Dietz, Sebastian J.; Kneringer, Philipp; Mayr, Georg J.; Zeileis, Achim

Working Paper

Forecasting low-visibility procedure states with tree-based statistical methods

Working Papers in Economics and Statistics, No. 2017-22

Provided in Cooperation with:
Institute of Public Finance, University of Innsbruck

Suggested Citation: Dietz, Sebastian J.; Kneringer, Philipp; Mayr, Georg J.; Zeileis, Achim (2017) : Forecasting low-visibility procedure states with tree-based statistical methods, Working Papers in Economics and Statistics, No. 2017-22, University of Innsbruck, Research Platform Empirical and Experimental Economics (eeecon), Innsbruck

This Version is available at:
https://hdl.handle.net/10419/180172

University of Innsbruck

Forecasting low-visibility procedure states with tree-based statistical methods

Sebastian J. Dietz, Philipp Kneringer, Georg J. Mayr, Achim Zeileis

Working Papers in Economics and Statistics

2017-22

Research Platform
eeecon
Empirical and Experimental Economics
University of Innsbruck
http://eeecon.uibk.ac.at/

University of Innsbruck
Working Papers in Economics and Statistics

The series is jointly edited and published by

- Department of Banking and Finance
- Department of Economics
- Department of Public Finance
- Department of Statistics

Contact address of the editor:
Research platform “Empirical and Experimental Economics”
University of Innsbruck
Universitaetsstrasse 15
A-6020 Innsbruck
Austria
Tel: + 43 512 507 7171
Fax: + 43 512 507 2970
E-mail: eeecon@uibk.ac.at

The most recent version of all working papers can be downloaded at
http://eeecon.uibk.ac.at/wopec/

For a list of recent papers see the backpages of this paper.

Forecasting Low-Visibility Procedure States with Tree-Based Statistical Methods

Sebastian J. Dietz
Universität Innsbruck

Philipp Kneringer
Universität Innsbruck

Georg J. Mayr
Universität Innsbruck

Achim Zeileis
Universität Innsbruck

Abstract

Low-visibility conditions at airports can lead to capacity reductions and therefore to delays or cancelations of arriving and departing flights. Accurate visibility forecasts are required to keep the airport capacity as high as possible. We generate probabilistic nowcasts of low-visibility procedure (*lvp*) states, which determine the reduction of the airport capacity due to low-visibility. The nowcasts are generated with tree-based statistical models based on highly-resolved meteorological observations at the airport. Short computation times of these models ensure the instantaneous generation of new predictions when new observations arrive. The tree-based ensemble method "boosting" provides the highest benefit in forecast performance. For *lvp* forecasts with lead times shorter than one hour variables with information of the current *lvp* state, ceiling, and horizontal visibility are most important. With longer lead times visibility information of the airport's vicinity, humidity, and climatology also becomes relevant.

Keywords: aviation meteorology, visibility, nowcast, decision tree, bagging, random forest, boosting.

1. Introduction

Low-visibility conditions reduce the operational capacity of airports. At peak hours capacity reductions may lead to flight delays or even cancelations. Consequently, costs for airports and airlines as well as the environmental impact increase. These effects grow considerably when low visibility is predicted incorrectly. Pessimistic visibility predictions may cause an overly strong reduction of the airport capacity. Hence, scheduled short-distance flights may be kept on ground at their airport of departure to ensure the flight safety. This action results in costs for airlines, as well as in decreased revenues from landing fees for airports. On the other hand, if visibility is predicted too optimistically, the number of arriving aircraft might exceed the remaining capacity of an airport. Consequently, many en route flights have to circle into stacks until landing is possible. Results are increased costs for airlines through crew scheduling, fuel consumption, and emissions. Accurate forecasts of low visibility can help to avoid such scenarios. Precise forecasts can be used by air traffic controllers to plan the number of arriving and departing aircraft in a safe and economical way.

The reduction of aircraft movements due to low visibility depends on different safety operations, which decelerate the air traffic and increase the distance between successive arrivals and departures. The execution of these operations is defined by low-visibility procedure (*lvp*) states. Upcoming available capacity is therefore directly connected to *lvp* forecasts at airports. Typically, *lvp* states are determined by particular thresholds of horizontal and vertical visibility. The exact thresholds and resulting capacity reductions, however, vary for each airport. In this study *lvp* forecasts are investigated for Vienna International Airport (VIE). Nowcasts with lead times up to two hours are of main interest. Currently, human forecasters generate the *lvp* predictions at VIE by using observations at the airport and information from numerical weather prediction (NWP) models. Providing probabilistic predictions of *lvp* states, however, would support the forecasters producing their predictions.

Two different types of approaches are typically employed for automatic predictions of meteorological variables. The first one is based on numerical modeling and uses all relevant physical equations to compute forecasts. Many physically-based models for visibility and fog forecasts were developed in the past, e.g. HRRR (Benjamin *et al.* 2016), The London Model (Boutle *et al.* 2016), and PAFOG (Bott and Trautmann 2002). Generally, these models are computationally expensive and special end-user related variables such as *lvp* have to be derived afterwards from their output.

The second approach uses historical data to train a statistical model and produces probabilistic forecasts of variables such as *lvp* directly. This approach is generally computationally cheaper so that new predictions are available instantaneously with new input information. In the past various statistical methods were tested to generate visibility forecasts. Vislocky and Fritsch (1997) for example used multiple linear regression to generate visibility forecasts from observations. Their forecasting system was later improved by Leyton and Fritsch (2003, 2004) through increasing the density and frequency of the surface observations. An operational visibility forecasting system for several lead times and locations was developed by Ghirardelli and Glahn (2010), using again multiple linear regression. Glahn *et al.* (2017) combined this system with the physically-based forecasts of Benjamin *et al.* (2016) to improve the performance. Other statistical techniques to forecast visibility are for example neural networks (e.g., Pasini *et al.* 2001; Marzban *et al.* 2007), Bayesian model averaging (e.g., Roquelaure *et al.* 2009), or decision trees (e.g., Bartoková *et al.* 2015; Dutta and Chaudhuri 2015). Herman and Schumacher (2016) compared various statistical methods for visibility predictions at airports and found that no specific model performs best overall.

The first investigation of *lvp*, which is the relevant variable for airport operations, was conducted in a companion paper by Kneringer *et al.* (2017). They used ordered logistic regression (OLR) models to produce *lvp* nowcasts at VIE for the cold season (September to March) with lead times up to 2 hours and 30-minutes resolution. To provide decision makers with more detailed information we increase the temporal resolution of the predictions to 10 minutes and extend the predictor variable setup of Kneringer *et al.* (2017) with temporally lagged and spatially averaged visibility information. During the cold season the peak of *lvp* is in the early morning hours, coinciding with the airport's rush hours (Kneringer *et al.* 2017). Since low visibility has most impact during these times we generate *lvp* predictions in this paper only from September to March during 6 and 9 UTC. Tree-based statistical methods are used as a flexible nonparametric alternative to the parametric OLR for generating the forecasts. The predictions of these models are compared amongst each other and to the forecasts of the OLR models of Kneringer *et al.* (2017). Furthermore, the impact of the predictors on the forecasts

is analyzed to provide information of the most important inputs for statistically-based *lvp* nowcasts.

