Extremal Random Forests to model airport departure delays

Job Reijns

525325jr@eur.nl

A Bachelor Thesis presented for the degree of

Econometrics & Operations Research

under the supervision of Prof. Chen Zhou

and second assessor Venes Schmidt, A

ERASMUS UNIVERSITY ROTTERDAM

Erasmus School of Economics

03-07-2022

Abstract

The modern commercial air transport industry poses an extremely complex logistic and planning problem. As bottlenecks are bound to arise and problems tend to accumulate, departure delays happen frequently. This is a costly event for all stakeholders (airport, airline, and passenger) as delays of over 4 hours are usually accompanied with financial consequences in the form of a fine or compensation. Therefore, it is in their interest to be able to model and understand the occurrence and length of flight delays. In this paper, we propose to use the recently introduced Extremal Random Forest algorithm (Athey et al., 2019) for this purpose. We introduce the algorithm and show its potential in a simulation study where we compare it to alternative methods such as Generalized Random Forest and an Unconditional Generalized Pareto Distribution. When applied to a dataset on all departing flights from New York airports in 2013 (Wickham, 2021), plots of parameters and quantiles suggest ERF is adequately able to model extreme quantiles of the delays conditional on a predictor space. However, a cross-validation scheme aimed at quantifying its performance suggests ERF does not significantly reduce the prediction loss over the alternative methods.

Contents

1 Introduction

The commercial air transport industry has grown into one of the largest industries in the world. With around 100.000 flights departing around the globe every single day, scheduling these aircraft and their departure time poses an extremely hard challenge. The largest obstacle to tackle in this process is how to account for the stochastic nature of both departure- and flight times. Delays are an often occurring phenomenon on all airports and airlines around the world. As an example, almost 20% of flights in 2013 departing from one of the three New York airports had not left within 20 minutes of scheduled departure time. Every delay can have big consequences for all stakeholders (airport, airline and passenger) and often starts a ripple effect that impacts non-stakeholders as well (Zhou et al., 2022). The Federal Aviation Administration estimated the total costs of delays over 2019 to be 33 billion USD (Lukacs, 2020). As of today, many European airlines are required to reimburse or even compensate their passengers when they experience extreme delays (often 3+ hours). In the US, the Transportation Department can impose a fine on the airline of up to 27500 USD per passenger when a 3+ hour delay is found to be their fault. Airports may also get heavily involved in these claims (Darroch, 2022). These are costly endeavours that should be avoided as much as possible. For operational and financial purposes, it is thus be helpful to provide an indication of risk of (extreme) delay for a certain flight. Therefore, this thesis will focus on the financial consequences of delays and aims to model the extreme delays with an Extremal Random Forest (ERF) model. The results of this approach should help airlines or airports make decisions on flight schedules and their Terms and Services.

The main question we intend to answer in this research is: "Does an Extremal Random Forest model significantly improve the estimation of extreme quantiles in flight delays?"

2 Literature

Our research will adapt the model proposed by Gnecco et al. (2022), who propose the ERF model for extreme quantile analysis that is able to handle large predictor spaces and highly nonlinear covariate structures as opposed to classical methods, that often break down for such scenarios. Their model, which is a combination of random forests and extrapolation, is able to extrapolate outside of the data range so it does not need many

extreme observations to train on. In greater detail, they estimate a quantile at an intermediate level with a classical quantile regression technique. They assume all observations that exceed this intermediate quantile follow a Generalized Pareto Distribution (GPD) and use that to extrapolate outside of the training data. The parameters of the GPD are estimated by minimizing a weighted negative log-likelihood function with the weights determined by a General Random Forest (GRF) with quantile loss. The GRF, a less restrictive application of a random forest (RF), is proposed by Athey et al. (2019) and allows for custom loss functions and splitting rules. This offers more flexibility than a regular RF, which assigns weights based on minimizing the Mean Squared Error (MSE) of the expected value. Gnecco et al. (2022) prove their ERF estimator is consistent under some assumptions. They show by means of a simulation study and an application on real data (US wages) that their model outperforms existing methods both in terms of accuracy and robustness.

Zhou et al. (2022) creates a propagation relation network between different airports and researches the effect of a delay in 1 airport on the delays in other airports. They propose that long delays in large airports can be spread out to smaller neighbouring airports. Apart from serving as motivation for our research, the research of Zhou et al. (2022) also allows for a useful implementation of ERF. This is because, in the process of delay propagation, a decision has to be made over which airport(s) the delay is spread out. In this decision process, ERF provides the useful ability to take in to account predictors such as airport, distance and weather to give an indication on the maximum length of potential further delays given a specific set of predictors.

