Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Home Affairs FDHA
Federal Office of Meteorology and Climatology MeteoSwiss

Technical Report No. 250

Forecasting probabilistic elements of TAF based on COSMO-2 model

Maëlle Zimmermann

ISSN: 2296-0058

Technical Report MeteoSwiss No. 250

Forecasting probabilistic elements of TAF based on COSMO-2 model

Maëlle Zimmermann

Recommended citation:
Maëlle Zimmermann: 2014, Forecasting probabilistic elements of TAF based on COSMO-2 model, *Technical Report MeteoSwiss*, 250, 62 pp.

Editor:
Federal Office of Meteorology and Climatology, MeteoSwiss, © 2014

MeteoSwiss
Operation Center 1
8058 Zürich-Flughafen
T +41 58 460 91 11
www.meteoswiss.ch

Abstract

In aviation meteorology, a Terminal Airport Forecast (TAF) is a concise weather forecasting message aimed at airmen and airline managers, which contains information on very local future weather conditions. These reports are used worldwide and play a crucial role in flight planning, since adverse meteorological conditions can cause great disruptions, affecting the efficiency and safety of the airline traffic. Elements such as low visibility on the runway or strong gusts of wind have to be accurately predicted in order to optimize the traffic.

While TAFs for each airport are currently manually written by human forecasters, recent efforts have been made towards automatizing this process. Although human forecasters have an expert meteorological knowledge and usually produce very reliable forecasts, the cost of having resources dedicated to this task daily is substantial, and prevents many airports from effectively delivering TAF reports. In this context, several meteorological institutions have started to develop forecasting methods aiming specifically at producing some of the elements which constitute a TAF report, thus providing guidance to forecasters.

The focus of this work is the probabilistic forecasts which are part of a TAF, indicating uncertain changes in weather patterns with an approximate probability. To produce such forecasts, an essential tool is Model Output Statistics (MOS), a statistical technique which post-processes raw output from Numerical Weather Prediction (NWP) models. MOS comprehend a wide array of statistical models, but the approach which is suited to derive probabilities is logistic regression. In this work, we present the results of a study applying logistic regression models on numerical forecasts from the COSMO-2 model, in order to obtain probabilistic forecasts for rare meteorological events concerning a given set of parameters.

We assert that COSMO-2 provides good data which is suited to derive TAF probabilities, and we develop methods which have the potential to improve the post-processing framework in place at Meteoswiss. In particular, we apply strategies that allow to deal better with rare event data, one of the causes of unreliable automatic TAF forecasts. Our statistical regression models are also designed to better discern weather patterns which lead to rare meteorological phenomenon, which makes them especially suited to predict typical adverse events that TAFs need to foresee.

Contents

Abstract 5

1 Introduction 8

2 Aviation Meteorology 10

3 Statistical forecasting methods 12
3.1 Post-processing the output of NWP models . . . 12
3.2 Logistic regression as a post-processing method . . . 14
3.3 Estimating a logistic regression model . . . 16
3.4 Algorithms for predictor selection . . . 17
3.4.1 Forward selection based on correlation examination . . . 18
3.4.2 Forward selection based on likelihood ratio test . . . 18
3.4.3 Stepwise selection based on likelihood ratio test . . . 19
3.5 Verification methods . . . 20
3.5.1 The Calibration-Refinement factorization table . . . 21
3.5.2 The Reliability diagram . . . 21
3.5.3 The Refinement distribution plot . . . 21

4 Results and Analysis 24
4.1 Wind gusts . . . 24
4.1.1 The data . . . 24
4.1.2 The regression models studied . . . 25
4.1.3 Predictor selection . . . 25
4.1.4 Coefficients estimation . . . 26
4.1.5 Forecast verification . . . 29
4.1.6 Rescaling the forecasts . . . 33
4.1.7 Model improvements through predictor transformation . . . 33
4.2 Visibility . . . 36
4.2.1 Approach . . . 36
4.2.2 The data . . . 36
4.2.3 The regression models studied . . . 37
4.2.4 Predictor selection . . . 38
4.2.5 Coefficients estimation . . . 39
4.2.6 Forecast verification . . . 41
4.2.7 Model improvement through data sampling strategies . . . 43
4.3 Thunderstorms . . . 46
4.3.1 The approach . . . 46
4.3.2 The data . . . 46
4.3.3 The regression models studied . . . 47
4.3.4 Predictor selection . . . 47
4.3.5 Coefficient estimation . . . 48
4.3.6 Forecast verification . . . 48
4.3.7 Model improvements through variable transformation strategies . . . 49

4.3.8 Model improvements through data sampling strategies 50

5 Options for implementation **52**
5.1 COSMO-MOS 52

6 Conclusion **54**

Abbreviations **56**

List of Figures **57**

List of Tables **58**

References **59**

Acknowledgment **60**

1 Introduction

In aviation meteorology, a Terminal Airport Forecast (TAF) is a concise weather forecast message written in an international format aimed at pilots and airport authorities, containing information on future local weather conditions which are relevant for flight planning. TAFs disclose forecasts for a wide range of meteorological parameters, for example temperature or average wind speed and direction, and also indicate the occurrence of phenomenon such as thunderstorm or snowfall. They aim to predict the evolution of the meteorological conditions at the airport during a day-long time period, and especially changes in weather patterns which are considered significant.

Producing TAFs which respect international norms is part of the role played by Meteoswiss in ensuring air traffic safety from a meteorological point of view. The good quality of such forecasts is crucial to ensure that adverse weather conditions can be anticipated and managed by both airports and pilots. Nowadays, TAFs are manually written by human forecasters, usually within the airport complex to which the TAF applies. This task is performed on a regular basis, as airports must deliver every three hour TAFs valid for a 24h to 30h time period. Usually, when generating a TAF, forecasters base their predictions on 2-D plots, diagrams or values of parameters forecasted by Numerical Weather Prediction (NWP) models. However, these models are often too crude to account for tiny variations due to local topography, and do not directly deliver forecasts for complex phenomenon or parameters related to aviation, leaving a lot of inference work to the forecaster. In this respect, some techniques have been developed since the 1970s to provide more complete and aviation-specific meteorological products, serving as guidance for human forecasters, and heading towards an automatization of the process of writing a TAF.

A typical element in a TAF which is not directly produced by NWP models is the description of uncertain weather conditions through probabilistic forecasts. The syntax of a TAF includes the use of probabilities of 30 and 40 percents to indicate uncertain temporary changes in conditions, such as visibility losses or variations in wind speed. The aim of this work is to further develop methods which deliver probabilistic forecasts concerning relevant parameters for TAFs, and thus could be used as guidance for forecasters. We focus our work on specific items of the TAF, namely occurrences of intermittent and rare events which have a significant impact on air traffic management, targeting specifically wind gusts, visibility and thunderstorms.

In order to obtain such probabilistic forecasts, we apply and further develop Model Output Statistics (MOS) techniques. MOS encompass a wide range of statistical methods which are used to post-process outputs of NWP models, in order to correct their bias or to derive further information from their forecasts. In combining the strengths of both numerical and statistical models, MOS are widely used in weather forecasting, and account for local climatology and variations in the weather. In this work, we

use a statistical post-processing tool which is especially suited to quantify uncertainty into probabilistic forecasts, called logistic regression. We choose to apply these methods to the forecasts of COSMO-2, one of the numerical models currently used at Meteoswiss. Our goal is to show that COSMO-2 output can be post-processed into reliable probabilistic forecasts, to improve the current post-processing techniques, and to work towards providing a framework to compute daily MOS computations at Meteoswiss.

We start by giving an introduction to the context of our work: we explain the role played by Terminal Airport Forecasts in aviation meteorology, then we give an overview of the operating forecasting system at Meteoswiss and the place of post-processing in this procedure. We then detail the methodology and statistical tools we use. Starting with a mathematical introduction to logistic regression models, we move on to describe the specific features of the models we develop, in particular the automatic selection of appropriate predictors, and specific model improvement strategies. Also, we explain how we process to statistical verification of the forecasts delivered by our logistic regression models. In the Results section, we apply the methods to a set of data, and analyze the reliability of the models we built. Finally, we give perspectives of implementation in the operational routine of MeteoSwiss and draw conclusions.

2 Aviation Meteorology

Without the knowledge of meteorology we dispose of nowadays, the global air traffic we observe would be impossible. Reliable and precise weather forecasts are necessary to handle efficiently airline traffic and to prepare to face adverse meteorological conditions. Meteoswiss, responsible for air traffic security from a meteorological perspective, maintains a staff of observers and forecasters on a 24/7 basis, whose role is to deliver meteorological information tailored for the needs of the aviation community.

The products delivered by the central of Meteoswiss based at Zürich-Kloten airport include half-hourly messages transmitting observed meteorological data in an international format (METAR). To complement the METAR, other reporting messages announce weather forecasts rather than current weather, such as the TREND or the TAF. While the TREND announces changes in local weather conditions in the next two hours, the TAF delivers forecasts for a 24h to 32h horizon. The long forecast horizon of the TAF makes it a very valuable tool for airports and pilots to plan and optimize airline traffic. As an example, SWISS uses the information contained in the TAF to compute the best itinerary and maximum weight at take-off for planes, thus deciding on the quantity of fuel on board. The air traffic control service Skyguide also uses these forecasts to determine take-off and landing procedures.

TAFs are brief messages but contain large amounts of information. A vast array of parameters are predicted, wind speed and direction, temperature, cloud cover and height, visibility, precipitation amounts, as well as other weather descriptors. The aim of the TAF is to describe the expected evolution in time of these parameters, and to announce significant changes in weather conditions. What constitutes a change in weather pattern worth mentioning in a TAF is regulated by the International Civil Aviation Organization (ICAO). It is a relatively complex set of rules which differ for each parameter.

The TAF has a very stringent format which allows it to be read worldwide, of which we give below an example. This is a TAF emitted by Zürich-Kloten airport, on the 02.05.2013 at 5:25 UTC.

> LSZH 020525Z 0206/0312 VRB03KT 5000 SCT015 BKN025 TX20/0215Z TN11/0206Z TN10/0305Z PROB40 TEMPO 0206/0207 4000 BR BKN014 BECMG 0207/0210 05005KT BECMG 0211/0214 SCT030TCU PROB40 TEMPO 0216/0222 22015G27KT 4000 TSRA SCT030CB BKN040 BECMG 0300/0303 BKN010 PROB40 TEMPO 0300/0307 VRB02KT 4000 BR BECMG 0307/0310 05005KT SCT025 TEMPO 0310/0312 SCT025TCU.

A TAF always starts with the current state of the weather at the airport at the time of emission, which is followed by change groups indicating modification of the current conditions. For changes which are uncertain, the TAF includes change groups indicated by the abbreviation PROB which contain probabilistic forecasts, written in red in the above example. According to the ICAO rules, PROB is followed by a percentage, which can be either 30 or 40, as an informal indication of how likely the following events are. Therefore, these probabilities are the result of a convention, and not of an actual

computation. Going back to the example, the first red group reads: 40% probability that on the 2nd between 6 and 7 UTC, the visibility lowers to 4000, mist appears under broken clouds at height 1400 feet. The probabilistic elements of a TAF thus announce a possible temporary change in weather pattern, occurring within a short time period, and can concern one or several parameters.

In this work, we aim to automatically produce guidance for TAF probabilistic forecasts, and we target three specific parameters reported in the TAF: wind gusts, visibility and thunderstorms. Thunderstorms are simply reported in a TAF with the descriptor TS. In the case of visibility of wind gusts, a modification is indicated in the TAF if the value changes (deteriorates or improves) and reaches certain specific thresholds. For simplification purposes, we study the performance of the models with respect to a limited set of threshold values. We choose relatively rarely exceeded thresholds values corresponding to adverse meteorological conditions, as these cases are the most important to aviation. Hence, we analyze probabilistic forecasts that wind gusts exceed 15, 20 and 25 kt, and that visibility falls below 3000, 1500 and 600m.

3 Statistical forecasting methods

3.1 Post-processing the output of NWP models

The numerical forecasting model which is operated and further developed at Meteoswiss is COSMO, “Consortium for Small Scale Modeling”, born from the international collaboration of eight European national weather services, including Switzerland. The COSMO system contains three nested Numerical Weather Prediction (NWP) models. When integrated, all of them produce forecasts for a very vast set of meteorological parameters in their domain. However, like most weather and climate forecasting systems, Meteoswiss combines numerical and statistical models through post-processing, a method illustrated in Figure 1 which takes in the raw output of numerical models and delivers further products. Also called Model Output Statistics (MOS), this method is at the core of weather forecasting, and the key to producing forecasts that can be used as guidance for writing TAFs.

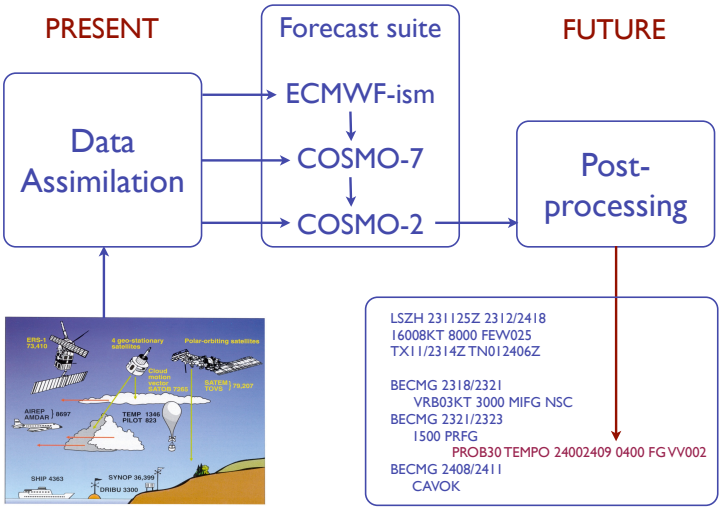


Figure 1: Operational weather forecasting : From data assimilation to post-processing

MOS have several advantages which make them very suitable for this purpose. First, they allow to derive probabilistic forecasts, in which the forecast is a probability instead of a fixed value, from a deterministic numerical model output. Probabilistic forecasts offer the advantage to take into account the uncertainty inherent to forecasting, unavoidable due to the chaotic nature of the atmosphere. Compared to Ensemble Prediction Systems (EPS), a method which derives probabilities from several model integrations based on a disturbed initial condition, MOS is a simple and efficient tool to obtain these probabilities from a single model integration. MOS have other noteworthy positive effects, such as cor-

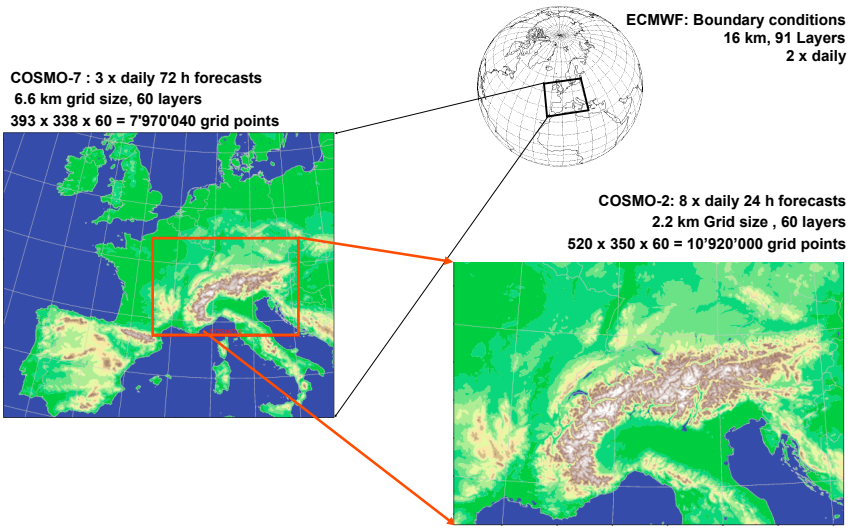


Figure 2: The three nested COSMO numerical weather models

recting bias and systematic errors that occur in numerical models due to the model's approximations or imperfect observations of initial conditions. Also, they enhance the quality of pure numerical forecasts, by including local climatology small-scale effects, which are best captured by statistical relationships. This makes MOS very appropriate to produce site-specific forecasts. Finally, MOS prove useful to derive information concerning meteorological parameters which are not directly forecasted by NWP models, such as visibility on the runway of a given airport.