2. Data

In this section the predictand and the predictors used for *lvp* nowcasts are described. The first part is about the determination of *lvp*, which has to be derived from horizontal and vertical visibility measurements. In the second part the predictor variables are described. These variables contain observations at VIE and its vicinity from September 2012 to March 2017 between 6 and 9 UTC.

2.1. Low-Visibility Procedure States

The *lvp* state is an ordered categorical variable that specifies the application of safety procedures at airports due to low visibility. Generally, *lvp* states are determined by nonlinear, threshold-bounded combinations of horizontal and vertical visibility. Naturally, the exact thresholds and the associated capacity reductions vary for each airport. At VIE, three *lvp* states are specified (Tab. 1). The horizontal visibility used for *lvp* determination is runway visual range (*rvr*), which is defined as the distance over which the pilot of an aircraft on the centerline of the runway can see the runway surface markings or the lights delineating the runway or identifying its center line (International Civil Aviation Organization 2005). The vertical visibility threshold is determined by ceiling (*cei*), the altitude of the cloud base with at least five octa coverage.

Each touchdown point at VIE is equipped to quantify *rvr* and *cei*. While *rvr* is measured directly by transmissiometers, *cei* has to be derived from ceilometer measurements. Typically, this procedure is conducted by human forecasters, who determine an airport-averaged *cei* every 30 minutes by comparing the backscatter profiles of several ceilometers at the airport. For the generation of 10-minutes resolved *lvp*, however, *cei* is required in a 10-minutes resolution. Hence, we developed an algorithm in collaboration with the forecasters at VIE to compute touchdown point specific *cei* with a 10-minute resolution. The predictand *lvp* is computed afterwards by using the touchdown-point-specific *cei* and the 10-minutes resolved *rvr* at each touchdown point, respectively. In this study we focus on forecasts at the touchdown point with the highest climatological *lvp* state occurrence.

Table 1: Definition of *lvp* states with their occurrence probability for the forecast period (cold seasons and early morning hours from 2012 to 2017) and the capacity limitations pertaining to each *lvp* state at VIE. The operational capacity utilization for each *lvp* state is given relative to the maximum capacity.

lvp state	*rvr*		*cei*	occurrence	capacity
0				90.4%	100%
1	<1200 m	or	<90 m	2.5%	75%
2	<600 m	or	<60 m	5.8%	60%
3	<350 m			1.3%	40%

2.2. Predictor Variables

All meteorological variables used as predictors are available in a 10-minutes output frequency. Moreover, most of them are measured at multiple locations within the vicinity of VIE. For example, wind speed is observed at nine different locations within an area of about 8 km^2. These measurements are highly correlated and thus the forecast performance of the statistical models would not increase when using all wind speed measurements. To this end, we conduct a manual preselection of the observations at VIE to generate a highly informative predictor setup. This predictor setup consists of single point measurements, derived observations, and climatological information. Some variables occur several times through spatial averaging and temporal lagging (Tab. 2, large setup).

The point measurements included in this predictor setup are *rvr*, horizontal visibility (*vis*), air temperature (*tl*), relative humidity (*rh*), wind direction (*dir*), and wind speed (*ff*). Postprocessed information from the observations are *lvp*, *cei*, dewpoint depression (*dpd*), temperature difference between 5 cm above the surface and 2 m (*dts*), and wind speed difference between the height of 100 m and 2 m (*ffd*). All locations of the point measurements are close to the touchdown point with exception of *vis*, which is observed at a distance of about 7 km northeast of the touchdown point. This location is well-known to the forecasters since radiation fog often forms there first.

Table 2: Predictor variables used in the statistical models. The variables on the upper part are used in the "standard setup" while the "large setup" additionally considers the variables on the lower part. Variables available in the standard setup contain information at forecast initialization and at the touchdown point. In the large setup *lvp*, *cei*, and *rvr* information is available also 10-minutes lags to forecast initialization and averaged for runway and airport, respectively. The variables dir_N, dir_{SE}, *rr*, and *sza* used in the standard setup are not included in the large setup (for more details to the standard predictor setup see Kneringer *et al.* 2017).

Variable	Unit	Description
lvp	[0,1,2,3]	Low-visibility procedure
rvr	[m]	Runway visual range
cei	[m]	Ceiling
vis	[m]	Visibility
dts	[°C]	Temperature difference $_{2m-5cm}$
dpd	[°C]	Dew point depression
rh	[0–100]	Relative humidity
dir_N	[no,yes]	Wind direction from north $_{binary}$
dir_{SE}	[no,yes]	Wind direction from south east $_{binary}$
rr	[no,yes]	Rain in the last 12 hours $_{binary}$
sza	[°]	Solar zenit angle
dir	[°]	Wind direction
tl	[°C]	Air temperature
ff	[ms^{-1}]	Wind speed
ffd	[ms^{-1}]	Wind speed difference $_{100m-2m}$
cc	[0–1]	Conditional climatology

Originally, *rvr* is censored at 2000 m because visibility above this range is not relevant for the landing approach. For possible fog advection, however, visibility information from ranges above 2000 m is required. Hence, we replace all censored *rvr* values with visibility information measured next to the transmissiometers.

Since *lvp* is a combination of *rvr* and *cei* these three variables are expected to have the strongest influence on *lvp* nowcasts. We therefore generate spatially averaged and temporally lagged predictors from these variables. Spatially averaged observations are averages at the runway and from the complete airport. Runway averages only contain information directly from the runway. In case of *rvr* three sensors are used for averaging (located at both touchdown points and the midpoint); *cei* averages at the runway contain information of two ceilometers (located at both ends of the runway). Airport averages, however, contain all observations of one variable available in the airport area and therefore cover an area of about 8 km^2. Averaged *lvp* information of the runway and airport are computed by using runway and airport averaged values of *rvr* and *cei*, respectively. The temporally lagged information used is *lvp*, *rvr*, and *cei* observations at the touchdown point from 10 minutes before forecast initialization.

Additionally, we include a conditional 31-day *lvp* 0 state climatology (*cc*) for each day of the year, containing the *lvp* 0 proportion of the actual day plus/minus 15 days around this day from September 2012 to March 2017 between 6 and 9 UTC.

3. Methods

For the generation of statistical *lvp* state forecasts ordered response models are required, which can be easily accommodated in decision trees. Such decision trees have a flowchart-type structure and are therefore easy to understand and interpret. Typically, the forecast performance of decision trees can be improved by aggregating an ensemble of trees using methods like bagging, random forest, or boosting (James *et al.* 2014). While such an ensemble usually improves predictive performance the interpretability is not straightforward anymore due to the more complex structure. However, computing variable importance measures still allows to determine the predictors with the highest benefit for the forecast. In the following, an overview is provided of the tree-based forecasting methods along with details for validation and interpretation.

3.1. Decision Tree

Decision trees are composed of a recursive partitioning algorithm, which splits the training sample into different cells, depending on the association between the forecast variable and its predictors. After the splitting procedure a constant model is usually computed for each terminal cell.

