking in the ncep global spectral model. *Monthly Weather Review*, 130 (12):3008–3021, 2002.

Andreas P. Weigel, Mark A. Liniger, and Christof Appenzeller. The discrete brier and ranked probability skill scores. *Monthly Weather Review*, 135(1): 118–124, 2007a.

Andreas P. Weigel, Mark A. Liniger, and Christof Appenzeller. Generalization of the discrete brier and ranked probability skill scores for weighted multimodel ensemble forecasts. *Monthly Weather Review*, 135(7):2778–2785, 2007b.

Andreas P. Weigel, Daniel Baggenstos, Mark A. Liniger, Frédéric Vitart, and Christof Appenzeller. Probabilistic verification of monthly temperature forecasts. *Monthly Weather Review*, 136(12):5162–5182, 2008.

Jeffrey S. Whitaker, Xue Wei, and Frédéric Vitart. Improving week-2 forecasts with multimodel reforecast ensembles. *Monthly Weather Review*, 134(8):2279–2284, 2006.

Daniel S. Wilks. Comparison of ensemble-mos methods in the lorenz 9́6 setting. *Meteorological Applications*, 13:243–256, 9 2006.

Daniel S. Wilks. Extending logistic regression to provide full-probability-distribution mos forecasts. *Meteorological Applications*, 16(3):361–368, 2009.

Daniel S. Wilks. *Statistical Methods in the Atmospheric Sciences*. Academic Press, 2011.

Daniel S. Wilks and Thomas M. Hamill. Comparison of ensemble-mos methods using gfs reforecasts. *Monthly Weather Review*, 135(6):2379–2390, 2007.

S. J. Woolnough, F. Vitart, and M. A. Balmaseda. The role of the ocean in the madden–julian oscillation: Implications for mjo prediction. *Quarterly Journal of the Royal Meteorological Society*, 133(622):117–128, 2007. ISSN 1477-870X.

Ping Zhang. Model selection via multifold cross validation. *The Annals of Statistics*, 21(1):299–313, 1993.

Ringraziamenti

Desidero ringraziare tutti coloro che, direttamente o indirettamente, mi hanno aiutato nella scrittura della tesi e, più in generale, hanno avuto un ruolo importante nel percorso di laurea, di cui questo lavoro rappresenta la conclusione.

Ringrazio innanzitutto la Professoressa Silvana Di Sabatino per avermi concesso la possibilità di svolgere la tesi al CNR-ISAC e il Dottor Piero Malguzzi per avermi seguito durante tutto lo studio, indirizzandomi verso le tematiche scientificamente più interessanti e guidandomi nella fase di stesura con le sue fondamentali correzioni e suggerimenti. Vorrei porgere un "grazie" particolarmente sentito al Dottor Daniele Mastrangelo: sono innumerevoli le volte che mi sono recato suo ufficio, anche per i più piccoli dubbi, ricevendo un sostegno costante e soluzioni puntuali ai quesiti posti.

Vorrei inoltre ringraziare la mia famiglia: i miei genitori, che non ringrazierò mai abbastanza per tutto il supporto che mi hanno fornito in questi anni, e la mia sorellina Ghineddu, che oltre a sopportarmi costantemente, mi ha veramente aiutato nella fase di correzione dell'inglese nella tesi.

Ovviamente non posso non menzionare i miei compagni di corso, sempre disponibili a dare una mano quando avevo bisogno: Thomas, col quale avrò trascorso almeno l'80% del tempo fuori e dentro l'università, Alessiofrancescobrunetti, che mi ha sopportato nei nostri progetti strampalati, il burbero Matteo (che sicuramente si lamenterà dell'epiteto), Ponzy, Giovanni, Jacopo, Sara, Barbi, Arianna (a cui avrei dovuto "killare" molti più processi sul cluster), Zonny e Giulio. Senza ombra di dubbio, è merito loro se ricorderò questo percorso di laurea come uno dei periodi più belli di sempre. Proseguo con tutti gli altri amici che mi hanno supportato durante questo percorso, in ordine casuale come al mio solito: Berta e David, che mi sono stati vicini nonostante le centinaia di km che ci separano, Minetti, Nicola e Daddario, che hanno rallegrato i (purtroppo) pochi momenti passati a Pianella, Pigna e Annachiara, sempre disponibili ad ascoltarmi, Ludovica, per tutte le corse, i biscotti e i film che mi hanno aiutato a superare anche i momenti più difficili in questo periodo, e Bellitti, la fonte della mia grande passione per la programmazione, fondamentale in tutto il mio percorso di laurea.

Infine, voglio ringraziare tutti gli insegnanti che hanno svolto un ruolo fondamentale nella mia formazione, ed in particolare la Maestra Flora delle scuole elementari, la prima a credere veramente nelle mie capacità, il Professor Roberto Della Guardia che mi ha fatto veramente apprezzare la fisica e la matematica, e il Professor Claudio Della Volpe dell'Università di Trento, che ha fatto nascere il mio interesse verso la fisica dell'atmosfera.