In this project, the model output statistics we perform is based on forecasts delivered by the most local numerical forecasting model, COSMO-2. It is nested and inherits initial and boundary conditions from a broader COSMO-7 model, itself nested in the global forecasting model from the ECMWF (the European Center for Medium Range Weather Forecast), as Figure 2 shows. The model COSMO-2 covers the alpine area on a rectangular domain with a diagonal reaching from Montpellier in France to Brno in the Czech Republic. Operated at a grid mesh of 2.2 km, COSMO-2 is non-hydrostatic, equipped with comprehensive physics explicitly tuned to cope with alpine topography. Physically, COSMO-2 solves a set of partial differential equations comprehending the three dimensional Navier-Stokes equation, thermodynamic and radiation balances, phases transitions of water governing precipitation, soil-atmosphere interactions and vertical energy exchanges related to terrain roughness in the atmospheric boundary layer. Spatial differential operators are implemented as finite difference schemes, a third order Runge Kutta operator cares for the temporal integration. COSMO-2 is integrated at the CSCS (the Swiss Center for Scientific Computing) eight times a day and delivers on a three hours basis 32-hours forecasts for almost all possible meteorological parameters in the domain.

Conceptually, the COSMO-2 numerical forecasting model operates like a Markovian process : it has no memory, in the sense that the future state on the model is computed based on the present state only, not taking into account the history of past states. COSMO-2 is thus integrated anew for each forecast, making use only of the initial condition delivered by the data assimilation process, which consists of collecting observations available within a short lapse of time before the time 0 of the forecast, as described in Figure 1 (left side).

As opposed to numerical models, Model Output Statistics (MOS) belong to the class of machine learn-

ing algorithms. Based on climatological data, MOS add to numerical forecasting models a kind of memory for specific weather patterns, in particular intermittent, rare or extreme meteorological events. This memory, encapsulated in the vector of regression coefficients β which we will describe in section 3.2, is used to correct the purely numerical forecast or deriving new information from them, taking into account what has been learnt from past data. Therefore, post-processing is based on the combination of two very complementary models, one numerical and the other statistical.

3.2 Logistic regression as a post-processing method

Regression is a statistical method to estimate the relationship between two variables y and x, or more frequently between a dependent variable y and a set of independent variables x_i, for $i = 1...K$. We call the variable y the predictand, and the variables x_i the predictors. The dependency between the vector x and y is modeled by an unknown vector of coefficients β, which has to be estimated from a sample of data. Regression is a technique frequently used in forecasting, as the regression equation yields a predicted value $\hat{y}$ of y given the observed values of the predictor variables x_i and the estimated regression coefficients $\hat{\beta}_i$. If the presumed relationship is linear for example, this gives Equation 1.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_K x_K \tag{1}$$

Regression can act as a statistical post-processing method when coupled with information from numerical models. In this case, the predictors x_i in the regression equation are not observations of meteorological parameters, but rather consist of their forecasted value. In other words, the output obtained from the integration of numerical weather models - in this case COSMO-2 forecasts - is taken as predictor variables in a regression equation. This equation can then be used to correct the forecasted value of one of the meteorological parameters, or to predict another variable.

So far, we understand how regression can enhance a forecast by taking other factors into account in the equation, but how do we actually derive probabilities? This can be achieved by a particular type of regression equation, called logistic regression. It describes a non-linear estimated relationship between a variable y and a set of variables x_i, for $i = 1...K$, and is fit to predict the outcome of categorical variables. In particular, it is commonly used in the case where there are only two available categories, to fit binary predictands. Thus, the logistic regression equation provides a fitted value $\hat{y}$ that is an estimation of the true binary value of the predictand y. Since this fitted value is bounded by the unit interval due to the shape of the equation, it can be interpreted as the estimated probability of occurrence of the event "$y = 1$". In other words, it is a probabilistic forecast.

The logistic regression equation takes a more complex shape than Equation 1. Formally, the binary predictand variable y follows a Bernoulli distribution with parameter π, taking the value 1 with probability π and 0 with probability $1 - \pi$. The parameter π is a function of the vector x, consisting of K predictor variables $x_i, i = 1...K$, a constant, and an unknown vector of coefficients β of size $K + 1$:

$$\pi = \frac{1}{1 + e^{-\sum_i x_i \beta_i}} \tag{2}$$

$$y \sim \text{Bernoulli}(\pi)$$

Given an estimate $\hat{\beta}$ of the value of β, we obtain an estimate $\hat{y}$ of the value of y. Because of the form of the Bernoulli probability mass function, $\hat{y}$ is simply $\hat{\pi}$, the evaluation of expression 2 with $\hat{\beta}$. The value $\hat{y}$ can also be seen as the estimated probability that y takes the value 1.

There exists an alternative definition of this model, from which it draws its name “logistic regression”. In this definition, a continuous variable y^* follows a logistic distribution with mean μ. The parameter μ is a linear function of the predictor variables x_i and the vector β given by

$$\mu = \sum_i x_i \beta_i$$

$$y^* \sim \text{Logistic}(\mu)$$

The predictand, however, is not y^* itself but a discrete binary variable y, taking the value 1 if $y^* > 0$, and the value 0 if $y^* \leq 0$. We observe and are interested in y, not y^*. However, we notice that both definitions are equivalent, since under both models the probability that the predictand y takes the value 1 is given by $\frac{1}{1+e^{-\sum_i x_i \beta_i}}$. This is obvious in the first case, but we can prove that we reach the same result with the alternative definition :

$$\begin{aligned}
\Pr(y = 1 \mid \beta) &= \Pr(y^* > 0 \mid \beta) \\
&= \int_0^\infty f(y^* \mid \mu)\, dy \\
&= \left[\frac{1}{1 + e^{-(y-\mu)}} \right]_0^\infty \\
&= 1 - \frac{1}{1 + e^{\mu}} \\
&= \frac{1}{1 + e^{-\sum_i x_i \beta_i}}
\end{aligned}$$

It now remains to explain how the value of the coefficient vector β, gathering the $K + 1$ unknown parameters of the logistic regression equation, is estimated. Since β represents a kind of long-term memory, it has to be computed from past data. Therefore, we need a database of observations y_j and x_j, for $j = 1, ..., n$, from which to learn patterns. Note that the size of the dataset, denoted by n, is not to be confused with K, the number of predictor variables. The coefficient vector β is then estimated in order to minimize the sum of the squared errors with respect to this dataset. Its estimate $\hat{\beta}$ is computed by maximum likelihood, an analytical method which consists of finding the value of β which maximizes the likelihood function. The likelihood function of the model is a function of β given the observed data, assuming that the observations are independent, and is in this case given by

$$L(\beta \mid y) = \prod_{j=1}^{n} \pi_j^{y_j} (1 - \pi_j)^{y_j}. \tag{3}$$

Intuitively, the likelihood is the probability of observing the given data as a function of β. Thus the most plausible β is the one maximizing the likelihood, in other words the one which explains best the data. Instead of maximizing this function, we maximize its logarithm, called the log-likelihood, since both will

be attained by the same value of β. The problem then simplifies to (*Greene* (1993))

$$\ln L(\beta \mid y) = -\sum_{j=1}^{n} \ln(1 + e^{(1-2y_j)x_j\beta}). \tag{4}$$

Finally, the vector $\hat{\beta}$ is obtained by solving simultaneously $K+1$ equations corresponding to the derivative of Equation 4 with respect to each parameter β_i.

3.3 Estimating a logistic regression model

Statistical post processing requires two stages of computations in order to deliver probabilistic forecasts. The first one consists in building the regression model with the help of past data; the second one in effectively applying the regression model to the latest COSMO-2 output after each NWP model integration, in order to obtain current forecasts. We call the first stage the learning process, and the second one the operational forecasting process, both described in Figure 3.

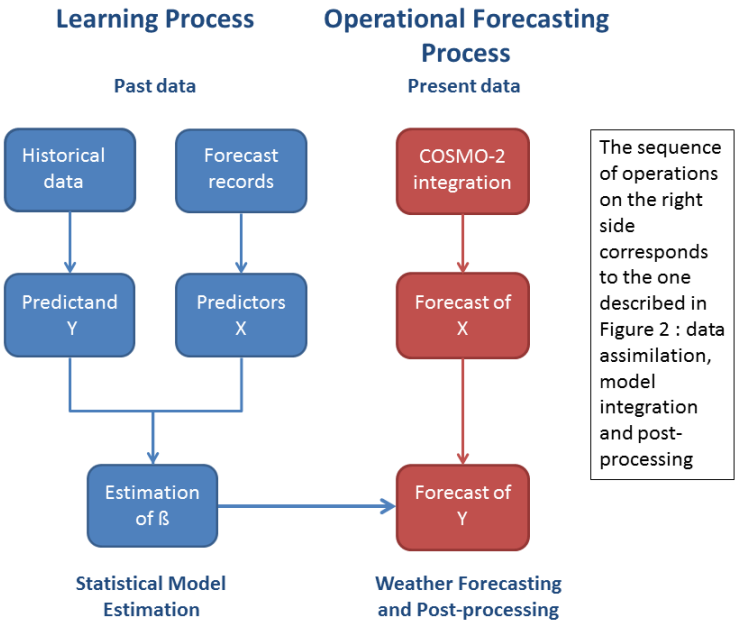


Figure 3: Learning process VS Operational forecasting process

The learning process is thus the task of estimating a logistic regression model from a sample of historical data, and consists itself of several steps. The starting point is to gather historical data concerning the event which we aim to model and predict (e.g. visibility lower than 600m). Two kinds of data are needed. First, a database of historical observations of the meteorological parameter involved (the predictand), and secondly a matching database of archived COSMO-2 forecasts of parameters that could be used as predictors. Additionally, the historical observations need to be transformed according to a certain threshold (corresponding to the event we want to model) to become binary.

In mathematical terms, this defines a binary variable y corresponding to an event to predict, and we aim to perform a regression on y with appropriate predictor variables $\{x_i\}_{i=1,\ldots,K}$. The database of

observations are realizations of the binary variables y stating whether the meteorological event has occurred ($y = 1$) or not ($y = 0$) after observation. Each predictor x_i corresponds to a meteorological parameter which is thought to have influence on the value of y. The predictor database consists of the COSMO-2-forecasted values of these parameters, for all times at which y is observed. In other words, MOS regression equations are developed using numerical forecasts for values of the predictors at the time at which the forecast pertains.

Once a large amount of data is available (typically several years of records), the logistic regression model can be estimated. This consists in selecting appropriate predictors x_i to enter the regression equation, among the available ones in the data, as well as estimating the value of the associated vector of regression coefficients $\hat{\beta}$ by maximum likelihood. The method used to select predictors for the regressions is detailed in section 3.4.

Once a logistic regression equation is developed, it can be used in a daily routine to predict future values of the predictand, in the phase of operational forecasting seen in Figure 3. In other words, the forecasted predictor values from each new model integration are given as input to the regression equation. Then, the estimated probability that the given meteorological event occurs given values of predictors $\{x_i\}_{i=1,\dots,K}$ is given by

$$\hat{\pi} = \frac{1}{1 + e^{-\sum_i x_i \hat{\beta}_i}}. \tag{5}$$

3.4 Algorithms for predictor selection

The main difficulty in developing a regression equation is to select appropriate predictors out of the pool of available candidates. A balance must be found between adding a sufficient number of predictors to obtain a good relationship, and avoiding overfitting the regression equation.

A regression equation is said to be overfit when it loses its ability to forecast once it is used on “independent data”, that is data which has not been used in the equation’s development. This usually happens when too many parameters have been included as predictors. Quite simply, the more predictors are used in a regression, the more degrees of freedom are available, and the better the data points can be approximated by the regression function. However, since the coefficients of the regression equation are computed to minimize specifically the errors of the development data sample, the regression will perform less well when given new independent predictor input, and this phenomenon is aggravated if a too high number of predictors has been used.

In order to avoid overfitting, it is important to have a large set of data at disposal to fit the regression. This will ensure that the estimated coefficients are stable, which means less depending on the sample that was used during development. The regression equation is then less likely to fall apart when used with new data. Also, the larger the sample size, the more predictors can be included and correctly estimated before reaching overfitting.

Screening and selecting predictors is therefore the most crucial step in fitting a regression. Since there is usually a very large number of parameters which could play a role in explaining the phenomenon of interest, the usual method is to create a pool of potentially relevant predictors and to select a subset within this pool. Subselecting allows not only to prevent overfitting, but also to avoid redundancy of information in the predictors, since most of the time meaningful predictors are mutually correlated.

In this project, we automatize the selection of predictors by implementing several algorithms in Matlab which decide how many and which predictors present in the data are sufficient to produce good forecasts.

3.4.1 Forward selection based on correlation examination

This first algorithm is based on Pearson's correlation coefficient, which describes the linear dependency between two variables. It was implemented to select predictors on trial linear regressions, before tackling logistic regression. However, since it requires less calculations than the algorithms which follow, we also tested it to select predictors for logistic regressions, in order to have an idea of how sensible to the algorithm the results were.

The algorithm starts with an empty model, to which significant predictors are added step by step as long as they satisfy a certain criteria. In the first step, the Pearson correlation coefficient between the predictand and each predictor is computed based on the data sample. The predictor which is most linearly correlated with the response variable is selected as x_1, and the corresponding linear one-parameter model is fit to the data.

In further steps, more parameters are added according to their partial correlation with the predictand. Partial correlation at step t between predictor x_i and predictand y is the Pearson correlation between the raw residuals of the linear regression at step $t-1$, that is $y = f(x_1, ..., x_{t-1})$, and the raw residuals of the linear regression $x_i = f(x_1, ..., x_{t-1})$. Raw residuals in a regression equation are the differences between observed and fitted value, also called error terms. Therefore, partial correlation measures how much a variable x_i explains the share of errors which are unaccounted for by the current predictors, when x_i is "cleared" of its linear dependance with the current predictors. The variable x_i that has the highest partial correlation is added to the subset of predictors, and a new regression equation is fitted.

In order not to include all parameters, a stopping criterion is needed, at which point predictors stop being added. In this algorithm, there is no automatic stopping criterion. On the contrary, manual cross-validation serves as decision rule for an approximate stopping point.

When this algorithm was tested to select predictors for logistic regressions, we followed the same steps, but fitted a logistic regression wherever a linear one had been used in the algorithm.

3.4.2 Forward selection based on likelihood ratio test

This second forward algorithm, proposed by *D. W. Hosmer and Sturdivant* (2013), is this time only used to select predictors for logistic regression. It is based on the likelihood ratio test, a statistical test used to compare two nested models, in other words models where one is a restriction of the other.

This test makes the hypothesis that the restricted model is better, called the Null Hypothesis. The statistic which is computed is the likelihood ratio, a measure of how many times more likely the data sample comes from the larger model than the other. The exact formula for the likelihood ratio D is minus twice the difference between each model's log-likelihood $\mathcal{L}$, which is the logarithm of the likelihood L given in Equation (3).

$$D = -2(\mathcal{L}(M_R) - \mathcal{L}(M_U)) \tag{6}$$

This statistic is known to be χ^2 distributed, with as many degrees of freedom as there are additional parameters in the larger model. Knowing the distribution of the test statistic allows us to determine levels of significance: these levels are values for the statistic D for which we can almost surely say that the larger model is better than the restricted one, thus rejecting the Null Hypothesis. Different values for D correspond to different degrees of certainty with which we reject the Null Hypothesis.

This test allows us to implement an algorithm which selects parameters in an ascendent manner, starting with the null model, and comparing always the model with one additional parameter with its restricted version.

The algorithm begins with a univariate analysis of each available predictor. In the first step, each predictor is examined in a trial one-parameter logistic model, and the significance level of each model is reported. The significance level of a model is computed based on the likelihood ratio test statistic D comparing it with the null model (containing no parameter). Therefore, the statistic is χ^2 distributed with 1 degree of freedom. At step 1, this statistic is $-2(\mathcal{L}(0) - \mathcal{L}(M_1^i))$ for the model M_1^i fitted with parameter x_i. The model with the highest value for the statistic is the one chosen at step 1 and is denoted M_1, and its parameter is the first selected predictor.