Classic decision trees, such as CART (Breiman *et al.* 1984) and C4.5 (Quinlan 1993) select their split variables by maximizing an information measure of node impurity for all possible splits. Such variable selection criteria, however, prefer split variables with many possible splits or missing values and tend to overfit the data (Hothorn *et al.* 2006b). The decision trees developed in this study are based on the unified conditional inference framework of Hothorn *et al.* (2006b). These trees separate the variable selection and splitting procedure into two steps and do not suffer from a systematic tendency towards split variables with many

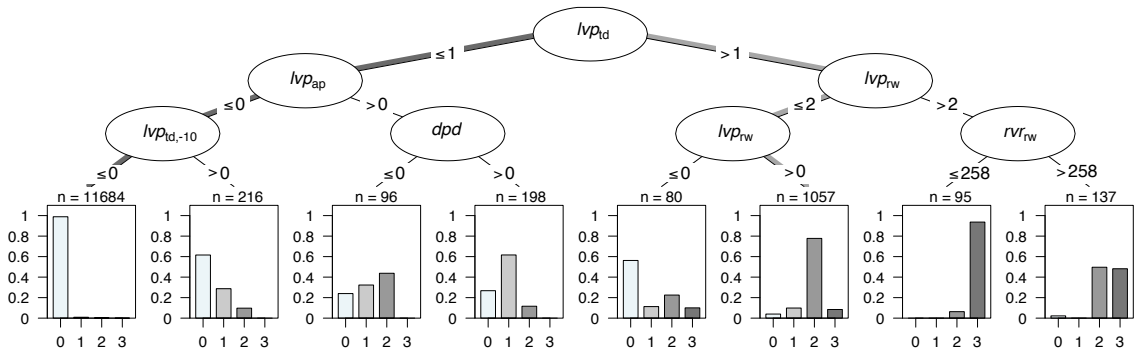

Figure 1: Example of a decision tree for half-hourly *lvp* state forecasts at Vienna International Airport. The subscripts refer to: at touchdown point (td), averaged over runway (rw) or airport (ap), 10 minutes prior (-10); *dpd* is dew point depression, n is the number of observations in the terminal cell. The two parts highlighted in different shades of gray refer to the examples discussed in the text.

possible splits or many missing values (Hothorn *et al.* 2006b).

In the first step of the tree growing process the association between the response and each of the covariates is computed by deriving the p-value for each association. In case of ordinal variables such as *lvp*, the p-values are derived by an χ^2-test (Hothorn *et al.* 2006a). If the minimum p-value is below a prespecified nominal level α, the covariate with lowest p-value is selected as split variable. Otherwise no split is computed. Hence, α can be seen as statistical stopping criterion to avoid overfitting.

For the chosen split variable the optimal split point is computed by applying a second test statistic over all subsets which can result from possible splits. The split point is set where the discrepancy between two subsets is maximized. In case of *lvp* forecasts, the χ^2-test is used again as the test statistic (Hothorn *et al.* 2006a). Both steps, the variable selection and the split point determination, are repeated recursively until a stopping criterion is reached (e.g. all p-values exceed α or a maximum growth depth is reached). The conditional distribution of the response in the particular terminal cells represents the probabilistic forecast of the tree.

As an example of the described algorithm a decision tree for a 30-minutes *lvp* forecast at VIE is shown in Fig. 1. The first split variable of this tree is the current *lvp* state at the touchdown (lvp_{td}) of the runway. If it is 2 or 3 (> 1), the next split variable is the averaged *lvp* state at the runway (lvp_{rw}). If, however, lvp_{td} is 0 or 1, the *lvp* state over the airport (lvp_{ap}) is required to generate a forecast. The process of comparing the observations with the splitting rule has to be repeated at each subsequent split point until a terminal cell is reached and a probabilistic forecast can be generated (histograms in Fig. 1).

A common observation at VIE is for example *lvp* state 0 at touchdown, runway, and the complete airport at forecast initialization as well as 10 minutes before. The resulting prediction from this observation is almost 100% *lvp* 0 (see dark gray path in Fig. 1). In the tree building process most of the observations drop into this terminal cell ($n = 11684$). The second most common combination of the tree ($n = 1057$, light gray path) results in a prediction of 4% *lvp* 0, 10% *lvp* 1, 78% *lvp* 2 and 8% *lvp* 3. This prediction is the most frequent one for prevailing *lvp* conditions (for example *lvp* 2 is observed at each measurement site at forecast initializa-

tion). All other possible predictions occur only rarely, which can be seen in the number of observations in the terminal cells.

3.2. Tree-Based Ensemble Approaches

A well-known weakness of single decision trees is their high variance, i.e., the tree structure may change considerably when learning it on randomly perturbed (sub)samples from the same data set. Growing an ensemble of decision trees and aggregating them into one model typically reduces the variance and improves the forecast accuracy (James *et al.* 2014). Hence, the ensemble methods bagging, random forest, and boosting are employed subsequently.

Bagging
In bagging multiple trees are grown and their predictions are merged to reduce the variance over single decision trees (Strobl *et al.* 2009; Bühlmann and Yu 2002). In the first part of bagging multiple training samples of the size of the original training sample are generated by drawing randomly observations from the original training sample with replacement (bootstapping). Afterwards, an individual tree is fitted to each new sample and the predictions of the individual trees are merged. We use the aggregation pattern developed by Hothorn *et al.* (2004), which collects all observations contained in the obtained terminal cells of the particular trees. The forecast of bagging is then just the distribution of the collected observations.

Random Forest
While bagging considers all predictor variables for splitting at any stage of any tree, random forest only considers a different randomly drawn subset of predictors at each stage. Thus, bagging is a special case of random forests (Strobl *et al.* 2009) but the latter leads to trees that are less correlated with each other.

The principle of including additional randomness into the single trees can often improve the prediction accuracy of an ensemble. Commonly one predictor has potentially stronger power than the other covariates. Most of the bagged trees will therefore select this predictor as first split variable so that the predictions of the individual trees will have a strong correlation, and consequently only a small variance reduction over a single tree when aggregating these trees (James *et al.* 2014). Trees grown in a random forest, on the contrary are quite diverse amongst each other. Through the random preselection of possible split variables in random forests weaker predictors have the chance to be selected first and may reveal interaction effects with other variables that would have been missed otherwise (Strobl *et al.* 2009).

Boosting
The third method used to combine an ensemble of decision trees is boosting. New decision trees grow in boosting always on forecast information of previously grown trees (James *et al.* 2014). The new tree is fitted therefore to residual information of the previous trees and is subsequently added to them. Afterwards the residual information of the new model is computed and a new tree is fitted to it. The computation of the residual information varies with different boosting methods. In this study we use the component-wise gradient boosting algorithm developed by Bühlmann and Hothorn (2007), which computes the residual information with the negative gradient vector of the loss-function from the current model. To boost ordinal response variables such as *lvp* the log-likelihood of the proportional odds model (Agresti 2003)

is used as loss-function (Schmid *et al.* 2011). Moreover, a shrinkage parameter is used for tree aggregation to avoid overfitting.

The trees used in the boosting algorithm are the conditional inference trees of Hothorn *et al.* (2006b, see Sec. 3.1). Since boosting is able to project additive data structures, and decision trees can model nonlinear data features, boosting trees can project both, additive and non-linear data features. The branch depth of the individual trees can be used to control whether boosting rather captures additive or nonlinear structures.