Manley (2008) analyses so called Ground Delay Programs (GDP). These are scheduling tools that hold a departing plane on the ground as long as there is no empty gate at the arrival airport at the expected arrival time. This is done to avoid flights having to perform airborn holding patterns which are environmentally and financially damaging. The standard procedure for a GDP is "first-scheduled, first served" as this is the fairest among different airlines. However, Manley (2008) prove that other rationing rules, such as rules based on the number of passengers, benefit both passengers and airlines. We suggest an ERF approach benefits the GDP decision-making process. This is because our model can give an indication on the maximum delay of a certain flight, given that it is already significantly delayed. This might help in modeling the releases of empty gates,

which in turn is useful for imposing GDP's.

3 Data

To answer our research question, we will use the nycflights13 dataset by Wickham (2021). This dataset includes data on all flights that departed (to destinations in the US, Puerto Rico, and the American Virgin Islands) from a New York airport (JFK,EWR,LGA) in 2013. 336766 flights departed in 2013 from which we have reliable actual departure times of 328521 flights. The variable to be used as response variable in the ERF is the departure delay, denoted as `dep_delay` in the dataset. The summary statistics of all delays are shown in Table 1, note that a negative delay means the flight got airborne before its scheduled departure time. As we are interested in the extreme delays, a histogram of delays of more than 240 minutes is shown in Figure 1. The number of flights with a delay exceeding 4 hours is 9779, which is 2.9% of the dataset. The dataset also includes many contextual variables, of which a number will be used as covariates/predictors in the ERF. Examples include the time of departure, the operating airline, the type of aircraft, the destination and many weather related variables such as temperature, windspeed and visibility. The weather related variables are available at an hourly frequency and are fused with the nycflights13 dataset based on the (rounded) hour of departure. Due to computational limitations, a subset of 100.000 observations is selected before implementing the extreme quantile estimation methods. To reduce variability in the predictor space, we select the flights operated by a carrier that operates at least 10.000 flights a year from New York. From the 6 major carriers that remain, 100.000 flights are randomly selected.

	Count	**Mean**	**Median**	**Std**	**Min**	**Max**	**Missing Values**
Delays	336766	12.64	-2.00	40.21	-43.00	1301.00	8255

Table 1: Summary Statistics of all delays.

4 Methodology

We follow the steps from Gnecco et al. (2022) in setting up an Extremal Random Forest. We consider a response variable $Y \in \mathbb{R}$ and are interested in its distribution and tails

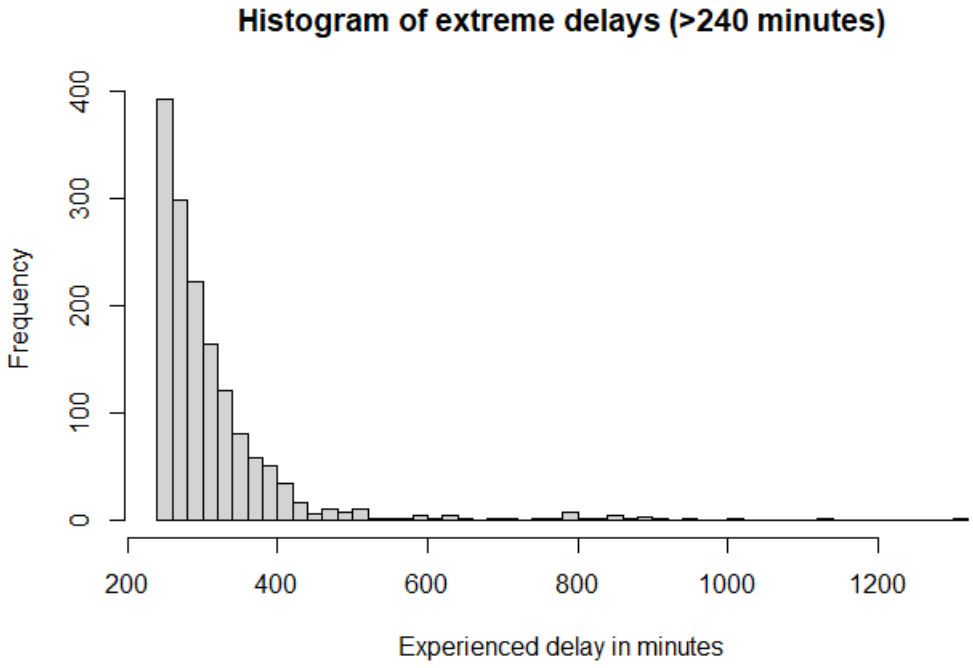

Figure 1: Histogram of all observed delays over 240 minutes.