In each further step t, all possible models containing one additional parameter are fitted, and their likelihood ratio test statistics against the restricted model are again compared. For the model M_t^i containing x_i as additional parameter, the statistic is given by $-2(\mathcal{L}(M_{t-1}) - \mathcal{L}(M_t^i))$, where M_{t-1} is the restricted model valid at step $t-1$. The variable which corresponds to the highest level is selected, and trial logistic regressions continue to be fitted and compared during the next steps.

The stopping criterion is determined by a certain level of significance, below which parameters are not added anymore to the set of predictors. In this case, we pick the 0.05 significance level of the χ_1^2 distribution, for which the value of the likelihood ratio test statistic D is 3.84. The 0.05 level is commonly used in statistics, and it means that we are rejecting the restricted model in favor of the larger one with 95% certainty.

3.4.3 Stepwise selection based on likelihood ratio test

This stepwise algorithm, also proposed by *D. W. Hosmer and Sturdivant* (2013) works in a similar manner as the one previously described, also basing its selection of predictors on the likelihood ratio test. The difference between forward and stepwise selection is that parameters which once entered the model may be removed in further steps. The addition of one parameter to the predictor subset is hence not definitive.

This algorithm can be viewed as an improved version of the forward selection procedure. All steps described in the previous paragraph are implemented as well in this method. However, after each forward selection step, there is also a stage of backtesting, in which all variables included so far are tested again for significance. The backtesting computes all possible regressions in which one of the previously added parameters is removed, and compare them to the current model. Since we are again comparing a model with a restricted version of itself, we can perform this comparison with the likelihood ratio test. We check for each one of these variables the level of the likelihood ratio D, and if one does not reach the significance level, the associated variable is removed from the current model. Stepwise selection therefore alternates between forward steps and backwards steps, until no new parameter can

enter without being immediately expelled.

One may think that once a parameter has entered, it has no reason to be expelled. However, as new parameters gradually enter the equation, the information is spread differently between the available predictors due to their mutual correlation. The coefficient associated with each predictor takes a different estimated value at each step, and therefore a parameter can become irrelevant after the inclusion of another variable.

3.5 Verification methods

Before using a model in the operational routine for prediction, as described in Figure 3, we have to validate it by assessing the quality of its forecasts. Verifying the output delivered by a model is thus an integral part of its development. There are several methods to verify forecasts, depending on whether they are probabilistic or deterministic, and what kind of outcome they predict. In this project, we are dealing with probabilistic forecasts for binary predictands. Such forecasts are harder to verify than deterministic ones, since they contain a share of uncertainty which complicates the process of judging their correctness. A single forecast can therefore not be assessed “right” or “wrong” ; on the contrary, a clear-cut decision can only be made on a large collection of forecasts. Therefore, we will assign a portion of the data to forecast verification, in order to have a number of forecast and observation pairs to compare.

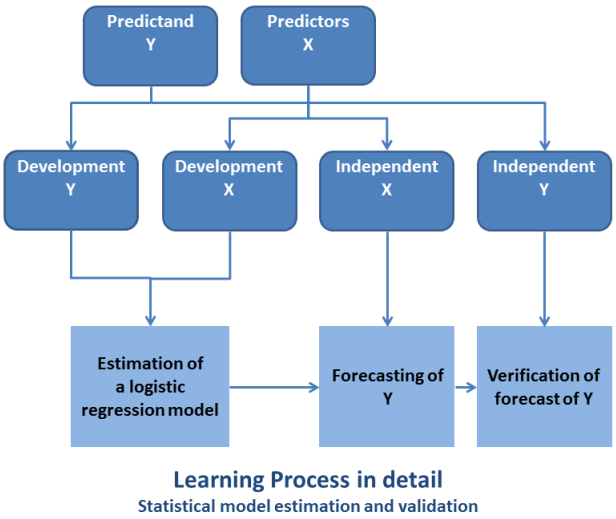


Figure 4: The verification of forecasts within the learning process

The data we use to produce these test forecasts is called “independent data”. Naturally, it must not have been used in the logistic model’s development, since we aim to evaluate the ability of the model to deliver predictions given new data. Once the model has been estimated, we use the independent predictor data to yield forecasts of the predictand, which we compare with the known observed values, as shown in the flow chart in Figure 4.

In the following, we describe statistical measures to compare the probabilistic forecasts of the model

with the observations, as proposed by *Wilks* (2006).

3.5.1 The Calibration-Refinement factorization table

The measure that sums up best the quality of the model is given by the Calibration-Refinement factorization table. The table categorizes the forecasted probabilities in a discrete set of possible values (0,0.1,0.2,...,1), by rounding them to the nearest decimal. For each categorical value y_i, the table shows how often the value was forecasted by the model, and also gives the percentage of times $p(o_1 \mid y_i)$ that the meteorological event did occur in these cases. The main inconvenient of this measure is that numeric tables are tedious to read and do not present intuitively the information content.

3.5.2 The Reliability diagram

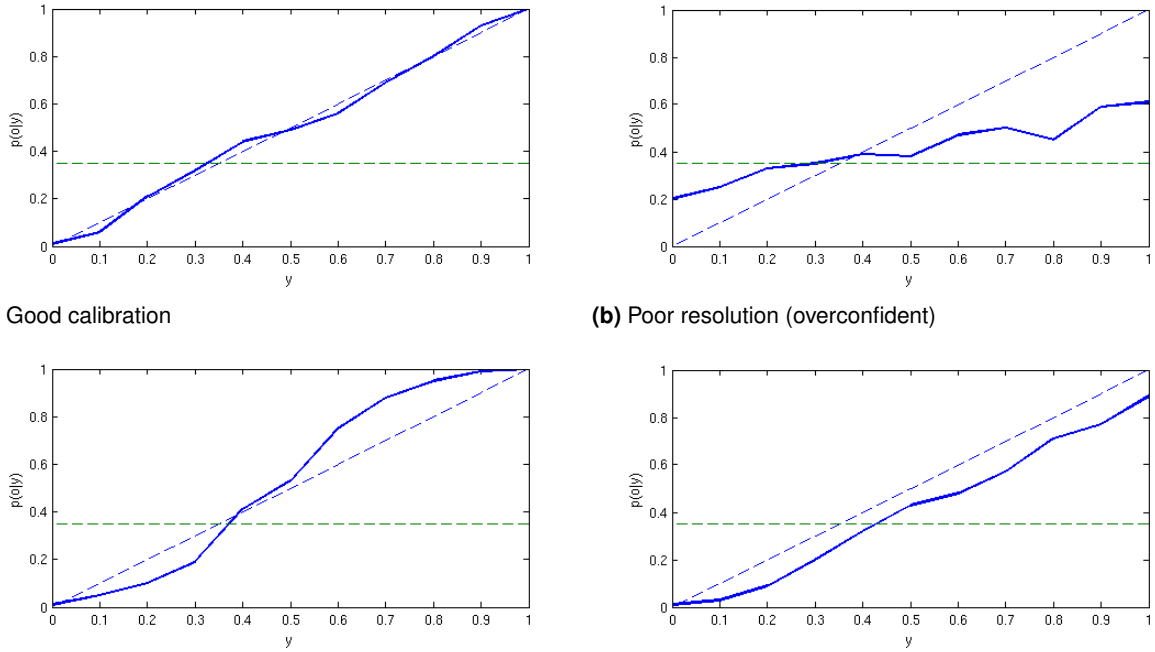
The Reliability Diagram is useful to present some of the information in the Calibration-Refinement table in a more readable manner, through a graphical device. It consists of a graph with on the x-axis all possible rounded-up probabilistic forecast values (0,0.1,0.2,...,1), and on the y-axis, the conditional probability of occurrence of the event given the forecast value, $p(o_1 \mid y_i)$. For each discrete value y_i, the graph therefore compares y_i with the probability $p(o_1 \mid y_i)$, which is equal to the percentage of times that the outcome o_1 (the meteorological event) was observed when it was forecasted with probability y_i. The graph also contain a horizontal line which indicates the climatological average frequency of the event.

The Reliability Diagram thus measures the calibration of the model. When a certain probability of occurrence is forecasted by the model, how often does the forecasted event really occur on average? Ideally, each time a probability p is forecasted by the model, the average occurrence should also be approximately equal to p. Hence, the closer to the 1-1 diagonal line the dots in the graph fall, the more accurate and well-calibrated the model is, as Figure 5a shows. If on the contrary, the graph line stays close to the horizontal line of the climatological average, it means that the observed frequency of the meteorological event does not change much, no matter whether high or low probabilities are predicted. In this case the model is poorly calibrated, as we see in Figure 5b.

3.5.3 The Refinement distribution plot

This plot presents graphically the other half of the information contained in the Calibration-Refinement table. It consists of a histogram of the rounded-up forecasted values. Each column in the plot indicates how many times the model forecasted the corresponding value. It is a measure of the capacity of the model to distinguish with high certainty between events and non-events. Ideally, the values that are most often forecasted should be both very low and very high probabilities. If the histogram is on the contrary centered on the climatological average, the model exhibits low confidence.

Instead of plotting one refinement distribution with the whole data, it is more informative to draw two histograms, one displaying the repartition of forecasts each time the event was observed, and the other when no event occurred. If the model forecasts with confidence, we expect that the events plot



(a) Good calibration

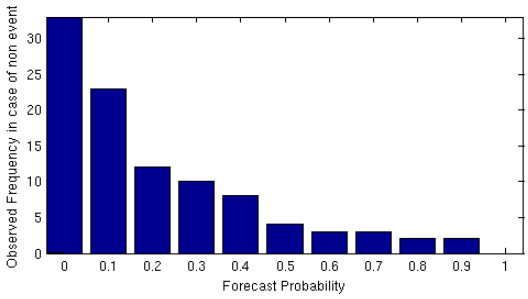
(b) Poor resolution (overconfident)

(c) Good resolution (underconfident)

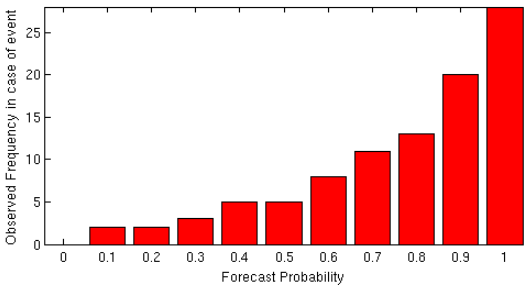
(d) Bias

Figure 5: Example reliability diagrams. The plots (a) to (d) show reliability diagrams corresponding to different models which perform more or less well. In (a) the model is well calibrated and reliable, however (b) gives barely more indication than the climatological average, and (d) yields biased forecasts. The model in (c) has a good predictive ability, and could forecast with more confidence.

will have peaks for the highest possible forecast values, and that the histogram of non-events will be on the contrary centered on low values, like in Figure 6. While it is not always possible to obtain such a clear-cut situation, we hope that the two histograms will be sensibly different, and that globally the one for events will be more right-centered than the one for non-events.



(a) Histogram of non-events



(b) Histogram of events

Figure 6: Example refinement distributions. On the left, the histogram of forecasted values each time the event was observed, on the right the same histogram in the other case. The model described by these plots exhibits good confidence.

4 Results and Analysis

In this section, we study the results of the application of our statistical post-processing methods on three different meteorological parameters. We describe in each case the data used for the study, detail which meteorological events we modeled and under which conditions, and analyze the resulting probabilistic forecasts with the help of the reliability plots and refinement distributions previously mentioned. The statistical analysis and estimation of logistic regression models was implemented in the software MATLAB, a high-level language and interactive environment for numerical computation, visualization, and programming.

An important remark is that the script written in MATLAB enables the estimation of models for several day times, lead times, and varying settings. However, we focus in this section on the analysis of a few specific models for each parameter, which are representative of our methods.

4.1 Wind gusts

The first parameter this work focuses on is wind gusts. The precise value we are interested in is the hourly maximum of the wind's speed. Since wind speed is a relatively well-forecasted COSMO-2 parameter, we expect that we can develop relatively simple and efficient logistic models for our three thresholds of interest: 15, 20 and 25kt.

4.1.1 The data

In order to estimate a reliable model, we need a large database of observations of wind gusts and forecasts of our chosen predictors. We obtain this data from two softwares, Climap and Fieldextra. Climap is a software designed to easily retrieve observations of all meteorological parameters stored in the Data WareHouse (DWH). Fieldextra is a generic tool with many applications, one of which is to generate COSMO forecasts for a given set of dates, locations and parameters.

The predictand data consists in a Climap-generated file listing wind gusts observations at all times of the day for a two-year long period, between May 2011 and May 2013. The predictor data is created by Fieldextra and contains archived COSMO-2 forecasts for a chosen set of parameters, with matching dates. For each hour we have a predictand observation and a matching set of predictor forecasts.

As we already mentioned, we need both"developmental data", in order to estimate the logistic regression models, and "independent data", in order to verify the performance of the models, before they can be used for prediction. We must therefore separate our total data in two different samples. In

order to avoid a seasonal bias in parameter estimation, we decide to include observations from days of both years in the development data, so that we obtain a sample as representative as possible of the different weather patterns. Hence, we use a sample consisting of every other day in the total data to fit the logistic regression, and leave out the remaining days for testing.

4.1.2 The regression models studied

An important meteorological observation is that wind gusts are not induced by the same phenomenons in winter and summer. While in summer, strong gusts of wind occur during storms, they are on the other hand caused by cold fronts and variations of pressure in winter. In order to take this fact into account, we study a division of the data into summer (May to September) and winter (October to April) months and estimate different models for each season.

Additionally, we compare the performance of wind gusts models estimated for different thresholds (15,20 and 25kt), with a frequency of events varying from occasional to very rare.

The forecasts data at our disposal corresponds to the daily 03 UTC integration of the model, with lead time varying from +0h to +23h throughout the day. In practice, this means that the quality of the forecast will vary from morning until evening, the evening forecasts being less reliable due to the advanced lead time. This simulates well the situation of a human forecaster who has to write a TAF for a long forecast horizon based on the current model integration. However, the models should concretely only be used for operational post-processing on a 03 UTC integration.

Although this is an extreme simplification, we choose not to estimate separate models for each time of the day in this study, in order to pally the lack of data. Therefore, the data is aggregated and the models we present forecast wind gusts for any day time. However, we would recommend to estimate different models for both run and day times for more precision, in the practical implementation of these post-processing methods.

4.1.3 Predictor selection

We first decide on a pool of meteorological parameters issued from COSMO-2 output that are likely to have an effect on wind gusts, given in Table 1. From this initial pool of physically relevant parameters, we aim to select a mathematically relevant subset. Each model has its own subset of parameters, and we test all three predictor selection algorithms in each case.

Both algorithms based on the likelihood ratio test yield the same results, independantly of backward verification. The algorithm simply based on correlation examination selects the parameters in a slightly different order. However, the first 2 significant parameters are always identical.

Table 2 displays the predictors selected by the stepwise algorithm for the 6 models we study (3 different thresholds and 2 different seasons), in their order of significance. The main difference between the chosen subsets of predictors in winter and summer is that the parameter WSHEAR_0-3km is always a very important predictor in winter, and never in summer. Since wind shear is a feature indicative of cold fronts, this mathematical result reflects a meteorological reality. Quite expectedly, we also notice that parameters related to wind speed such as FF_10M and VMAX_10M are excellent predictors in almost

	Abbreviation	Parameter
1	VMAX10m03h	Maximum wind gusts at 10m in the last 3h
2	CAPE_MU	Convective available potential energy of most unstable parcel
3	FF_10M	Hourly average wind speed at 10m
4	DD_10M	Hourly average wind direction at 10m
5	PS	Air pressure
6	DURSUN12h	12h Cumulated sun duration
7	WSHEAR_0-3km	Wind shear between surface and 3km
8	V_REL_700	Relative vorticity on pressure surfaces at level 700hPa
9	DELTA_PS	Difference of air pressure between past hour and hour to come

Table 1: Initial pool of parameters given to the predictor selection algorithms

all situations.

However, we also remark some less expected differences between the predictors chosen in summer and winter. Surprisingly, the convective available potential energy CAPE_MU, which is a measure of the atmosphere's instability and potential for storms, is not a always a significant parameter in summer. Another unforeseen difference is that the maximum speed over a 3 hour period, VMAX_10M3h, is a better predictor in summer. Although we expected this parameter to play a crucial role, it is not always significant in winter and comes only second in summer. A possible explanation is that VMAX_10M3h is very correlated with FF_10M, thus entering the model only when no other unrelated predictors bring valuable information.