3.3. Computational Implementation

Decision Tree

The conditional inference trees used are implemented in the R package **party** (Hothorn *et al.* 2017b). Each tree developed in this investigation can grow until the p-value between the response and its most associated predictor exceed the default α value of 0.05.

Bagging and Random Forest

Bagging and random forest models used in this investigation are also implemented in the R package **party** (Hothorn *et al.* 2017b). Both models contain 500 single decision trees, which turned out to be a reasonable number to ensure short computation times with good forecast performance. The settings of the individual trees in bagging and random forest are default. In random forests the number of randomly preselected split variables is set to five, which is approximately the square root of the number of predictors (typically recommended in the literature; see James *et al.* 2014).

Boosting

The boosting method used in this study is implemented in the R package **mboost** (Hothorn *et al.* 2017a). Each boosting model consists of 1500 trees, which is a reasonable number to guarantee high forecast performance with low computational cost. The shrinkage parameter for tree aggregation is set to the default value 0.1. Each tree in the boosting can grow to a maximum branch depth of three because we rather combine several small trees instead of few large ones. Using this adjustment we are able to model both, additive and non-linear data structures.

3.4. Reference Model

To analyze the benefit of the different tree-based models we compare their forecast performance to an ordered logistic regression model (OLR), which was shown to outperform climatology, persistence, and predictions of human forecasters at VIE (Kneringer *et al.* 2017).

For consistency in model comparison we change the half hourly averages of *cei* and *lvp* used by Kneringer *et al.* (2017) to the 10-minutes resolved values described in Sec. 2.1. Furthermore, the size of the training sample is reduced to 5 cold seasons instead of the original 9 cold seasons.

3.5. Forecast Verification

The forecast performance of the models is analyzed using the ranked probability score (RPS;

Epstein 1969; Murphy 1971), which is a well-known validation criterion for probabilistic forecasts of ordered response variables such as *lvp* (Wilks 2011). The RPS of a forecast-observation pair i is computed by the squared errors of the cumulative distribution function of the forecast probabilities with respect to the observations:

$$\mathrm{RPS}_i = \frac{1}{J-1} \sum_{s=1}^{J} \left[\sum_{j=1}^{s} y_{ij} - o_{ij} \right]^2,$$

with the forecast probabilities y_{ij} and observations o_{ij} for each category $j = 1, \ldots, J$. A perfect forecast results in an RPS of 0; the worst possible forecast has an RPS of 1. For model comparison we average the RPS values of all forecast-observation pairs produced from one model.

Moreover, for direct comparison of the forecast performance of a model relative to a reference the ranked probability skill score (RPSS) can be computed. This criterion requires just the RPS of the model and the reference for its computation:

$$\mathrm{RPSS} = 1 - \frac{\mathrm{RPS}}{\mathrm{RPS}_{\mathrm{reference}}}.$$

The RPS values of each model are computed by applying a season-wise 5-fold cross-validation. To this end, the dataset is subdivided into five samples, each of which contains one cold season. Afterwards, the models are trained on four samples and validated on the remaining one. This training and validating procedure is repeated five times, always with another sample for validation.

For the computation of the model uncertainty we additionally bootstrap the RPS values calculated in the cross-validation algorithm above (bootstrapping described in Sec. 3.2 – bagging). Therefore we generate 2000 bootstrapped samples and compute the mean RPS for each sample. The model uncertainty is described by the distribution of the mean RPS values.

3.6. Variable Importance Measurement

To identify the variables with the highest impact on the forecasts we apply variable permutation tests in which the true information of a particular predictor is replaced by randomly drawn information from the predictor's true distribution. The information of a predictor is thus intermingled randomly. Stronger decrease in forecast performance shows a higher impact of the permuted variable.

To conduct the variable permutation test we again use 5-fold cross-validation. Therefore we always fit the models to four cold seasons and test them on the remaining one. After generating predictions on the original test sample we randomly permute one predictor variable and produce forecasts on the modified sample. This procedure is repeated for each predictor variable in the test sample. The performance of the predictions from each modified sample is computed by the RPS (Sec. 3.5). We have to mention that permuting predictors with a strong association amongst others can lead to discrepancies in the model and therefore to strong effects in the forecast performance (for example lvp_{td} to cei_{td} and rvr_{td}). Nevertheless, the variables with the highest impact on the forecast performance can be effectively identified using this analysis

4. Results

4.1. Model Comparison

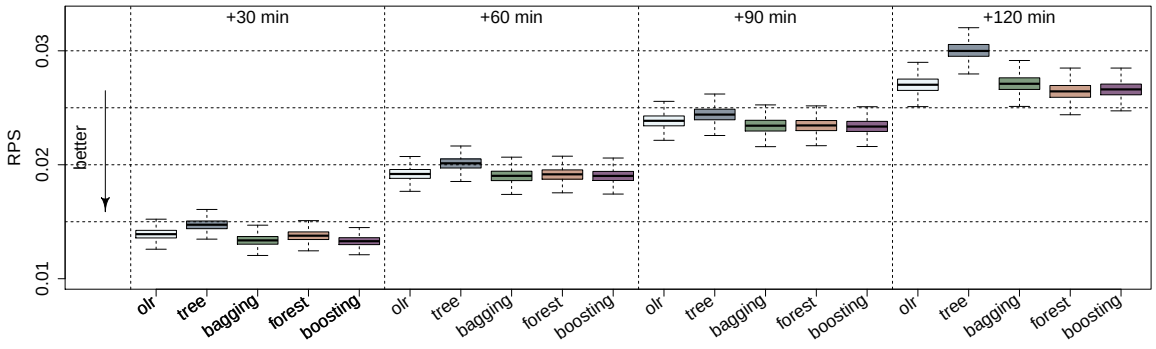

Figure 2: Ranked probability scores (RPS) of the OLR model from Kneringer *et al.* (2017) (*olr*) and the tree-based models decision tree (*tree*), bagging (*bagging*), random forest (*forest*), and boosting (*boosting*) for lead time +30, +60, +90 and +120 minutes. For each model the standard predictor variable setup described by Kneringer *et al.* (2017) and in Tab. 2 is used.

The comparison of tree-based models and the OLR model (*olr*) of Kneringer *et al.* (2017) for *lvp* nowcasts is shown in Fig. 2. All models established for this comparison are based on the standard predictor setup described in Tab. 2 and tested for lead time +30, +60, +90, and +120 minutes.

OLR outperforms decision trees (*tree*) at each lead time. The benefit of OLR over decision trees is between 2.5 and 11 percent with highest value at +120 minutes. Tree-based ensemble methods, however, outperform OLR at almost all lead times. At the lead time +30 minutes random forest (*forest*) performs similar to OLR. With longer lead times, the benefit of random forest increases and is highest at the lead time +120 minutes. In contrast, bagging (*bagging*) has the highest benefit over OLR at lead time +30 minutes. At longer lead times this benefit disappears and bagging performs similar to OLR. Boosting (*boosting*) outperforms OLR at each lead time and has highest benefit over all models except for random forests at +120 minutes. The benefit of boosting over OLR varies between 1 and 4 percent.