given a set of covariates $X \in \mathbb{R}^p$. Thus, we define $Q_x(\tau)$ as the quantile at $\tau \in (0,1)$ of the conditional distribution of $Y|X = x$, so $Q_x(\tau) = F_x^{-1}(\tau)$ with F_x the CDF of Y conditional on $X = x$. Note that we aim to get an accurate estimate of $Q_x(\tau)$ for τ very close to 1 and that $X = x$ may not be close to our training data, especially for high dimensionality p. Gnecco et al. (2022) imposes an intermediate quantile at $\tau_0 < \tau$ that can be estimated with classical quantile regressions and aims to extrapolate to a more extreme quantile at τ using a Generalized Pareto Distribution (GPD). The approximation using the GPD is shown in equation (1)

$$Q_x(\tau) \approx Q_x(\tau_0) + \frac{\sigma(x)}{\xi(x)} \left[\left(\frac{1-\tau}{1-\tau_0} \right)^{-\xi(x)} - 1 \right], \tag{1}$$

where $\sigma(x) > 0$ and $\xi(x) \in \mathbb{R}$ are the conditional scale and shape parameters of the GPD respectively. Classical quantile regression techniques use the fact that $Q_x(\tau)$ minimizes the expectation of the quantile loss function $\rho_\tau(c) = c(\tau - \mathbb{1}\{c < 0\}), c \in \mathbb{R}$, as in equation (2)

$$Q_x(\tau) = \arg\min_{q \in \mathbb{R}} \mathbb{E}\left[\rho_\tau(Y - q) \mid X = x\right] \tag{2}$$

As in most practical cases $x \notin X$, the expectation can not be evaluated at sample level and thus Gnecco et al. (2022) propose an estimator, as shown in equation (3).

$$\hat{Q}_x(\tau) = \arg\min_{q \in \mathbb{R}} \sum_{i=1}^{n} w_n(x, X_i)\, \rho_\tau(Y_i - q) \tag{3}$$

where weights $w_n(x, X_i)$ are similarity weights that localize x within observed covariates X. Athey et al. (2019) propose to set $w_n(x, X_i)$ equal to the weights obtained from a Generalized Random Forest (GRF), since these are able to model large predictor spaces and complex quantile surfaces. A GRF is an ensemble method that fits B decision trees on some data using a custom split procedure. We explore a GRF trained on n independent copies of random vector (X, Y), with a splitting procedure based on quantile loss $\rho_\tau(c)$. After growing the trees, we define $L_b(x) \subset \mathbb{R}^P$ as the rectangular region or "leaf" that x belongs to in the b'th tree. For each tree, Athey et al. (2019) now assign equal weights to the observations that fall in the same leaf as x. As these sum to 1 they can be written as $w_{n,b}(x, X_i) := \mathbb{1}\{X_i \in L_b(x)\} / |\{j : X_j \in L_b(x)\}|$. The complete forest now considers the average weights over all grown trees, that is $w_n(x, X_i) = B^{-1} \sum_{b=1}^{B} w_{n,b}(x, X_i)$, to plug in equation (3) and obtain an estimate of $\hat{Q}_x(\tau_0)$. We can now define the exceedances $Z_i = \left(Y_i - \hat{Q}_{X_i}(\tau_0)\right)_+, i = 1, \ldots, n$. These exceedances are assumed to follow a GPD and we estimate its parameters $\theta(x) = (\sigma(x), \xi(x))$ by a maximum-likelihood approach. The contribution of the i'th exceedance Z_i to the negative log-likelihood is shown in equation (4).