Comparing the models with respect to the level of the threshold, we conclude that the models for 15 and 20kt are similar. However, a greater change appears in the model for gusts above 25kt, especially in summer. The parameters CAPE_MU and DELTA_PS, which are both storm indicators, gain suddenly importance. This result makes sense, since wind gusts above 25kt are more correlated to storms than lower-speed wind gusts of 15 or 20kt.

To sum up the results, the most important parameters in summer appear to be FF_10M and VMAX_10M, while in winter they are FF_10M and WSHEAR_0-3km. In both case, it is the hourly average wind speed at 10m which is the best predictor of the maximum wind speed reached during the hour.

4.1.4 Coefficients estimation

Once the predictors x_i have been determined, their associated regression coefficients β_i can be estimated by MATLAB by maximum likelihood. This method of estimation consists of maximizing the likelihood function, which expresses the probability of obtaining the sample data as a function of the unknown coefficients. The coefficients satisfying this property are thus called the *maximum likelihood estimators*.

In order to evaluate the goodness of the fit, we examine the values of the estimated coefficients and discuss their coherence. In this analysis, the most important element to observe is the p-value of each

	Wind gusts > 15kt		Wind gusts > 20kt		Wind gusts > 25kt
	Summer model		Summer model		Summer model
1	FF_10M	1	FF_10M	1	FF_10M
2	VMAX_10m03h	2	VMAX_10m03h	2	VMAX_10m03h
3	V_REL_700	3		3	DELTA_PS
4	DD_10M	4		4	DD_10M
5	PS	5		5	CAPE_MU
	Winter model		Winter model		Winter model
1	FF_10M	1	FF_10M	1	FF_10M
2	WSHEAR_0-3km	2	WSHEAR_0-3km	2	WSHEAR_0-3km
3	CAPE_MU	3	CAPE_MU	3	CAPE_MU
4	DD_10M	4	V_REL_700	4	
5	VMAX_10M3h	5	VMAX_10M3h	5	

Table 2: Ordering of predictors resulting from the stepwise selection procedure

coefficient, as it indicates the probability that the predictor has in fact no influence on wind gusts. In other words, it is a measure quantifying the certainty with which the predictor is added to the model, where the lowest value indicates the highest certainty. Secondly, it is also important to look at the sign of the estimated coefficient, as it indicates whether the selected predictor is a contributing or hindering factor in the equation. First, we discuss the models for wind gusts above 15kt. Therefore, we report in Figures 7a and 7b the output of the estimation procedure in both cases, which consists of parameter estimates and summary statistics.

There is a coefficient for each predictor, and an additional intercept coefficient associated to the constant term of the model. The coefficients of parameters FF_10M, VMAX_10M3h, V_REL_700, WSHEAR_0-3km and CAPE_MU are positive. This means that the higher the value of these parameters, the more likely gusts above 15kt become. This corresponds to our expectations, since these parameters are all factors intensifying wind speed. They also all have significantly low p-values, apart from V_REL_700.

On the other hand, the coefficients of parameters DD_10M and PS do not have extremely low p-values, which means that their importance is the model is mathematically questionable. Meteorologically, the inclusion of these variables does not make sense. Wind direction being quantified in degrees, the lowest and highest possible values (in degrees) correspond to the same wind direction, therefore no interpretation can be made about the meaning of the coefficient. Furthermore, storms are characterized by high pressure variations, but it is difficult to draw information about storms from the value of pressure at a single time. We conclude that these two parameters could probably be removed from the model without any change in performance.

The outputs of the estimation results concerning the models for 20 and 25kt respectively show the same characteristics, and we do not detail them.

```
Generalized Linear regression model:
    logit(Y) ~ 1 + FF_10M + VMAX_10M3h + V_REL_700 + DD_10M + PS
    Distribution = Binomial

Estimated Coefficients:
                    Estimate       SE            tStat      pValue
    (Intercept)        21.249       11.781      1.8036      0.071289
    FF_10M             0.9883      0.062121     15.909     5.4665e-57
    VMAX_10M3h         0.12365     0.022503     5.4948      3.912e-08
    V_REL_700           678.73        221.9     3.0587      0.0022227
    DD_10M          -0.0015533    0.00059463    -2.6123      0.0089936
    PS              -0.00027006    0.00012167    -2.2196       0.026449


3672 observations, 3666 error degrees of freedom
Dispersion: 1
Chi^2-statistic vs. constant model: 1.03e+03, p-value = 5.07e-220
```

(a) Summer model

```
Generalized Linear regression model:
    logit(Y) ~ 1 + FF_10M + W_SHEAR_0_3km + CAPE_MU + DD_10M + VMAX_10M3h
    Distribution = Binomial

Estimated Coefficients:
                    Estimate      SE            tStat      pValue
    (Intercept)        -5.5862       0.17224    -32.432    9.7838e-231
    FF_10M              1.0917      0.056184     19.431     4.2126e-84
    W_SHEAR_0_3km      0.085884      0.008104     10.598     3.0526e-26
    CAPE_MU           0.0070818     0.0013753     5.1492     2.6166e-07
    DD_10M           -0.0022351    0.00047972    -4.6592     3.1742e-06
    VMAX_10M3h         0.072424      0.020503     3.5323     0.00041195


5112 observations, 5106 error degrees of freedom
Dispersion: 1
Chi^2-statistic vs. constant model: 2.21e+03, p-value = 0
```

(b) Winter model

Figure 7: Wind models: Outputs of estimation results for wind gusts models above 15kt

4.1.5 Forecast verification

In order to be able to verify the accuracy of the logistic models once they are used for prediction purposes, we now turn to the independent data that we have so far set aside. While the development data was used to compute the regression equations, the testing data now serves as an independent verification sample. The probabilistic forecasts resulting from all 6 models are examined in Figures 8 to 10.

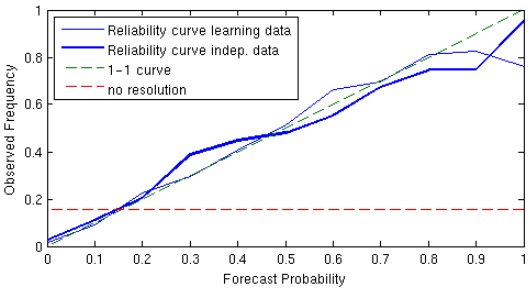
We recall that the reliability diagrams verify how often on average forecasted events occurred. For each possible probability forecasted by the model, the graph indicates the observed frequency of the forecasted event. A well-calibrated model should follow as close as possible the green 1-1 diagonal line. The red horizontal line indicates the average observed frequency of the event, which we call the climatological average. A precision is that the thin blue line represents the performance of development data, while the thick one is drawn from independent data. Since the model is built with development data, the thin blue line is often closer to the diagonal line. However, we are interested in the performance of the model on new data and therefore look and comment on the thick line.

We observe that the models based on winter data appear both more reliable and stable than the summer ones, probably because the sample of data for winter is larger. We also notice a decrease in the quality of forecasts as the level of wind gusts increases for both seasons. When the models aim at predicting gusts above 25kt, the events have a much lower average occurrence than what is forecasted. This is an expected result, as gusts above 25kt are significantly rarer than gusts above 15kt, as can be seen on the climatological average of the plots (in red), and logistic regression on rare events data is known to be difficult to reliably estimate. However, although the line on the reliability plots of Figure 10 is far from being perfectly diagonal, it is still significantly different from the climatological average, meaning that the models already deliver valuable information, even for gusts above 25kt.

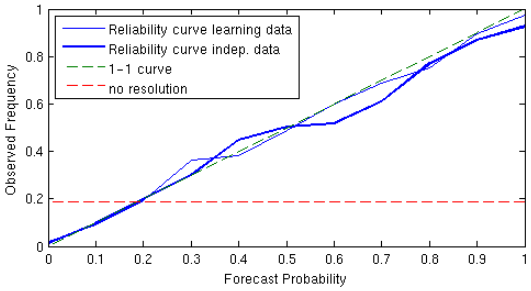
The refinement distribution plots show the number of times each possible rounded-up probability was forecasted by the model, both when wind gusts occurred and when not. It is in this sense a histogram of the distribution of events and non-events by the model. We remind that we expect events to be distributed above the climatological average, and non-events below it.

Given the low climatological average (between 0.02 and 0.2 approximatively), the distributions obtained are good. The plots show that the absence of gusts is well predicted, with a 0 probability delivered by the model the great majority of the time. In addition, wind gusts above the threshold are almost always predicted with a probability higher than their average occurrence, indicating the model's capacity to discern them. However, we notice that the certainty of the predictions decrease as higher wind gusts must be predicted. For gusts of 20 and 25kt, fewer very high probabilities are forecasted, and the model makes less risky forecasts centered on middle values. The superiority of the models in winter is also confirmed by these plots.

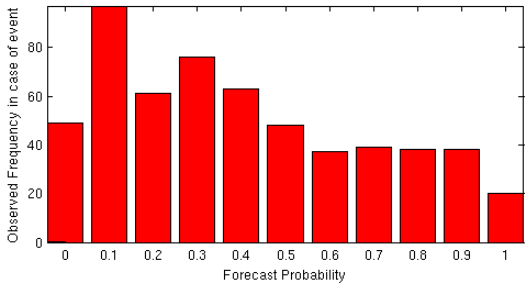
Finally, all plots indicate that the predictions are better in winter than summer. This could be due both to the fact that there is more winter data, and that the occurrence of strong wind gusts is higher in winter, thus easier to predict. The very rare occurrence of the wind gusts above 25 kt in the summer sample (122 times out of 7341 hours of data) means that probabilities above 0.6 are almost never forecasted, and thus difficult to verify on average. However, we conclude that the models deliver predictions that are in accordance with the 30% and 40% probability requirement of the TAF.



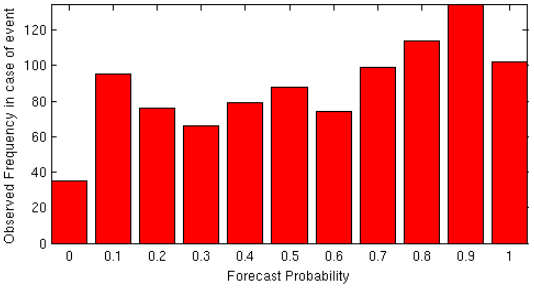
(a) Summer: reliability diagram



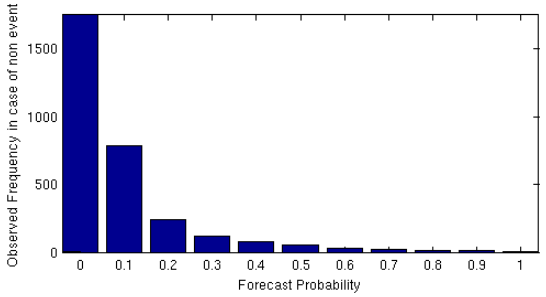
(b) Winter: reliability diagram



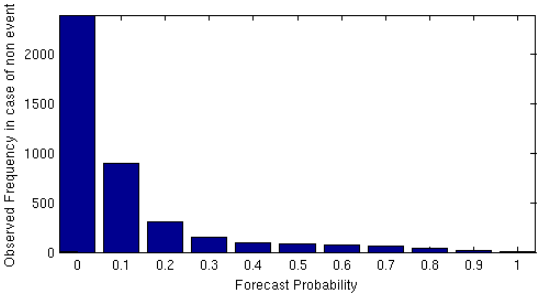
(c) Summer: distribution of events



(d) Winter: distribution of events

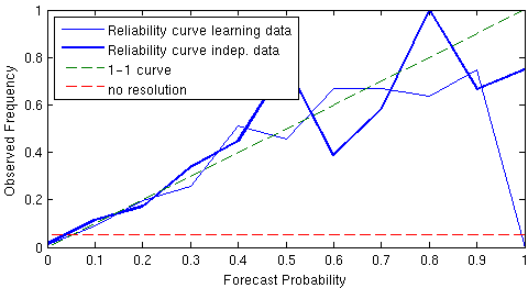


(e) Summer: distribution of non-events

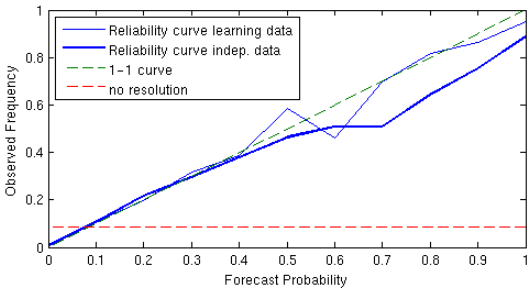


(f) Winter: distribution of non-events

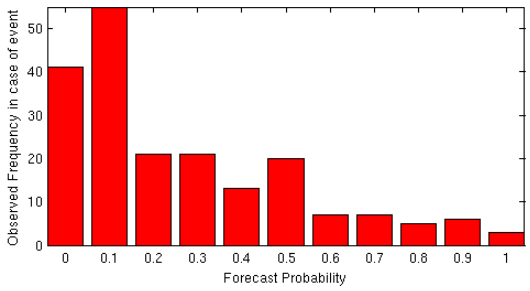
Figure 8: Wind models: Diagnostic plots for wind gusts models above 15 kt, in winter and summer.



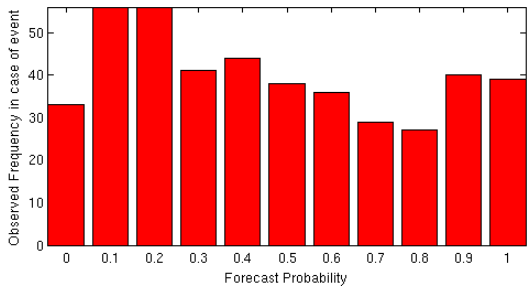
(a) Summer: reliability diagram



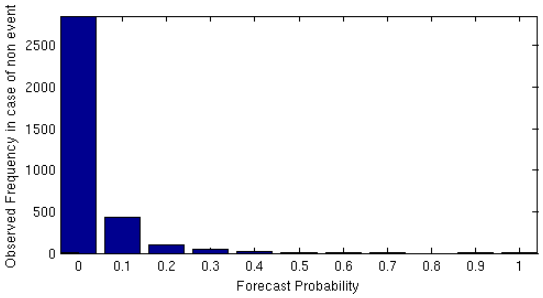
(b) Winter: reliability diagram



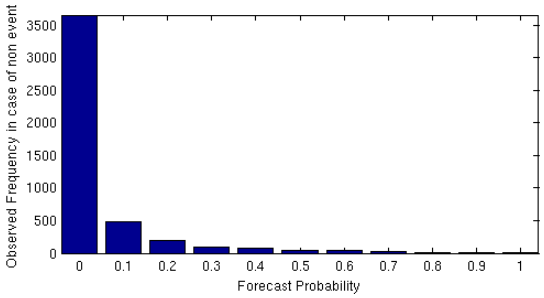
(c) Summer: distribution of events



(d) Winter: distribution of events

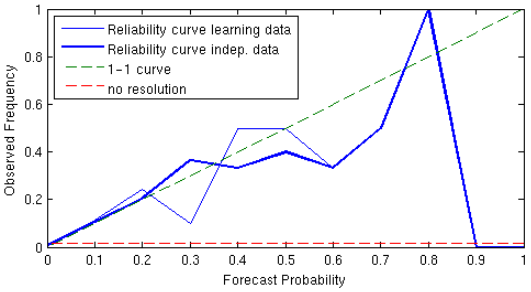


(e) Summer: distribution of non-events

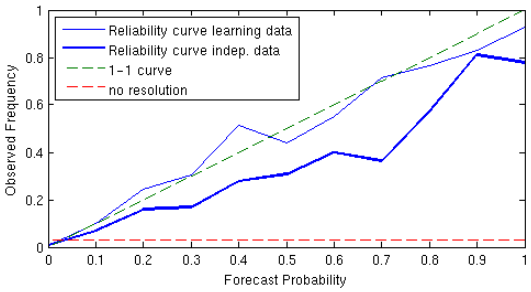


(f) Winter: distribution of non-events

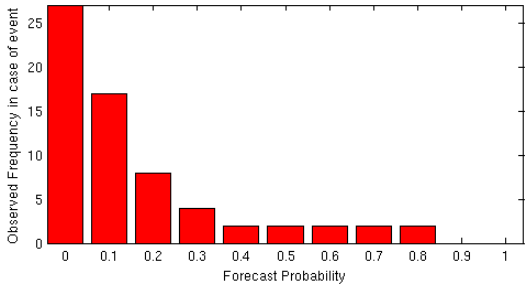
Figure 9: Wind models: Diagnostic plots for wind gusts models above 20 kt, in winter and summer.