These results show that aggregating multiple trees substantially improves the forecast performance of a single decision tree. The most efficient aggregation method concerning *lvp* forecasts is boosting. Random forests perform best at +120 minutes lead time when the forecast depends on multiple predictors instead of only few with very high importance. If, however, only few of the available predictors control the forecast like at short lead times, where the forecast relates strongly on persistence, the performance of random forests is worse. In such cases bagging leads to better results.

4.2. Impact of More Predictors

To improve the forecast performance of the models established in Sec. 4.1 we provide them the large predictor setup specified in Sec. 2.2. The performance of the tree-based models with the large predictor setup relative to a reference is shown in Fig. 3. The reference model for

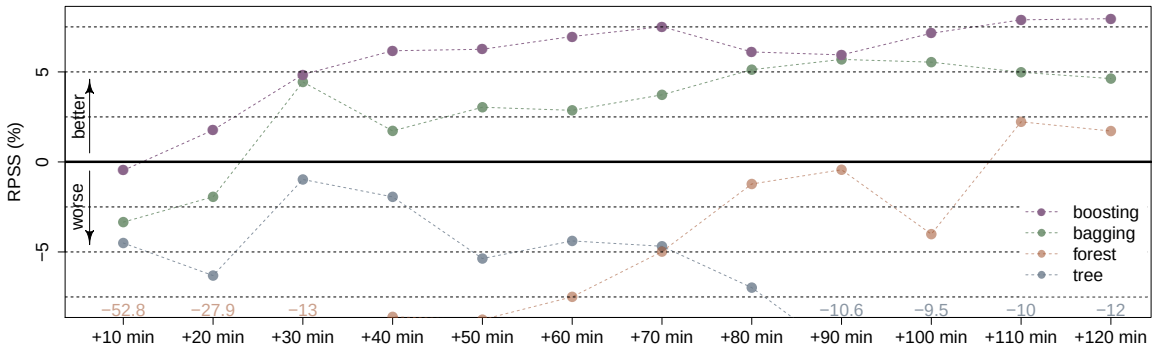

Figure 3: Ranked probability skill score (RPSS) comparison of the tree-based models with the large predictor setup defined in Sec. 2.2 relative to the boosting models defined in Fig. 2. The dots show the median values of the RPSS for the lead times +10 minutes to +120 minutes in 10-minute resolution.

this analysis is the boosting model from Sec. 4.1, which is the best performing model with the standard predictor setup.

Single decision trees with the large predictor setup are outperformed by the reference model at each lead time. The difference in forecast performance becomes even bigger with longer lead times. On the contrary, the benefit of the tree-based ensemble models random forest, bagging, and boosting compared to the reference increases slightly with longer lead times. Random forest has advantages only for the longest lead times. Bagging outperforms the reference after +20 minutes, and boosting after a lead time of +10 minutes. At lead time +120 minutes the benefit over the reference is approximately 5 percent for bagging and 7 percent for boosting. Again boosting performs best.

At the shortest lead time boosting with the large predictor setup performs almost equally to the boosting with the standard setup (reference). The other models again perform somewhat worse, especially random forest which shows a decrease in forecast performance of more than 50% compared to the reference. The reason therefore is the reduced set of randomly drawn split variables in the tree building process. Since the models mainly try to reproduce the persistence at the shortest lead times, the single trees strongly require information of current *lvp*. If this predictor is not included in the randomly selected split variable setup, the single trees perform badly. Aggregating many poorly performing trees results again in worse forecast performance.

4.3. Impact of Predictor Variables

The analysis of the variables with highest impact on the forecasts is conducted by variable permutation tests (Sec. 3.6). Therefore, the forecast performance from a test sample with randomly permuted information of a particular predictor is computed and compared to the forecast performance of the original test sample. Fig. 4 shows the fractional changes in RPS for predictions on permuted test samples relative to the original sample for bagging (4a) and boosting (4b). The permuted variables with strongest decrease in forecast performance are plotted for lead time +10, +60, and +120 minutes.

The highest impact on *lvp* nowcasts with a lead time of +10 minutes has the current *lvp* state

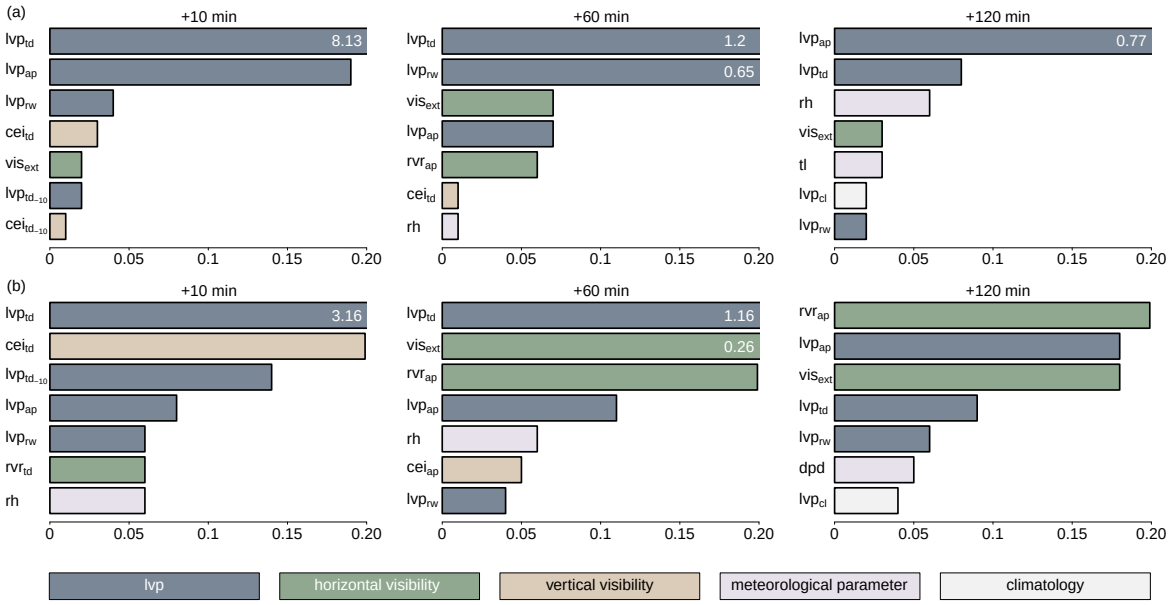

Figure 4: Variable importance analysis for the models bagging (a) and boosting (b). Each bar in the plot shows the fractional decrease in forecast performance for a test sample with a randomly permuted predictor compared to the original test sample. Note that the bars are cut at a maximum fractional decrease of 0.2. Higher values in fractional decrease are written in the bars. For each model the most important variables are plotted for the lead times +10, +60 and +120 minutes.

at the touchdown point (lvp_{td}). Permuting this variable decreases the forecast performance of bagging on average more than 8 times, and of boosting more than 3 times. Additional variables with impact on the forecasts are the averaged *lvp* state of the airport (lvp_{ap}) and runway (lvp_{rw}) for bagging, and *cei* information at touchdown (cei_{td}), as well as the *lvp* state at touchdown 10 minutes before forecast initialization ($lvp_{\mathrm{td}_{-10}}$) for boosting. With growing lead times the strong dependence on *lvp* information at the touchdown point decreases. For the lead time +60 minutes lvp_{td} still has the highest impact on the forecasts. However, lvp_{rw} also has a strong influence in bagging at +60 minutes lead time. Further variables with influence are the visibility outside the airport area (vis_{ext}), lvp_{ap}, and the airport averaged *rvr* (rvr_{ap}). Their influence is stronger in boosting, yet. At the lead time +120 minutes lvp_{ap} has the strongest influence in bagging. Standard meteorological information of humidity (*rh*) and temperature (*tl*), as well as lvp_{td} and vis_{ext} have further influence. For boosting lvp_{ap}, rvr_{ap}, and vis_{ext} have the strongest impact. Additional impact has *lvp* information at touchdown and runway. Moreover, the conditional *lvp* 0 state climatology (lvp_{cl}) has influence on forecasts of both models at the lead time +120 minutes.

These results confirm the predictor selection analysis for the OLR model of Kneringer *et al.* (2017), which also show strongest impact of current *lvp* information at short lead times, and climatological impact at longer lead times. Altogether, the variables with the highest influence are similar for both models for each lead time. At short lead times point measurements at touchdown have the highest influence, whereas averaged airport information becomes more important with increasing lead time. In bagging the dependency on *lvp* information is

stronger, instead horizontal visibility information has higher impact in boosting.

5. Discussion and Conclusion

Nowcasts of *lvp* (low-visibility procedure) states at Vienna International Airport are produced in a high temporal resolution using different tree-based models. The various models are compared amongst each other and to the OLR (ordered logistic regression) model of Kneringer *et al.* (2017). Tree-based boosting leads to the best forecasts, outperforming bagging, random forests, and OLR – all three of which perform similarly – while single decision trees have the lowest predictive performance for all lead times.

To improve the forecast performance of *lvp* nowcasts we enlarge the predictor setup and increase the output resolution compared to the investigation of Kneringer *et al.* (2017). The models established on the enlarged predictor setup are compared to boosting trees with the standard predictor setup. Boosting performs again most accurate. At the shortest lead time boosting with the large predictor setup performs about equally to boosting with the standard setup. With longer lead times the benefit of models with the large predictor setup increases. Bagging also performs well, however, it is outperformed by boosting at each lead time. Random forests only perform well at longer lead times, while single decision trees perform well only at very short lead times and become badly with longer lead time.

The reason of this pattern is based on the different working processes of the models. At very short lead times the predictions strongly depend on information of the current *lvp* state at the touchdown point. Since this predictor often occurs as split variable in the trees all aggregated decision trees look similar. Hence, the performance of bagging and decision tree is similar at the shortest lead time. If most information for the forecasts is contained in only one predictor variable, random forest perform worse due to their random preselection of potential split variables. With longer lead times the strong dependency to only one predictor variable decreases and therefore the benefit of the ensemble-merging models increases strongly compared to single decision trees.

In summary, we have shown that tree-based methods are suitable tools for *lvp* state nowcasts. Their computational costs are comparable with linear regression methods and therefore they are able to produce instantaneous forecasts when new observations arrive. Especially the tree-based ensemble methods bagging and boosting perform well for *lvp* predictions. The predictors with the highest impact to *lvp* nowcasts contain information of the current *lvp* state and horizontal visibility at forecast initialization.

6. Acknowledgments

This study is supported by the Austrian Research Promotion agency (FFG), 843457. We thank Markus Kerschbaum, Andreas Lanzinger, and the staff at Vienna International Airport for fruitful information and discussions regarding these investigations. The authors also thank the Austro Control GmbH for providing access to the observation data.

References

Agresti A (2003). *Categorical Data Analysis*. John Wiley & Sons, Inc. doi:10.1002/0471249688.

Bartoková I, Bott A, Bartok J, Gera M (2015). "Fog Prediction for Road Traffic Safety in a Coastal Desert Region: Improvement of Nowcasting Skills by the Machine-Learning Approach." *Boundary-Layer Meteorology*, **157**(3), 501–516. doi:10.1007/s10546-015-0069-x.

Benjamin SG, Weygandt SS, Brown JM, Hu M, Alexander CR, Smirnova TG, Olson JB, James EP, Dowell DC, Grell GA, Lin H, Peckham SE, Smith TL, Moninger WR, Kenyon JS, Manikin GS (2016). "A North American Hourly Assimilation and Model Forecast Cycle: The Rapid Refresh." *Monthly Weather Review*, **144**(4), 1669–1694. doi:10.1175/MWR-D-15-0242.1.

Bott A, Trautmann T (2002). "PAFOG – A New Efficient Forecast Model of Radiation Fog and Low-Level Stratiform Clouds." *Atmospheric Research*, **64**(1–4), 191–203. doi:10.1016/S0169-8095(02)00091-1.

Boutle IA, Finnenkoetter A, Lock AP, Wells H (2016). "The London Model: Forecasting Fog at 333 m Resolution." *Quarterly Journal of the Royal Meteorological Society*, **142**(694), 360–371. doi:10.1002/qj.2656.

Breiman L, Friedman J, Stone CJ, Olshen RA (1984). *Classification and Regression Trees*. Wadsworth Statistics Series, CRC Press, Boca Raton, FL, USA.

Bühlmann P, Hothorn T (2007). "Boosting Algorithms: Regularization, Prediction and Model Fitting." *Statistical Science*, **22**(4), 477–505. doi:10.1214/07-STS242.

Bühlmann P, Yu B (2002). "Analyzing Bagging." *The Annals of Statistics*, **30**(4), 927–961. doi:10.1214/aos/1031689014.

Dutta D, Chaudhuri S (2015). "Nowcasting Visibility During Wintertime Fog over the Airport of a Metropolis of India: Decision Tree Algorithm and Artificial Neural Network Approach." *Natural Hazards*, **75**(2), 1349–1368. doi:10.1007/s11069-014-1388-9.

Epstein ES (1969). "A Scoring System for Probability Forecasts of Ranked Categories." *Journal of Applied Meteorology*, **8**, 985–987. doi:10.1175/1520-0450(1969)008<0985:ASSFPF>2.0.CO;2.

Ghirardelli JE, Glahn B (2010). "The Meteorological Development Laboratorys Aviation Weather Prediction System." *Weather and Forecasting*, **25**(4), 1027–1051. doi:10.1175/2010WAF2222312.1.

Glahn B, Schnapp AD, Ghirardelli JE, Im JS (2017). "A LAMP-HRRR MELD for Improved Aviation Guidance." *Weather and Forecasting*, **32**(2), 391–405. doi:10.1175/WAF-D-16-0127.1.

Herman GR, Schumacher RS (2016). "Using Reforecasts to Improve Forecasting of Fog and Visibility for Aviation." *Weather and Forecasting*, **31**(2), 467–482. doi:10.1175/WAF-D-15-0108.1.

Hothorn T, Buehlmann P, Kneib T, Schmid M, Hofner B (2017a). *mboost: Model-Based Boosting*. R package version 2.8-0, URL https://CRAN.R-project.org/package=mboost.

Hothorn T, Hornik K, Strobl C, Zeileis A (2017b). *party – A Laboratory for Recursive Partytioning*. R package version 1.2-3, URL http://CRAN.R-project.org/package=party.

Hothorn T, Hornik K, van de Wiel MA, Zeileis A (2006a). "A Lego System for Conditional Inference." *The American Statistician*, **60**(3), 257–263. doi:10.1198/000313006X118430.