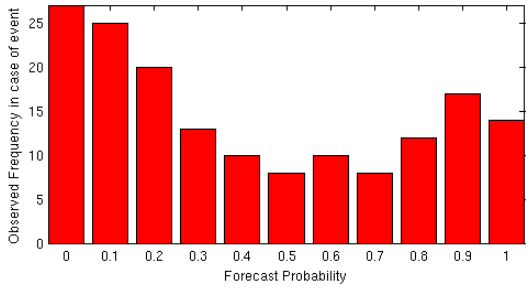
(a) Summer: reliability diagram



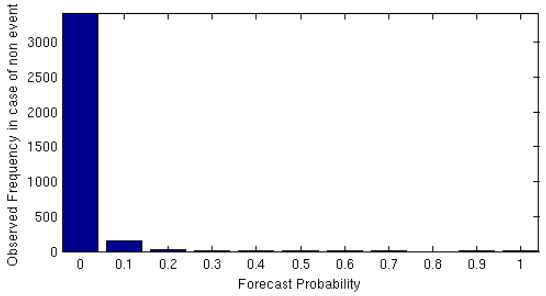
(b) Winter: reliability diagram



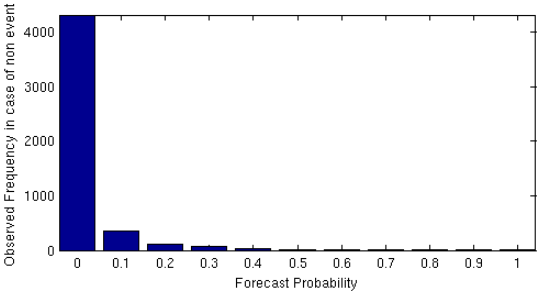
(c) Summer: distribution of events



(d) Winter: distribution of events



(e) Summer: distribution of non-events



(f) Winter: distribution of non-events

Figure 10: Wind models: Diagnostic plots for wind gusts models above 25 kt, in winter and summer.

4.1.6 Rescaling the forecasts

When very rare events are forecasted, it is difficult for a logistic regression model to estimate coefficients properly. As a result, forecasting an event with great certainty is almost impossible. Such models usually deliver very few probabilistic forecasts taking high values. In the wind models previously introduced, the predictions of wind gusts above 25 kt rarely exceeded probabilities of 0.3. This lack of "prediction-observation" pairs hinders the statistical verification of forecasts above 0.3, therefore the model is considered unreliable above the 0.3 threshold.

In order to take into account this unavoidable uncertainty, we decide to study the same plots after applying a transformation on the probabilistic forecasts produced. By rounding down the forecast values above 0.4, we ensure that the model delivers predictions within the $[0, 0.4]$ interval. Additionally, this is convenient to fit the format of TAF predictions, and corresponds to what a TAF forecaster would do if he was given probabilistic information in the $[0, 1]$ scale. We decide to only apply this scaling to the models for wind gusts above 25 kt, since the other models yield satisfactory probabilistic forecasts.

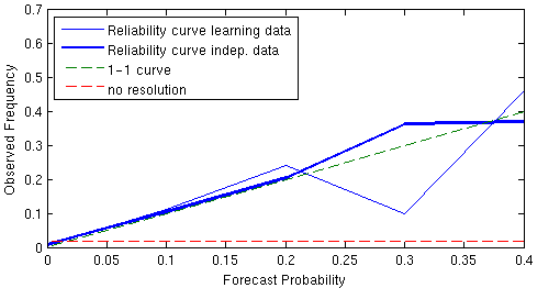
Figures 11 show the reliability plot and the refinement distributions of the rescaled forecasts, for both the winter and summer data. We see that up to probabilities of 0.4, it is possible to obtain reliable information about the occurence of wind gusts above 25 kt.

4.1.7 Model improvements through predictor transformation

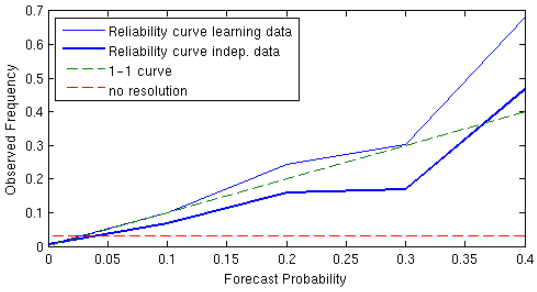
In this section, we briefly discuss the result of an experiment in which a transformed variable is used as a predictor, in order to indicate to which season an observation belongs. The predictor we use is not directly the month expressed in number, because for similar winter months such as January and December, the predictor would take opposite values (1 and 12), which is not coherent. Therefore, we transformed the month variable into a binary predictor indicating whether the season is winter ($x = 1$) or summer ($x = 0$). The aim of introducing a transformed predictor indicating seasons is to be able to use the whole sample for estimation, without estimating separate models based on winter and summer data. We want to find out if this new predictor will be selected as important, and whether predictions can be improved.

The results show that the season predictor is indeed selected by the algorithms, but not by all three models: it exclusively appears in the regression equation of the model for wind gusts above 15kt, and is only the 7th most important predictor. The other selected predictors are both the ones important in summer and winter. Therefore, it seems that separating the data is a better way to truly account for seasons, at least when it comes to the composition of the models.

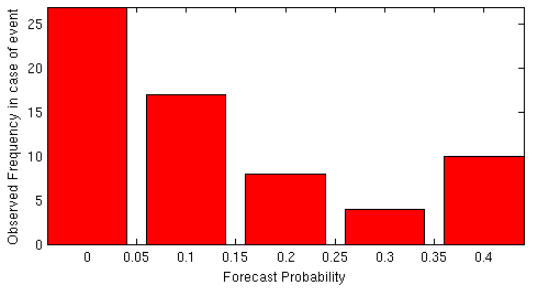
Looking at the diagnostic plots displayed in Figure 12, we see that the forecasts are nevertheless reliable, although the chosen predictors are form a less meteorologically coherent set. The loss of information concerning seasons does not seem to have a dramatic effect. However, we should not conclude that this information loss is unimportant, but rather that its negative impact on the results was



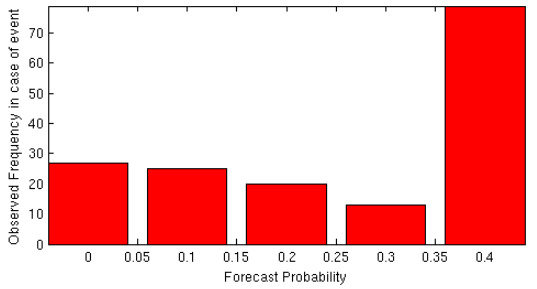
(a) Summer: reliability diagram



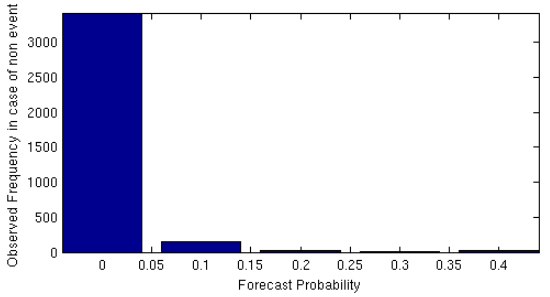
(b) Winter: reliability diagram



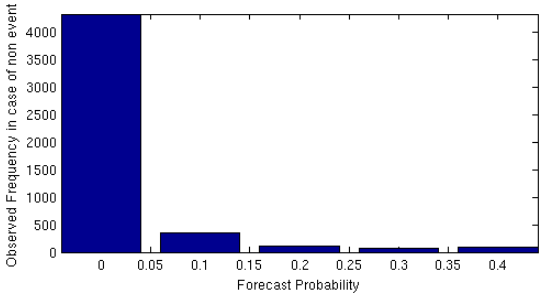
(c) Summer: distribution of events



(d) Winter: distribution of events

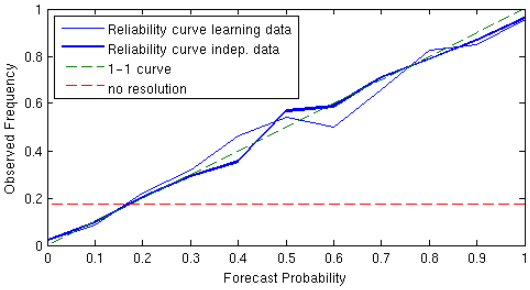


(e) Summer: distribution of non-events

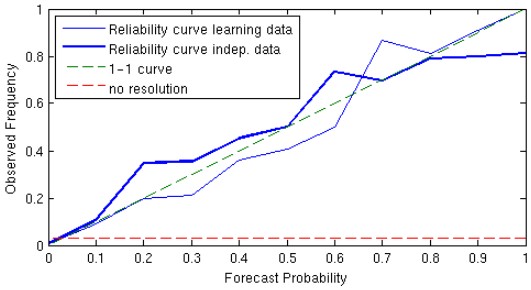


(f) Winter: distribution of non-events

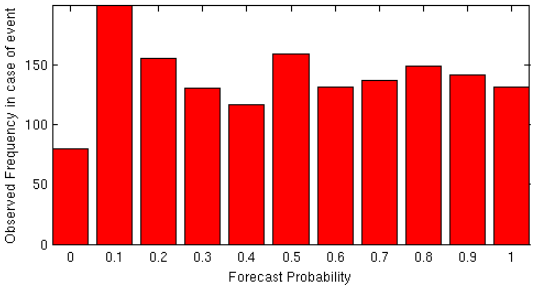
Figure 11: Wind models: Diagnostic plots for scaled forecasts of wind gusts models above 25 kt, in winter and summer.



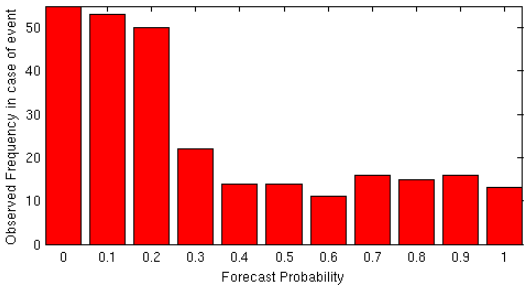
(a) 15 kt: reliability diagram



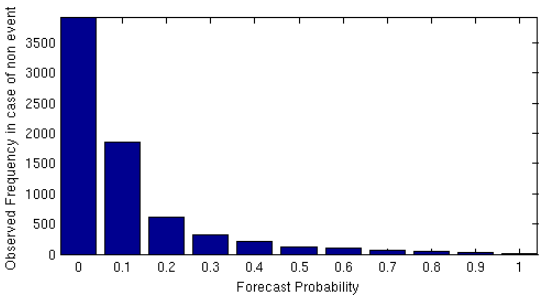
(b) 25 kt: reliability diagram



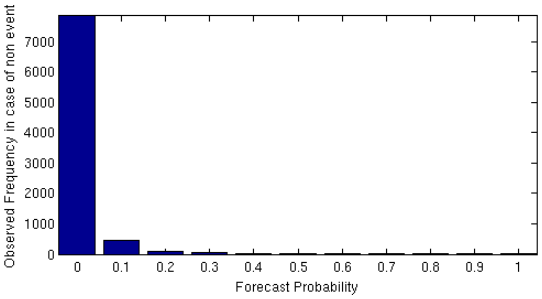
(c) 15 kt: distribution of events



(d) 25 kt: distribution of events



(e) 15 kt: distribution of non-events



(f) 25 kt: distribution of non-events

Figure 12: Wind models: Diagnostic plots for models with a season binary predictor.

compensated by the information gained by expanding the sample size.

Finally, if a very large amount of data is available (and the operational system permits it), the best method is still to estimate separate models for different times of the year, but if these conditions are not met, implementing a season predictor is a good compromise.

4.2 Visibility

In this section, we discuss the performance of the logistic regression models predicting limit values for the visibility. We give information on the data samples we use, and analyze both the coherence of the predictors and the performance of the models. We also explain how the methodology in this case differs from the one applied to wind gusts.

4.2.1 Approach

The models developed for the visibility possess two additional features compared to the ones studied in the previous section, which are meant to help dealing with the complexity of predicting a meteorological phenomenon such as fog.

Firstly, we estimate different models for different day times, assuming that if a single model is computed, the daily cycle is not implicitly accounted for by the predictors. This approach differs from the one used for wind gusts, where the time of the day played no role in the model. The wind gust model was based on data observed at all times of the day, in order to predict wind gusts at any time in the same way. On the contrary, the methods we implement enable us to subselect specific time periods in the data, and to estimate daytime-specific models. In the following analysis, we look at a time period of length 4 hours in the morning (between 6h and 9h local time), since it is during morning that fog is most likely to be observed. Additionally, this corresponds to the critical opening time of the airport, at which forecasts are especially important. Although we can easily configure the MATLAB script to estimate models for varying time periods, we choose to focus the study on this model, since having as many cases of fogs as possible in the data facilitates the estimation process.

The second difference is the time lag between the forecasted predictors and the predictand. While a wind gust model delivers a forecast valid at time t based on forecasts of predictors which are also valid at time t, we aim to build a regression model which take into account the temporal evolution of the predictors in its equation. We base this approach on the assumption that fog is a phenomenon which develops over a whole night, and therefore want to examine the values of the predictors at times preceding the observation of the visibility. Therefore, the predictors used for visibility are a range of forecasts that are valid at varying times within the last 12 hours before the actual time at which the visibility is predicted. For example, in order to predict the visibility at 6h, we use forecasted values of the predictors at 6h, but also from 18h to 5h, a concept illustrated in Figure 13. In this sense, the learning data is a collection of short stories rather than instantaneous moments.

4.2.2 The data

Since the visibility is not a valid DWH parameter, we have to retrieve a sample of visibility observations through another method. This is why the predictand data is in fact extracted from a series of METAR observations. The predictor data containing relevant COSMO-2 forecasts, on the other hand, is again

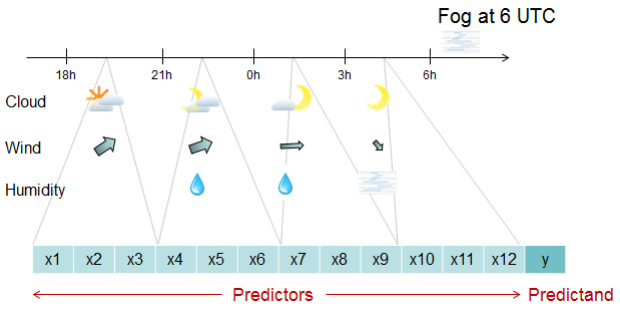


Figure 13: Illustration of the temporal evolution of predictors in one regression model

generated by Fieldextra. Since fog is a meteorological phenomenon occurring mostly in winter, the dates which are most interesting to us are the September-April months. We choose to gather data from this season, for as many years as possible. However, since the forecast records for our chosen predictors are not available earlier than 2009 due to changes in the COSMO model, our data consists of the "winter" months between September 2009 to April 2013.

The data is separated day by day between learning data and independent data, as previously done with wind gusts.

4.2.3 The regression models studied

We use only the data from one daily model run (12 UTC run) to estimate a model, since post-processing is always applied on the output of a single model integration at a time. Therefore, we thought that a database of forecasts from mixed model runs would yield a model with a remaining cycle of error, which furthermore would not be applicable to real operational post-processing. In practice, a model should be estimated for each run time (and operated only on the output of this precise run time), but we only study in this work the situation of the 12 UTC run time.

The models we further discuss are built based on observations of the visibility at 6,7,8 and 9h every day and a matching database of predictors. Each predictor is a meteorological parameter forecasted for a certain hour between -0h and -12h before the time the predictand was observed, according to the approach previously explained. The models we estimate are based on all observations between 6h and 9h, allowing for a certain "fuzziness" in time, and thus gathering more data. This is a compromise between estimating a different model for each day time, and estimating a too simple model which is valid at all times.

We did investigate the possibility of estimating a model for each specific day time, and finally decided to estimate a "morning" model with timing flexibility, assuming the conditions between 6h and 9h are rather similar. This is due to the inconvenient lack of data in the first approach, which becomes problematic to estimate models with very low thresholds. We do not detail here the comparison between both options.

Finally, similarly to wind gusts, we estimate and compare models for each of our thresholds of interest: 3000m, 1500m, 600m.

4.2.4 Predictor selection

The initial pool of predictors contains 169 predictors. This pool consists of 13 different meteorological parameters, each of which takes 13 different values (one per hour), from -0h until -12h before the target time. Table 3 shows the different parameters.

	Abbreviation	Parameter
1	CLCT_13GP	Total Cloud Cover in %
2	CLCH_GP	High Cloud Cover in %
3	CLCM_13GP	Middle Cloud Cover in %
4	CLCL_13GP	Low Cloud Cover in %
5	ATHB_S	Long-wave radiation balance at the surface
6	VMAX_10M	Maximum wind speed at 10m
7	U_10M_GEO	Wind speed at 10m in the W-E direction
8	V_10M_GEO	Wind speed at 10m in the S-N direction
9	TOT_PREC_05GP	Total precipitations
10	SNOW_%_05GP	Percentage of snow in precipitations
11	T_2M	Temperature at 2m
12	TD_2M	Dew point temperature at 2m
13	RH_2M	Relative humidity at 2m

Table 3: Initial pool of parameters given to the predictor selection algorithms

We use the stepwise selection algorithm based on the likelihood ratio test to select a meaningful subset of predictors for each threshold. The complete selection can be found in Table 4. However, Figure 14 displays a simplified summary of the general tendency visible in all models regarding the choice of predictors and coefficients sign.

We observe that the most relevant parameter is usually the maximum wind speed, VMAX_10M. The percentage of snow in precipitation, the cloud cover at various levels, the relative humidity and sometimes the dew point temperature are also important predictors. On the contrary, the long-wave radiation balance, the total precipitations and the average wind speed in specific directions are irrelevant.

Concerning the times, we notice that the chosen time of the day of each forecasted predictor varies greatly. For parameters such as snow or cloud cover, it is their value at time 0, at which the predictand forecast is valid, which matters the most in the equation. This makes sense, as for example the percentage of snow in precipitations has an immediate influence on the visibility. However, the value of wind speed or relative humidity plays a role at an earlier time in the process, around 10 to 12 hours before fog is actually observed. We suppose however that another reason why these early times were selected is also because they are more precise, since the earlier we look, the smaller the lead time of the forecasts are.

In general, the chosen times for the predictors are either simultaneous to the forecasted event (-0h or -1h), or well ahead of it (-11h or -12h). Some predictors are important only in one of these timings, for

	3000m model		1500m model		600m model
1	VMAX-11h	1	VMAX-11h	1	CLCL0h
2	SNOW-0h	2	CLCT-1h	2	VMAX-11h
3	CLCT-0h	3	TD-12h	3	TD-12h
4	VMAX-2h	4	RH-12h	4	RH-12h
5	RH-11h	5	CLCL-0h	5	CLCM-1h
6	TD-5h	6	SNOW-0h	6	CLCT-12h
7	CLCM-11h	7	T-8h	7	VMAX-7h

Table 4: Ordering of predictors resulting from the stepwise selection procedure

others such as wind speed and cloud cover, it appears important to know both values. Fortunately, we observe a constancy among the three different models.

We chose to keep a maximum of 7 predictors. We fixed a limit to the number of predictors in the equation, below the absurdly high number of predictors actually selected by the stepwise procedure (more than 20). This is because many predictors are very mutually correlated, and adding too many of them decreases only very little the errors while complicating vainly the equation. The number 7 has been chosen as an approximately right stopping point.

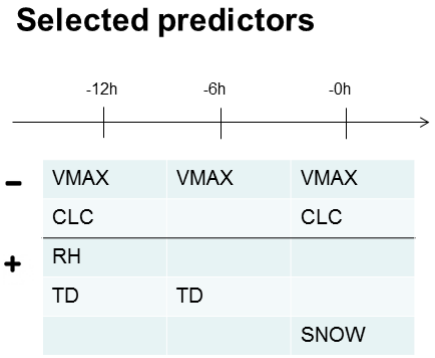


Figure 14: General tendency regarding the choice of predictors based on all visibility models studied. The approximate chosen times for each selected meteorological parameter are shown on the top of the figure. The left side indicates the sign of the associated coefficient.

4.2.5 Coefficients estimation

Figure 15 shows the output of the estimation procedure. All parameters are statistically significant, and a closer look at the value of the coefficients reveals that the model is coherent with our expectations.

The sign of the coefficients associated to wind and cloud cover is almost always negative, meaning that the higher these values, the less likely is the visibility to fall below the threshold. Since strong wind and a cloudy sky prevent the formation of fog, the values of these coefficients seem reasonable.

On the other hand, relative humidity, temperature and snowy rain have positive coefficients. Therefore, the higher the value of these parameters, the more likely is the visibility to reach low thresholds. This

```
Generalized Linear regression model:
    Y ~ [Linear formula with 8 terms in 7 predictors]
    Distribution = Binomial

Estimated Coefficients:
                   Estimate     SE           tStat      pValue
    (Intercept)      -2.2896       0.6126    -3.7375    0.00018586
    VMAX_11h        -0.34631     0.044728    -7.7426    9.7382e-15
    SNOW_0h         0.026549    0.0033574     7.9074    2.6272e-15
    CLCT_0h        -0.012041    0.0019251    -6.2549    3.9782e-10
    VMAX_2h         -0.33334     0.050415     -6.612    3.7921e-11
    RH_11h            0.0417    0.0076263     5.4679    4.5544e-08
    T_ROSEE_5h       0.04905     0.011837     4.1438    3.4159e-05
    CLCM_11h       0.0066732    0.0018663     3.5757    0.00034924


1928 observations, 1920 error degrees of freedom
Dispersion: 1
Chi^2-statistic vs. constant model: 359, p-value = 1.48e-73
```

(a) 3000m model

```
Generalized Linear regression model:
    Y ~ [Linear formula with 8 terms in 7 predictors]
    Distribution = Binomial

Estimated Coefficients:
                   Estimate     SE           tStat      pValue
    (Intercept)      -7.1955       1.2751    -5.6432    1.6694e-08
    VMAX_11h        -0.36544     0.049847    -7.3313    2.2797e-13
    CLCT_1h         -0.01015    0.0024535    -4.1371    3.5175e-05
    T_ROSEE_12h     -0.23018     0.079816     -2.884     0.0039272
    RH_12h          0.081096     0.014442     5.6152    1.9636e-08
    CLCL_0h        -0.016193    0.0031205    -5.1895     2.109e-07
    SNOW_0h         0.023278    0.0042696     5.4519    4.9821e-08
    T_8h             0.32558     0.084282      3.863    0.00011201


1928 observations, 1920 error degrees of freedom
Dispersion: 1
Chi^2-statistic vs. constant model: 257, p-value = 8.15e-52
```

(b) 1500m model

```
Generalized Linear regression model:
    Y ~ [Linear formula with 8 terms in 7 predictors]
    Distribution = Binomial

Estimated Coefficients:
                   Estimate     SE           tStat      pValue
    (Intercept)      -3.2825      0.89823    -3.6545    0.00025771
    CLCL_0h        -0.025337    0.0041673    -6.0799    1.2024e-09
    VMAX_11h        -0.33337     0.083254    -4.0043    6.2203e-05
    T_ROSEE_12h      0.10059     0.018268     5.5067    3.6562e-08
    RH_12h          0.038762     0.010754     3.6043    0.00031303
    CLCM_1h        -0.013285    0.0029773    -4.4622     8.111e-06
    CLCT_12h        0.011493    0.0031199     3.6839     0.0002297
    VMAX_7h         -0.40688     0.092517     -4.398    1.0928e-05


1928 observations, 1920 error degrees of freedom
Dispersion: 1
Chi^2-statistic vs. constant model: 265, p-value = 1.75e-53
```

(c) 600m model

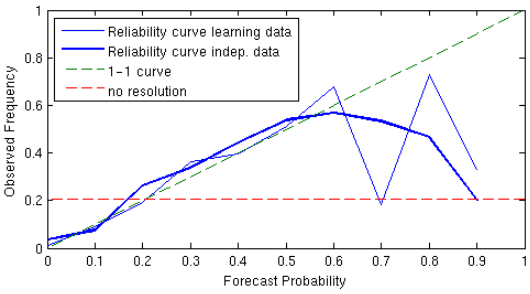
Figure 15: Visibility model: Outputs of estimation results

is again coherent with the fact that snowy rain decreases the visibility, while high relative humidity intensifies fog.

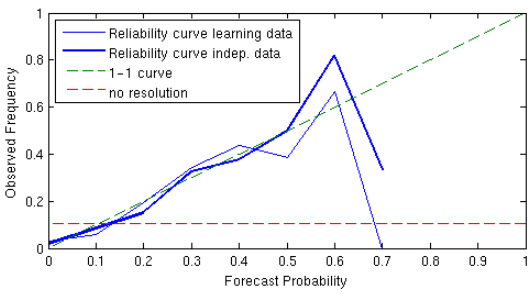
4.2.6 Forecast verification

The results of the forecast verification with independent data are shown in Figures 16 and 17.

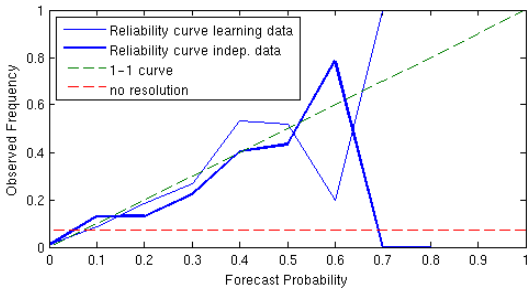
We see from the reliability diagram in Figure 16 that the global model can predict well up to probabilities of 0.5, since the green diagonal line is well followed up to this limit. There are very few occurrences of probabilistic forecasts in each category above 0.6, thus at this point the verifications are unreliable, and the blue curve follows a random pattern. We conclude that we cannot expect the model to yield very large probabilities of high loss of visibility with certainty, however probabilities between 0 and 0.5 can be accurately predicted.



(a) 3000m



(b) 1500m



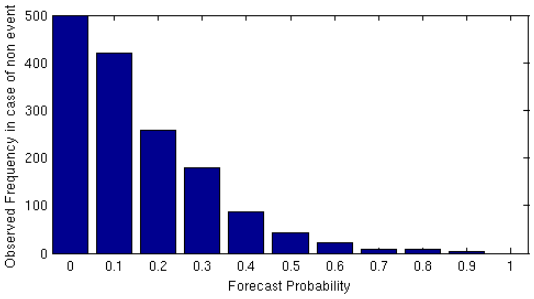
(c) 600m

Figure 16: Visibility models: Reliability diagrams for models corresponding to different visibility levels.

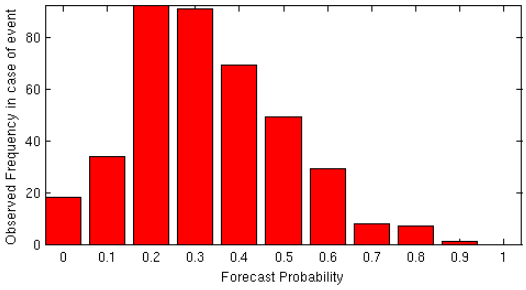
By looking at the refinement distribution in blue in Figure 17, we see that when the threshold was not exceeded, very low probabilities were most of the time delivered by the models. On the other hand, the red distribution of events show that the forecasted probabilities are not very high even when the threshold was exceeded. Positively, the average forecast in case of event is higher than the climatological average, and the average forecast in case of non-events is lower, however the certainty of the model is still quite low.

This shows that predicting visibility is a lot more complex than wind gusts, and we cannot expect to build a model which will deliver forecasts with high certainty that an event does occur, especially when the threshold is set at the limit of the possible range of values.

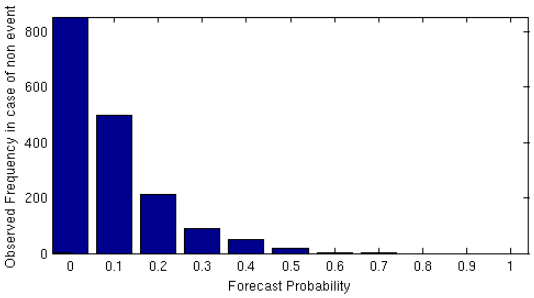
In the next sections, we will explore some possible model improvements.



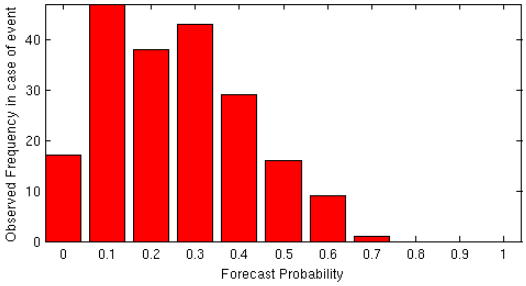
(a) Distribution of non-events: 3000m



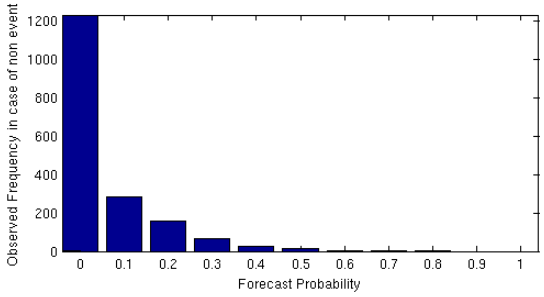
(b) Distribution of events: 3000m



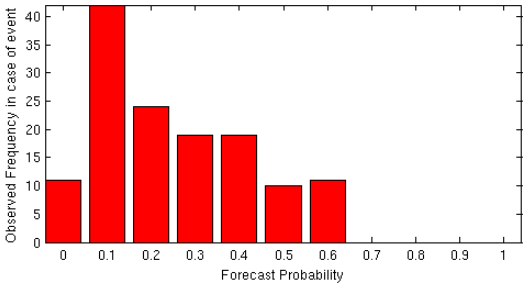
(c) Distribution of non-events: 1500m



(d) Distribution of events: 1500m



(e) Distribution of non-events: 600m



(f) Distribution of events: 600m

Figure 17: Visibility models: Distribution of non-events and events in models corresponding to different visibility levels.

4.2.7 Model improvement through data sampling strategies

We now discuss again the model's performance, in the light of improvements of the model through data sampling strategies. Data sampling consists of gathering data for the model's estimation, and this can be conducted in different manners. The usual strategy, which was used so far, is known as "random sampling". It collects a random sample of data, or all the available data if there is little of it, without distinction. The basic model introduced previously was based on this approach, since we included all available data in our sample. However, when dealing with forecasting rare events, some data sampling strategies are more effective than the simple random method.

Two of these data sampling strategies are *exogenous stratified* sampling and *endogenous stratified* sampling. Exogenous sampling consists of selecting a biased data sample according to the values taken by the exogenous variables, that is to say the predictors. The data is selected within categories defined by X. On the other hand, endogenous stratified sampling amounts to selecting a sample based on categories of the predictand Y.

These data sampling strategies are used to curtail the problem of the rarity of events (the cases when the visibility falls below the threshold). We want to see the effect of these two strategies on the performance of the model, therefore we first state the details of the sampling method, and then discuss the forecasts verification plots of the newly estimated models.

We first state the results of the endogenous stratified sampling. Since the endogenous variable, in other words the predictand, is in this case binary, we have two distinct categories of values (0 or 1). The aim of this sampling strategy is to improve the proportion of events ($Y = 1$) in the data sample, in other words *oversampling*. In order to gather a biased sample, we can either lessen the number of cases where $Y = 0$ by subselecting within this category, or increase the number of cases where $Y = 1$, by duplicating data within this category. Since the first approach implies reducing the size of the dataset, which is already rather small, we opt for the second approach. This allows to give more weight to cases of events in the estimation, without losing information or jeopardizing the stability of the coefficients.