Hothorn T, Hornik K, Zeileis A (2006b). "Unbiased Recursive Partitioning: A Conditional Inference Framework." *Journal of Computational and Graphical Statistics*, **15**(3), 651–674. doi:10.1198/106186006X133933.

Hothorn T, Lausen B, Benner A, Radespiel-Tröger M (2004). "Bagging Survival Trees." *Statistics in Medicine*, **23**(1), 77–91. doi:10.1002/sim.1593.

International Civil Aviation Organization (2005). "Manual of Runway Visual Range Observing and Reporting Practices." *Technical Report Doc 9365 AN/908*.

James G, Witten D, Hastie T, Tibshirani R (2014). *An Introduction to Statistical Learning: With Applications in R*. Springer Texts in Statistics, New York, NY, USA. ISBN 1461471370, 9781461471370.

Kneringer P, Dietz S, Mayr GJ, Zeileis A (2017). "Probabilistic Nowcasting of Low-Visibility Procedure States at Vienna International Airport During Cold Season." *Eeecon Working Paper. Faculty of Economics and Statistics, University of Innsbruck*. URL https://eeecon.uibk.ac.at/wopec2/repec/inn/wpaper/2017-21.pdf.

Leyton SM, Fritsch JM (2004). "The Impact of High-Frequency Surface Weather Observations on Short-Term Probabilistic Forecasts of Ceiling and Visibility." *Journal of Applied Meteorology*, **43**, 145–156. doi:10.1175/1520-0450(2004)043<0145:TIOHSW>2.0.CO;2.

Leyton SM, Fritsch M (2003). "Short-Term Probabilistic Forecasts of Ceiling and Visibility Utilizing High-Density Surface Weather Observations." *Weather and Forecasting*, **18**, 891–902. doi:10.1175/1520-0434(2003)018<0891:SPFOCA>2.0.CO;2.

Marzban C, Leyton S, Colman B (2007). "Ceiling and Visibility Forecasts via Neural Networks." *Weather and Forecasting*, **22**(3), 466–479. doi:10.1175/WAF994.1.

Murphy AH (1971). "A Note on the Ranked Probability Score." *Journal of Applied Meteorology*, **10**, 155–156. doi:10.1175/1520-0450(1971)010<0155:ANOTRP>2.0.CO;2.

Pasini A, Pelino V, Potesta S (2001). "A Neural Network Model for Visibility Nowcasting from Surface Observations: Results and Sensitivity to Physical Input Variables." *Journal of Geophysical Research: Atmospheres*, **106**(D14), 14951–14959. doi:10.1029/2001JD900134.

Quinlan JR (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Incorporated, San Francisco, CA, USA. ISBN 1-55860-238-0.

Roquelaure S, Tardif R, Remy S, Bergot T (2009). "Skill of a Ceiling and Visibility Local Ensemble Prediction System (LEPS) According to Fog-Type Prediction at Paris-Charles de Gaulle Airport." *Weather and Forecasting*, **24**(6), 1511–1523. doi:10.1175/2009WAF2222213.1.

Schmid M, Hothorn T, Maloney KO, Weller DE, Potapov S (2011). "Geoadditive Regression Modeling of Stream Biological Condition." *Environmental and Ecological Statistics*, **18**(4), 709–733. doi:10.1007/s10651-010-0158-4.

Strobl C, Malley J, Tutz G (2009). "An Introduction to Recursive Partitioning: Rationale, Application, and Characteristics of Classification and Regression Trees, Bagging, and Random Forests." *Psychological Methods*, **14**(4), 323–348. doi:10.1037/a0016973.

Vislocky RL, Fritsch MJ (1997). "An Automated, Observations-Based System for Short-Term Prediction of Ceiling and Visibility." *Weather and Forecasting*, **12**, 31–43. doi:10.1175/1520-0434(1997)012<0031:AAOBSF>2.0.CO;2.

Wilks D (2011). *Statistical Methods in the Atmospheric Sciences*. Academic Press. ISBN 9780123850225.

Affiliation:

Sebastian J. Dietz, Philipp Kneringer, Georg J. Mayr
Institute of Atmospheric and Cryospheric Sciences
Faculty of Geo- and Atmospheric Sciences
Universität Innsbruck
Innrain 52
6020 Innsbruck, Austria
E-mail: Sebastian.Dietz@uibk.ac.at, Philipp.Kneringer@uibk.ac.at, Georg.Mayr@uibk.ac.at

Achim Zeileis
Department of Statistics
Faculty of Economics and Statistics
Universität Innsbruck
Universitätsstraße 15
6020 Innsbruck, Austria
E-mail: Achim.Zeileis@uibk.ac.at

University of Innsbruck - Working Papers in Economics and Statistics
Recent Papers can be accessed on the following webpage:

http://eeecon.uibk.ac.at/wopec/

2017-22 **Sebastian J. Dietz, Philipp Kneringer, Georg J. Mayr, Achim Zeileis:** Forecasting low-visibility procedure states with tree-based statistical methods

2017-21 **Philipp Kneringer, Sebastian J. Dietz, Georg J. Mayr, Achim Zeileis:** Probabilistic nowcasting of low-visibility procedure states at Vienna International Airport during cold season

2017-20 **Loukas Balafoutas, Brent J. Davis, Matthias Sutter:** How uncertainty and ambiguity in tournaments affect gender differences in competitive behavior

2017-19 **Martin Geiger, Richard Hule:** The role of correlation in two-asset games: Some experimental evidence

2017-18 **Rudolf Kerschbamer, Daniel Neururer, Alexander Gruber:** Do the altruists lie less?

2017-17 **Meike Köhler, Nikolaus Umlauf, Sonja Greven:** Nonlinear association structures in flexible Bayesian additive joint models

2017-16 **Rudolf Kerschbamer, Daniel Muller:** Social preferences and political attitudes: An online experiment on a large heterogeneous sample

2017-15 **Kenneth Harttgen, Stefan Lang, Judith Santer, Johannes Seiler:** Modeling under-5 mortality through multilevel structured additive regression with varying coefficients for Asia and Sub-Saharan Africa

2017-14 **Christoph Eder, Martin Halla:** Economic origins of cultural norms: The case of animal husbandry and bastardy

2017-13 **Thomas Kneib, Nikolaus Umlauf:** A Primer on Bayesian Distributional Regression

2017-12 **Susanne Berger, Nathaniel Graham, Achim Zeileis:** Various Versatile Variances: An Object-Oriented Implementation of Clustered Covariances in R

2017-11 **Natalia Danzer, Martin Halla, Nicole Schneeweis, Martina Zweimüller:** Parental leave, (in)formal childcare and long-term child outcomes

2017-10 **Daniel Muller, Sander Renes:** Fairness views and political preferences - Evidence from a large online experiment

2017-09 **Andreas Exenberger:** The Logic of Inequality Extraction: An Application to Gini and Top Incomes Data

2017-08 **Sibylle Puntscher, Duc Tran Huy, Janette Walde, Ulrike Tappeiner, Gottfried Tappeiner:** The acceptance of a protected area and the benefits of sustainable tourism: In search of the weak link in their relationship

2017-07 **Helena Fornwagner:** Incentives to lose revisited: The NHL and its tournament incentives

2017-06 **Loukas Balafoutas, Simon Czermak, Marc Eulerich, Helena Fornwagner:** Incentives for dishonesty: An experimental study with internal auditors

2017-05 **Nikolaus Umlauf, Nadja Klein, Achim Zeileis:** BAMLSS: Bayesian additive models for location, scale and shape (and beyond)

2017-04 **Martin Halla, Susanne Pech, Martina Zweimüller:** The effect of statutory sick-pay on workers' labor supply and subsequent health

2017-03 **Franz Buscha, Daniel Müller, Lionel Page:** Can a common currency foster a shared social identity across different nations? The case of the Euro.