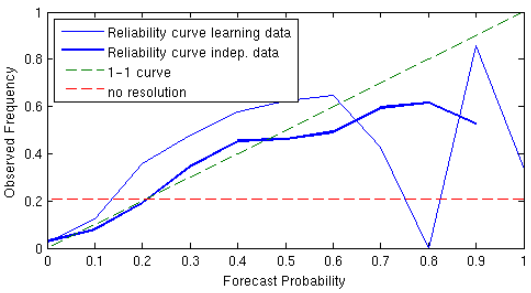
However, we have to be careful when using oversampling, as a logistic model estimated on a biased sample has to be adapted to be used with new data, which has a normal proportion of events. The estimates of the coefficients thus need to be corrected before the model can be used for prediction. An efficient statistical method to correct the estimates is called *prior correction*. It leaves unchanged all coefficients apart from the constant term estimate $\hat{\beta}_0$, which is corrected as follows

$$\hat{\beta}_0 - \ln\big((\frac{1-\tau}{\tau})(\frac{y}{1-\bar{y}})\big)$$

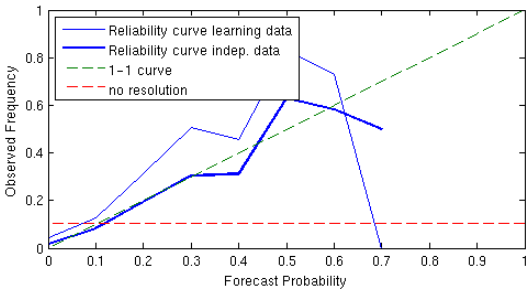
where τ is the climatological average occurrence of events, and $\bar{y}$ is the proportion of events in the biased data sample.

Figures 18 and 19 detail the results. We conclude that the model developed with this method is also good, but not a significant improvement of the existing model. The model for the 3000m threshold

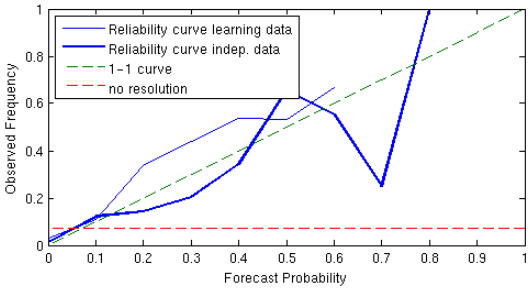
shows a well-calibrated reliability diagram and slightly higher probabilities in case of events, which is positive. However for lower thresholds, we do not obtain improved distributions of probabilities, and the reliability diagrams depart too much from the diagonal line. Even with this method, forecasts above 0.5 are unreliable.



(a) 3000m



(b) 1500m



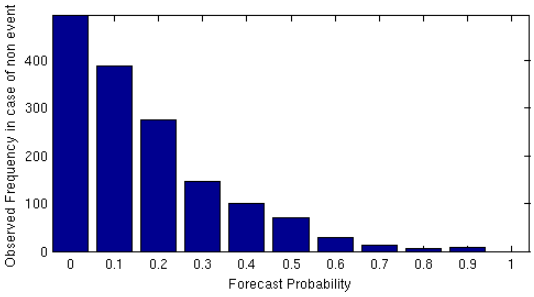
(c) 600m

Figure 18: Visibility models: Reliability diagrams for models estimated with oversampled data.

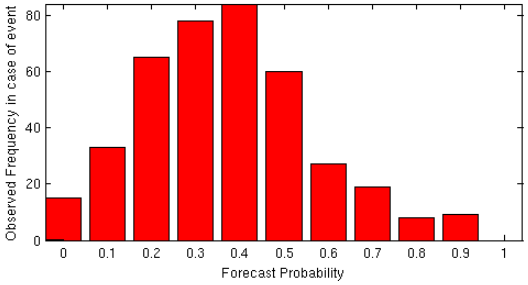
We now discuss exogenous stratified sampling. The strategy we use consists of selecting a sample of days during which the visibility is especially likely to be low given the values of the predictors, and estimating a new model on this subset. This first screening is done by looking at the the probabilities of visibility below 3000m delivered by our basic model, which is a function of the predictors values X, and keeping only those reaching a certain level. Unlike previously, the selection is not done on the actual values of Y. Similarly, the purpose here is to get rid of a large amount of data which most likely has a probability 0 to reach low values, and to keep only relevant data. This relevant data is used to estimate models for lower visibility levels (1500m, 600m).

We note that this two-step approach is analogous to the method employed by a human forecaster, who after assessing whether there is a risk of bad visibility, looks into more detail into the risky cases. This biased sample of “risky cases’ ’ constitutes our exogenous stratified sample, which serves to estimate models for visibility below 1500 and 600m.

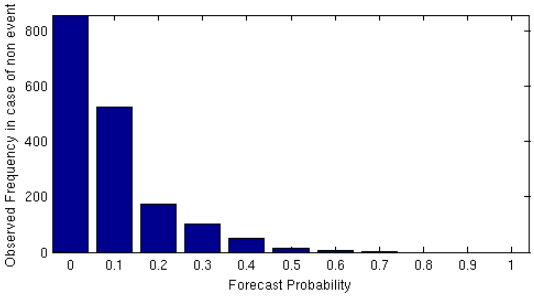
An analysis of this method, which can be found in intermediary reports, shows that it brings nothing more than the previously analyzed oversampling strategy. We thus drop this approach.



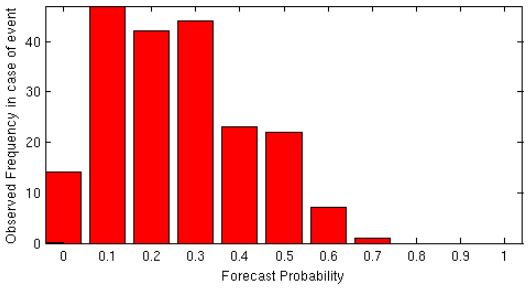
(a) Distribution of non-events: 3000m



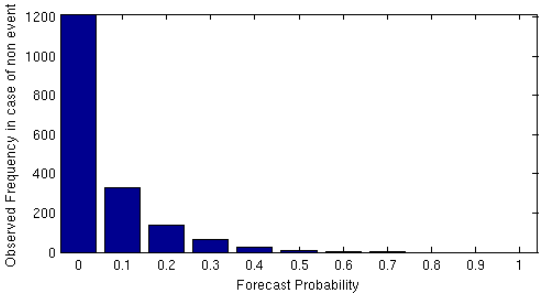
(b) Distribution of events: 3000m



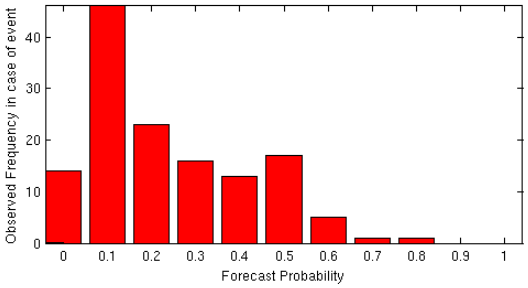
(c) Distribution of non-events: 1500m



(d) Distribution of events: 1500m



(e) Distribution of non-events: 600m



(f) Distribution of events: 600m

Figure 19: Visibility models: Distribution of non-events and events in models estimated with oversampled data.

4.3 Thunderstorms

The last phenomenon we are interested in is thunderstorms. In this section, we explain the characteristics and the performance of the thunderstorm regression models we developed.

4.3.1 The approach

Since thunderstorms are meteorological phenomenon which develop over a certain time period before bursting, we apply the same methodology as for visibility models, described previously in section 4.2.1. We build a predictor table consisting of a collection of predictor values over varying times, so that the models are based on the evolution in time of the predicting parameters. The values which we look are forecasts of the predictors, for each hour between the actual time of the thunderstorm and 6 hours before.

Secondly, we also choose to estimate daytime specific models. We do not estimate one model per day time, but form groups corresponding to wider time periods of length 4 hours, in order to increase data sample size.

4.3.2 The data

The predictor data comes from records of past COSMO-2 forecasts, and the predictand data is extracted from METAR observations, but not directly. Contrarily to visibility or wind gusts, there is no continuous parameter characterizing thunderstorms. A thunderstorm is in itself an event to predict. Therefore, in order to obtain a binary variable defining the occurrence of a thunderstorm, we build it from observations of two other parameters in the METAR. The first one is the indication of cumulonimbus specified by the abbreviation CB, with a cloud cover equivalent to FEW or more. The second one is the descriptor TS signifying thunderstorm. We define the observation of a thunderstorm - and accordingly set the predictand variable to the value 1 - when both TS and CB appear in the METAR.

Since thunderstorms occur most of the time in summer, we gather data within the May-September months, starting in 2009 and ending in 2013. The data is separated day by day between learning and testing samples.

To sum up, our data consists of daily observations of the thunderstorm predictand and a matching database of predictors. Each predictor is a meteorological parameter dynamically forecasted for a certain hour between -0h and -5h before the time the predictand was observed, similarly to the fog learning approach but for a shorter time period.

4.3.3 The regression models studied

Thunderstorms occur most often during afternoons or evenings in summer. Therefore, the focus of the study is logistic regression models calibrated specifically for these day times. The first time interval corresponds to the late afternoon period (from 16h to 19h), and the second one to the evening (from 20h to 23h). We had initially decided to estimate only one model in order to have more data at disposal. However, since we aim to predict thunderstorms at very different times, from afternoon at 16h until evening at 23h, and since the values of the predictors change significantly within this time period, it appeared to be a too extreme simplification.

Finally, all models in this study are based on predictor forecasts from the 03 UTC run. In practice, different models for each run time should be estimated, but we only study the 03 UTC example.

4.3.4 Predictor selection

The initial pool of predictors contains 36 predictors. It consists of 6 different meteorological parameters, each of which takes 6 different values (one per hour), from -0h until -5h before the target time. Table 5 shows the different parameters.

	Abbreviation	Parameter
1	GLOB	Gloabl solar radiation at surface in the last hour
2	CAPE_MU	Convective available potential energy of most unstable parcel
3	VMAX_10M	Maximum wind speed at 10m
4	T_2M	Temperature at 2m
5	RH_2M	Relative humidity at 2m
6	PS	Surface pressure

Table 5: Initial pool of parameters given to the predictor selection algorithms

The predictors are selected with the stepwise algorithm, but the result of the selection is compared with the forward method, in order to examine how dependent on the algorithm the chosen subset is. Table 6 shows the subset of predictors selected by the stepwise algorithm, for three different models: the models corresponding to afternoon and evening day times, and the daily model based on all observations.

The main difference between these subsets and the ones selected with the forward selection procedure is that the stepwise procedure (which has an automatic stopping criterion) chooses to stop adding parameters much earlier. Even though we allow the model to have up to 7 parameters, all models have between 3 and 6 parameters only. The first selected predictors also slightly differ with the forward procedure : some of the parameters selected by the forward method are removed in a backwards step when the stepwise algorithm is used.

Concerning the choice of predicting parameters, we conclude from the selection of table 6 that two parameters are important well ahead of the time at which the predictand is observed: T_2M and RH_2M. Hence, the heat and relative humidity during the middle of the day determines the chance of thunder-

	Daily model		Afternoon model		Evening model
1	PS-1h	1	CAPE_MU-5h	1	T_2M-5h
2	T_2M-5h	2	PS-1h	2	VMAX_10M-0h
3	VMAX_10M-0h	3	VMAX_10M-0h	3	RH_2M-5h
4	CAPE_MU-1h			4	PS-2h
5	RH_2M-5h				
6	T_2M-4h				

Table 6: Ordering of predictors resulting from the stepwise selection procedure

storm in the late afternoon or evening. On the other hand, the values of maximum wind speed and surface pressure given by the parameters VMAX and PS are important at the time of the thunderstorm or shortly before it. The parameter CAPE_MU seems to have importance at all times.

The main observation concerning the grouping of the afternoon and evening observations into a daily model is that more predictors are selected in this case (6 instead of 3 or 4). When less data is available, fewer parameters are usually recognized to be significant, so this is not surprising. This confirms the fact that very large dataset must be gathered in order to optimally estimate such logistic regression models.

4.3.5 Coefficient estimation

Globally, the selected predictors and their coefficients are coherent with our knowledge of thunderstorms. The predictors T_2M, R_2M, CAPE_MU and VMAX_10M all have positive coefficients, while PS is the only predictor with a negative coefficient.

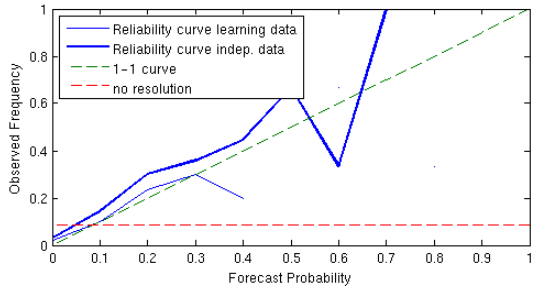
The interpretation is that the warmer and the more humid the weather is during the middle of the day (considering the -5h), the better thunderstorms develop and burst at the end of the day. Strong wind is also positively correlated with thunderstorms, without any time lag however. The instability of the atmosphere indicated by the parameter CAPE_MU is another strong sign of a storm arriving. Finally, the air pressure (with time lag -1h) is on the contrary negatively correlated with thunderstorms, meaning that a low pressure at some point in time indicates that a storm is more likely to occur one or two hours later.

4.3.6 Forecast verification

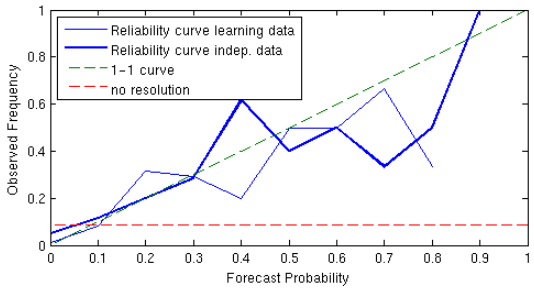
What appears in the diagnostic plots of these models (shown in Figure 20) is that thunderstorms are harder to predict than the previously studied phenomenons. The distribution of events is barely different from that of non-events, and almost no probabilities above 0.2 are delivered by the models.

Thus, the majority of cases of thunderstorms are completely missed. The predictions seem to be rather reliable up to 0.4, according to the reliability diagram, but in reality the model very rarely forecasts probabilities of 0.3 or above, which is what is needed in a TAF. The parameters selected in the model are yet consistent with the meteorological reality, therefore the result is a bit disappointing. We conclude

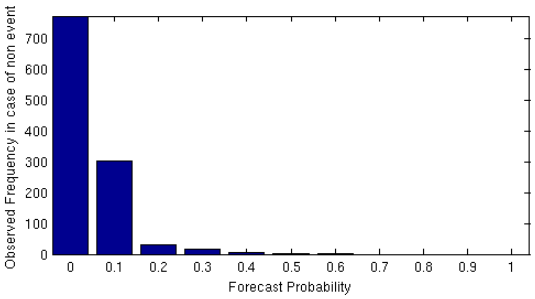
that predicting the occurrences of thunderstorms is very tricky, and that we could try some model improvements.



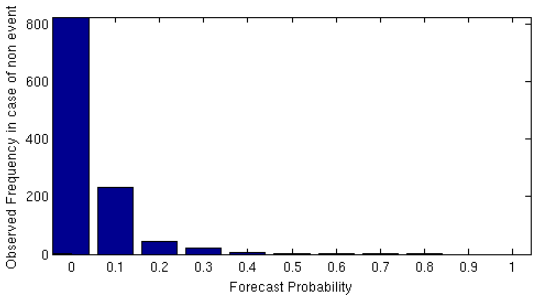
(a) Reliability diagram: afternoon



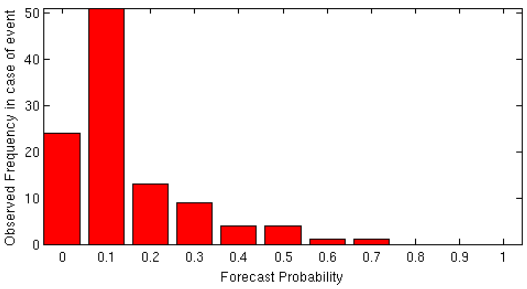
(b) Reliability diagram: evening



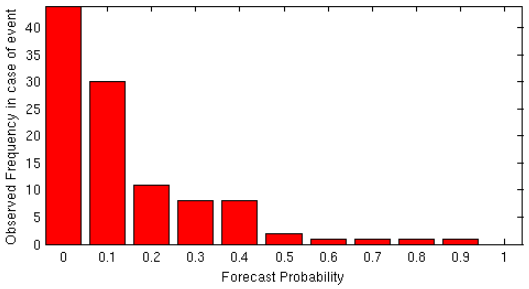
(c) Distribution of non-events: afternoon



(d) Distribution of non-events: evening



(e) Distribution of events: afternoon



(f) Distribution of events: evening

Figure 20: Thunderstorm models: Diagnostic plots for afternoon and evening models.

4.3.7 Model improvements through variable transformation strategies

In order to obtain better result, we test different predictor transformation strategies. We keep the same set of initial meteorological parameters, but carry out two specific transformations. The first one consists of taking the average of each parameter over a certain time period. These averaged parameters are then added to the set of existing predictors, so that we have for each parameter both values corresponding to a specific point in time, and an average value. This could be helpful for values of parameters which tend to vary a lot from an hour to the next, such as the CAPE_MU.

The second transformation only concerns the surface pressure parameter. Signs that a thunderstorm is coming are not only given by the value of pressure, but also by the pressure gradient from one hour to

the next. Indeed, the pressure before a storm is usually low, but suddenly rises after the thunderstorm bursts. Therefore, we add a new predictor DELTA_PS measuring the difference of pressure between one hour after and one hour before the predictand is observed.

This set of original and additional transformed predictors is given to the stepwise algorithm to estimate new models. We then compare the new selected subset of predictors to see which transformations were relevant. The result is that the models change very little and practically the same predictors are selected. The model corresponding to afternoon thunderstorms stays the same, and the one for evening thunderstorms has only one additional parameter, which is DELTA_PS. The coefficient associated with it is positive, which is coherent with the meteorological reality.

However, none of the averaged over time predictors was considered significant. The model continues to use unsmoothed values of predictors corresponding to a specific past hour. This does not mean that no useful information is contained in averaged predictors, but simply that “instantaneous” predictors are more adequate. If no predictors from past hours are available, the averaged over time predictors are a replacement, but they do not improve the existing model.

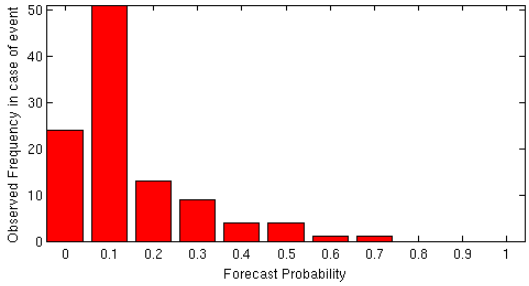
Finally, when we examine the reliability plots given by the model with the additional DELTA_PS parameter, we conclude that this new model has the same flaws than the previous one. Therefore, variable transformation strategies do not appear to be efficient ways to make significant model improvements.

4.3.8 Model improvements through data sampling strategies

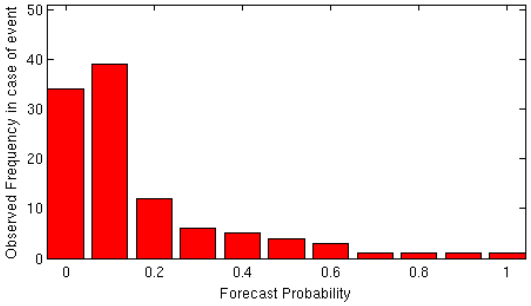
We aim to gather more cases of thunderstorms in our dataset, in order to improve the regression models. Currently, the data contains approximately 10 percent of “ones”. We resort to duplication of cases of thunderstorms to increase our proportion of events, a method of oversampling which has already been explained in section 4.2.7. We try to copy once the events of interest, then extend the experience by making several other copies, so that the cases of thunderstorms appear not once in the data, but 2,3, or even 10 times. Thus, in the process of estimating coefficients which best fit the data, more weight is given to the observations of thunderstorms.

We hope that this data sampling strategy will allow us to obtain more reliable forecasts, in which events of thunderstorms are predicted with higher probabilities.

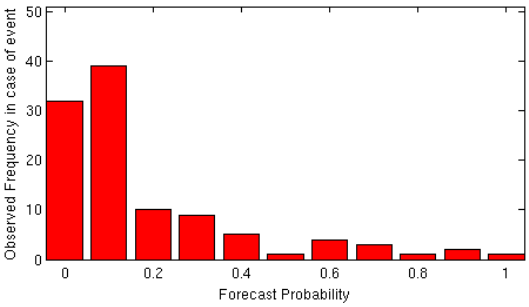
The results show that with this oversampling strategy, more predictors are selected by the algorithm, probably because this creates (artificially) more data. However, the probabilities delivered in case of events are still centered around 0.1, which is far too low. The model uses the whole scale of possible forecasts (between 0 and 1), but still very few forecasts are made above 0.3 certainty. Figure 21 illustrates this problem by showing the distribution of events for several models in which the cases of events were multiplied a number of times, with the afternoon data (result rather similar with the evening data). We notice that although there is a slight amelioration as more data is multiplied, most probabilistic forecasts are very low when a thunderstorm indeed occurs.



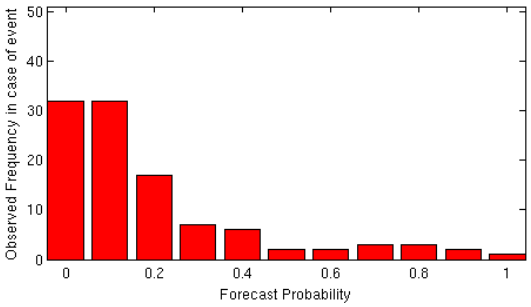
(a) Events non duplicated



(b) Events duplicated 1x



(c) Events duplicated 5x



(d) Events duplicated 10x

Figure 21: Thunderstorm models: Distribution of events for afternoon thunderstorm models estimated with oversampled data.

5 Options for implementation

Having asserted the possibility of forecasting some of the probabilistic elements of TAF with the method developed, we now discuss options to implement the project in the operational routine of Meteoswiss. The logistic regression equations that have been estimated need to be used daily to compute predictions which can then be used as indicators by forecasters writing TAF. Each integration of the COSMO-2 models needs to be post-processed to deliver a new set of probabilistic forecasts.

5.1 COSMO-MOS

COSMO-MOS is a software developed at Meteoswiss which is has been designed to post-process the output of the COSMO models. Currently, it computes daily corrections for specific parameters of the COSMO-2 model (mostly temperature-related parameters), using a Kalman filter algorithm. However, another branch of this software is intended to compute several kinds of regression equations on COSMO parameters, although it is not yet implemented in the operational routine. COSMO-MOS is managed with a configuration file, which defines firstly the parameters to be predicted, the parameters to use as predictors, the modalities of the learning process, and which drives the access to both observations and forecasts databases. Furthermore, a location list defines the places (e.g. airports, communication lanes, cities) for which the statistical models have to be computed. Therefore, a possibility to implement the project consists of writing a new configuration file which will let COSMO-MOS do the needed computations automatically.

However, several features of COSMO-MOS make it unsuitable to this task. COSMO-MOS is focused on the treatment of continuous weather parameters, e.g. temperatures or wind, and is designed to correct bias of the numerical forecasting model on a continuous basis. Instead of learning first on a sufficiently large number of days, and using the resulting model for a lasting time period, COSMO-MOS executes a daily learning cycle. This means that each day, it learns and operates new models. More precisely, the learning cycle is done on a predefined set of predictors, for a large set of locations, for each lead time of the forecast. Thus, if M is the number of locations, N the number of predictors per station, and L the number of lead times, COSMO-MOS delivers on a daily basis $M \cdot N \cdot L$ distinct statistical models that are then operated once, on that given day.

Problematically, the learning period is usually short, from few days up to three months. This modus operandi is not optimally suited for intermittent or rare events that may not have occurred during the learning period. As an example, would fog suddenly develop on a given morning after a long period free of fog, then the system would not have any “clue” or “experience” about fog and would accordingly not be able to deliver a suitable fog forecast. Instead, the software is adapted to correct systematic

errors for continuous parameters that are sure to be measured during the learning period.

Conceptually, COSMO-MOS does not seem to be the most appropriate option to implement logistic regression in order to predict intermittent or rare events. Practically, other issues also arise. The part of the software handling regression still needs adjustments before it can work without bugs in operational routine. Being still in development, it is very sensible to any update of the system, whose regular improvements are usually only verified to be compatible with finite products. Thus, its functioning is easily disrupted, and the branch of the software handling logistic regression is currently not running.

Provided that these practical issues are solved, we would still face some of the software's limitations. The most important one is that COSMO-MOS is configured to generate itself the needed data sample of predictand observations, from the DWH database. Therefore, regression can only be performed on parameters available in this system. In order to predict fog or thunderstorms, we need observations which are at the moment unavailable in this database, which we have so far obtained only from METAR data through a self-made Matlab script. If COSMO-MOS is to be used to forecast visibility or thunderstorm, it would be necessary to either add new parameters to the DWH, or to let it support other data sources.

First steps towards implementing the project through COSMO-MOS would thus start with internal updates making it compatible with the current versions of all systems at Meteoswiss. Then, it is needed to integrate a METAR-decoding script in COSMO-MOS to have access to new parameters. Such a mandate has already been given to the KmD. Another important step is then the simplification of the configuration file of COSMO-MOS. Currently, the section of this file commanding the extraction of predictors data through Fieldextra can only be handled by people with specific knowledge of Fieldextra. This weakness makes COSMO-MOS in practice not easily accessible to meteorological engineers who would further work on the implementation.

Overall, COSMO-MOS has a complex and inflexible architecture, and a lot of implementation work needs to be done before Meteoswiss is able to provide probabilistic forecasts for TAFs through this software.

6 Conclusion

Reliable weather forecasts play a crucial role in ensuring the safety and efficiency of today's global airline traffic. Meteorological phenomenon such as intense fog or thunderstorms can pose dangerous threats to flights if they are not well managed. Forecasting them early enables pilots to plan ahead and avoid having to make difficult choices (for example, whether to continue, land or deviate) in haste. The anticipation of adverse meteorological conditions also has a positive economic impact. By allowing airline managers to take better logistics decisions, for example regarding itineraries or level of fuel aboard, airlines companies can optimize their costs.

Terminal Airport Forecasts (TAF) are weather forecast messages aimed at the aviation community, which are routinely produced worldwide for almost every airport. As a site-specific forecast, the TAF delivers information regarding the probable evolution of the weather conditions of a given airport at a very local scale, and thus is an integral part of the pre-flight meteorological review of every airman. However, producing TAFs for airports worldwide on a daily basis is a costly and challenging task, since these weather reports are nowadays written manually by human forecasters. This is why several weather agencies have been developing statistical methods aimed at producing specific elements within a TAF report.

This work is part of the general effort of meteorological institutions towards the automatization of the process of writing a TAF. It focuses on the probabilistic items of a TAF, which quantify the uncertainty of changes in weather pattern through probabilities. Its purpose is to develop statistical models which can deliver probabilistic forecasts based on data from the most local NWP model used at Meteoswiss, COSMO-2, through a method called post-processing. In particular, we studied the case of probabilistic forecasts for rare meteorological events, targeting strong wind gusts, low visibility and thunderstorms, three phenomenon of great importance in flight planning.

The strength of this work consists of both its methodology and the study of its application to COSMO-2 data routinely produced by Meteoswiss. Based on the work of *D. W. Hosmer and Sturdivant* (2013), we implemented algorithms which select in a stepwise manner appropriate predictors for each forecast equation. Furthermore, we improved the predictor selection by enabling models to take into account the evolution in time of the predictors in the forecast equation. This method is based on a statistical learning process made on a large panel of data, which frames the weather conditions from -12h to the time 0 of each phenomenon to forecast. This allows us to identify sequences of events that statistically lead to an extreme meteorological phenomenon.

The results of the estimation for wind gusts, visibility and thunderstorm show that COSMO-2 provides good data which is appropriate to derive probabilistic forecasts. In each case, the predictors selected by the stepwise algorithm were in agreement with our meteorological knowledge. The models were

able to identify the parameters which are important throughout the formation of the meteorological events we considered. Satisfyingly, for both wind gusts and visibility, the forecasts delivered by the model are consistent with the 30 to 40% certainty required by the TAF. Additionally, we studied the effect of oversampling strategies on model estimation, in order to pally the main problem in forecasting extreme meteorological events, which is the lack of observed cases. It appeared that oversampling can lead to slightly better models, but overall a large amount of data is required.

We conclude that this work represents a first step towards rethinking the post-processing framework in place at Meteoswiss, aiming to deliver a greater variety of products, including probabilistic forecasts for TAFs.

Abbreviations

TAF	Terminal Airport Forecast
METAR	Meteorological Aerodrome Report
MOS	Model Output Statistics
NWP	Numerical Weather Prediction
COSMO	Consortium for Small Scale Modeling
DWH	Data Warehouse

List of Figures

Figure 1 Operational weather forecasting : From data assimilation to post-processing . . . 12
Figure 2 The three nested COSMO numerical weather models 13
Figure 3 Learning process VS Operational forecasting process 16
Figure 4 The verification of forecasts within the learning process 20
Figure 5 Example reliability diagrams 22
Figure 6 Example refinement distributions 23
Figure 7 Outputs of estimation results for wind gusts models above 15kt 28
Figure 8 Diagnostic plots for wind gusts models above 15 kt 30
Figure 9 Diagnostic plots for wind gusts models above 20 kt 31
Figure 10 Diagnostic plots for wind gusts models above 25 kt 32
Figure 11 Diagnostic plots for scaled forecasts of wind gusts models above 25 kt 34
Figure 12 Diagnostic plots for wind gusts models with a season binary predictor 35
Figure 13 Illustration of the temporal evolution of predictors in one regression model 37
Figure 14 General tendency regarding the choice of predictors based on all visibility models studied 39
Figure 15 Outputs of estimation results for visibility models 40
Figure 16 Reliability diagrams for visibility models 41
Figure 17 Refinement distributions for visibility models 42
Figure 18 Reliability diagrams for visibility models estimated with oversampled data 44
Figure 19 Refinement distributions for visibility models estimated with oversampled data . . . 45
Figure 20 Diagnostic plots for thunderstorm models 49
Figure 21 Refinement distributions for thunderstorm models estimated with oversampled data 51

List of Tables

Table 1 Initial pool of parameters for the wind gusts models 26
Table 2 Ordering of predictors resulting from the stepwise selection procedure 27
Table 3 Initial pool of parameters for the visibility models 38
Table 4 Ordering of predictors resulting from the stepwise selection procedure 39
Table 5 Initial pool of parameters for the thunderstorm models 47
Table 6 Ordering of predictors resulting from the stepwise selection procedure 48

References

D. W. Hosmer, S. L., and R. X. Sturdivant (2013), Applied logistic regression.

Greene, W. (1993), *Econometrics Analysis*, Pearson Education.

Wilks, D. (2006), *Statistical Methods in the Atmospheric Sciences*, Elsevier.

Acknowledgment

I am very grateful to all the people who helped me throughout this work. In particular, I would like to thank warmly Jacques Ambühl who dedicated a large amount of time to supervising this project. Jacques Ambühl incessantly guided me with ideas regarding the statistical methods described in this work, but also shared with me his knowledge of meteorology, which before this work I had no understanding of. I would also like to thank Andreas Asch, who works at the Meteoswiss office at Zürich Kloten airport, and kindly made me benefit from his expertise in the field of Terminal Airport Forecasts as well as aviation meteorology in a broader sense. Also, my thanks go to Petra Baumann, without whose efficient computing support this project would not have been possible.

Working on this project in such a friendly working environment as Meteoswiss was an unforgettable experience. Therefore, I am also very grateful to Philippe Steiner at the head of the APN department for giving me this opportunity and supporting me throughout my internship.

MeteoSchweiz
Operation Center 1
CH-8058 4 Zürich-Flughafen

T +41 58 460 91 11
www.meteoschweiz.ch

MeteoSchweiz
Flugwetterzentrale
CH-8060 Zürich-Flughafen

T +41 43 816 20 10
www.meteoswiss.ch

MeteoSvizzera
Via ai Monti 146
CH-6605 Locarno Monti

T +41 91 756 23 11
www.meteosvizzera.ch

MétéoSuisse
7bis, av. de la Paix
CH-1211 Genève 2

T +41 22 716 28 28
www.meteosuisse.ch

MétéoSuisse
Chemin de l'Aérologie
CH-1530 Payerne

T +41 26 662 62 11
www.meteosuisse.ch