2017-02 **Daniel Müller:** The anatomy of distributional preferences with group identity

2017-01 **Wolfgang Frimmel, Martin Halla, Jörg Paetzold:** The intergenerational causal effect of tax evasion: Evidence from the commuter tax allowance in Austria

2016-33 **Alexander Razen, Stefan Lang, Judith Santer:** Estimation of spatially correlated random scaling factors based on Markov random field priors

2016-32 **Meike Köhler, Nikolaus Umlauf, Andreas Beyerlein, Christiane Winkler, Anette-Gabriele Ziegler, Sonja Greven:** Flexible Bayesian additive joint models with an application to type 1 diabetes research

2016-31 **Markus Dabernig, Georg J. Mayr, Jakob W. Messner, Achim Zeileis:** Simultaneous ensemble post-processing for multiple lead times with standardized anomalies

2016-30 **Alexander Razen, Stefan Lang:** Random scaling factors in Bayesian distributional regression models with an application to real estate data

2016-29 **Glenn Dutcher, Daniela Glätzle-Rützler, Dmitry Ryvkin:** Don't hate the player, hate the game: Uncovering the foundations of cheating in contests

2016-28 **Manuel Gebetsberger, Jakob W. Messner, Georg J. Mayr, Achim Zeileis:** Tricks for improving non-homogeneous regression for probabilistic precipitation forecasts: Perfect predictions, heavy tails, and link functions

2016-27 **Michael Razen, Matthias Stefan:** Greed: Taking a deadly sin to the lab

2016-26 **Florian Wickelmaier, Achim Zeileis:** Using recursive partitioning to account for parameter heterogeneity in multinomial processing tree models

2016-25 **Michel Philipp, Carolin Strobl, Jimmy de la Torre, Achim Zeileis:** On the estimation of standard errors in cognitive diagnosis models

2016-24 **Florian Lindner, Julia Rose:** No need for more time: Intertemporal allocation decisions under time pressure

2016-23 **Christoph Eder, Martin Halla:** The long-lasting shadow of the allied occupation of Austria on its spatial equilibrium

2016-22 **Christoph Eder:** Missing men: World War II casualties and structural change

2016-21 **Reto Stauffer, Jakob Messner, Georg J. Mayr, Nikolaus Umlauf, Achim Zeileis:** Ensemble post-processing of daily precipitation sums over complex terrain using censored high-resolution standardized anomalies *published in Monthly Weather Review*

2016-20 **Christina Bannier, Eberhard Feess, Natalie Packham, Markus Walzl:** Incentive schemes, private information and the double-edged role of competition for agents

2016-19 **Martin Geiger, Richard Hule:** Correlation and coordination risk

2016-18 **Yola Engler, Rudolf Kerschbamer, Lionel Page:** Why did he do that? Using counterfactuals to study the effect of intentions in extensive form games

2016-17 **Yola Engler, Rudolf Kerschbamer, Lionel Page:** Guilt-averse or reciprocal? Looking at behavioural motivations in the trust game

2016-16 **Esther Blanco, Tobias Haller, James M. Walker:** Provision of public goods: Unconditional and conditional donations from outsiders

2016-15 **Achim Zeileis, Christoph Leitner, Kurt Hornik:** Predictive bookmaker consensus model for the UEFA Euro 2016

2016-14 **Martin Halla, Harald Mayr, Gerald J. Pruckner, Pilar García-Gómez:** Cutting fertility? The effect of Cesarean deliveries on subsequent fertility and maternal labor supply

2016-13 **Wolfgang Frimmel, Martin Halla, Rudolf Winter-Ebmer:** How does parental divorce affect children's long-term outcomes?

2016-12 **Michael Kirchler, Stefan Palan:** Immaterial and monetary gifts in economic transactions. Evidence from the field

2016-11 **Michel Philipp, Achim Zeileis, Carolin Strobl:** A toolkit for stability assessment of tree-based learners

2016-10 **Loukas Balafoutas, Brent J. Davis, Matthias Sutter:** Affirmative action or just discrimination? A study on the endogenous emergence of quotas *published in Journal of Economic Behavior and Organization*

2016-09 **Loukas Balafoutas, Helena Fornwagner:** The limits of guilt

2016-08 **Markus Dabernig, Georg J. Mayr, Jakob W. Messner, Achim Zeileis:** Spatial ensemble post-processing with standardized anomalies

2016-07 **Reto Stauffer, Jakob W. Messner, Georg J. Mayr, Nikolaus Umlauf, Achim Zeileis:** Spatio-temporal precipitation climatology over complex terrain using a censored additive regression model

2016-06 **Michael Razen, Jürgen Huber, Michael Kirchler:** Cash inflow and trading horizon in asset markets

2016-05 **Ting Wang, Carolin Strobl, Achim Zeileis, Edgar C. Merkle:** Score-based tests of differential item functioning in the two-parameter model

2016-04 **Jakob W. Messner, Georg J. Mayr, Achim Zeileis:** Non-homogeneous boosting for predictor selection in ensemble post-processing

2016-03 **Dietmar Fehr, Matthias Sutter:** Gossip and the efficiency of interactions

2016-02 **Michael Kirchler, Florian Lindner, Utz Weitzel:** Rankings and risk-taking in the finance industry

2016-01 **Sibylle Puntscher, Janette Walde, Gottfried Tappeiner:** Do methodical traps lead to wrong development strategies for welfare? A multilevel approach considering heterogeneity across industrialized and developing countries

University of Innsbruck

Working Papers in Economics and Statistics

2017-22

Sebastian J. Dietz, Philipp Kneringer, Georg J. Mayr, Achim Zeileis

Forecasting low-visibility procedure states with tree-based statistical methods

Abstract
Low-visibility conditions at airports can lead to capacity reductions and therefore to delays or cancelations of arriving and departing flights. Accurate visibility forecasts are required to keep the airport capacity as high as possible. We generate probabilistic nowcasts of low-visibility procedure (lvp) states, which determine the reduction of the airport capacity due to low-visibility. The nowcasts are generated with tree-based statistical models based on highly-resolved meteorological observations at the airport. Short computation times of these models ensure the instantaneous generation of new predictions when new observations arrive. The tree-based ensemble method "boosting" provides the highest benefit in forecast performance. For lvp forecasts with lead times shorter than one hour variables with information of the current lvp state, ceiling, and horizontal visibility are most important. With longer lead times visibility information of the airport's vicinity, humidity, and climatology also becomes relevant.

ISSN 1993-4378 (Print)
ISSN 1993-6885 (Online)