$$\ell_\theta(Z_i) = \log\sigma + \left(1 + \frac{1}{\xi}\right) \log\left(1 + \frac{\xi}{\sigma} Z_i\right), \quad \theta \in (0, \infty) \times \mathbb{R}, \tag{4}$$

if $Z_i > 0$ and zero otherwise. Gnecco et al. (2022) now define a weighted negative log-likelihood that uses the weights $w_n(x, X_i)$ from a GRF, not necessarily the same as in the intermediate quantile procedure, to assign a measure of importance to the likelihood of each exceedance Z_i as shown in equation (5):

$$L_n(\theta; x) = \sum_{i=1}^{n} w_n(x, X_i)\, \ell_\theta(Z_i). \tag{5}$$

It now follows that

$$\hat{\theta}(x) = \arg\min_{\theta} L_n(\theta; x).$$

Plugging $\hat{\theta}(x)$ in equation (1) returns an estimate of the extreme quantile at τ. Gnecco et al. (2022) prove consistency under some conditions. We integrate the equations and

estimators above in one algorithm. The pseudocode of the complete algorithm can be found below:

Algorithm 1 Estimate quantile $\hat{Q}_x(\tau)$ for $\tau \approx 1$ with an Extremal Random Forest

Denote the training data by $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^n$, the test covariate with $x \in \mathbb{R}^p$ and determine extreme and intermediate quantile levels τ and τ_0 such that $\tau_0 < \tau$. Denote α as a vector of hyperparameters for the splitting rules in the GRF.

1: **procedure** FIT_ERF($\mathcal{D}, \tau_0, \alpha$)

2: $\quad \hat{Q}_\cdot(\tau_0), w_n(\cdot,\cdot), \leftarrow \text{GRF}(\mathcal{D}, \alpha)$

3: $\quad$ **return** erf $\leftarrow \left[\mathcal{D}, \hat{Q}_\cdot(\tau_0), w_n(\cdot,\cdot)\right]$

4: **procedure** PREDICT_ERF(ERF, x, τ)

5: $\quad Z_i \leftarrow \left(Y_i - \hat{Q}_{X_i}(\tau_0)\right)$

6: $\quad \hat{\theta}(x) \leftarrow \arg\min_\theta L_n(\theta; x)$ as in equation (5)

7: $\quad$ **return** $\hat{Q}_x(\tau)$ as in equation (1)

Note that in line 2, we can use any quantile regression approach (or GRF with different hyperparameters) to estimate $\hat{Q}_\cdot(\tau_0)$ instead of GRF.

5 Simulation study

We implement a study using simulated datasets to analyse the performance of the ERF algorithm in 1 and compare it to less recent extreme quantile estimation methods, QRF, GRF and Uncomditional GPD .

5.1 Data Generating Process (DGP)

Our DGP is fundamentally the same as in Gnecco et al. (2022). That is, we consider a set of predictors X sampled from a uniform distribution on the cube $[-1, 1]^p$. Response variable Y is defined by its conditional distribution $Y|X = x \sim s(x)T_4$ where $s(x) = 1 + \mathbb{1}\{x_1 > 0\}$ and T_ν is a random variable that follows a Student's t-distribution with $\nu > 0$ degrees of freedom. This implies that only x_1 has true predictive power for Y and $x_2, \ldots, x_p$ are noise variables which will be useful for comparing performance between different models. This comparison will be made based on the Mean Integrated Squared

Error (MISE), shown in equation (6).

$$\text{MISE} = \frac{1}{S}\frac{1}{n'}\sum_{i=1}^{n'}\left(\hat{Q}_{x_i}(\tau) - Q_{x_i}(\tau)\right)^2 \tag{6}$$

where S is the number of simulations, n' is the number of observations of x_i and between the brackets are the estimated and true extreme quantiles respectively.

5.2 Hyperparameter tuning

5.2.1 Minimum node size κ

The generalized random forest(s) embedded within algorithm 1 offer quite some freedom in choice of hyperparameters such as number of trees, bootstrapping parameters and the stopping treshold for splitting individual trees (minimum node size). We implement a cross-validation scheme, similar to Gnecco et al. (2022), to make an informed decision. Quantile loss is no reliable metric for analysing performance since there are likely few or no observations for $\tau \to 1$. Instead, we use the assumption that exceedances $Z_i = \left(Y_i - \hat{Q}_{X_i}(\tau_0)\right)_+$ follow a GPD to define the scoring function as the deviance of the GPD with estimated parameters. Say we are interested in a set of J potential values for tuning parameter α: $\alpha_1, \ldots, \alpha_J$. Formally, we randomly split our data in M equally sized folds $\mathcal{N}_1, \ldots, \mathcal{N}_M$ and then fit an ERF on the set $(X_i, Y_i), i \notin \mathcal{N}_M$ for each fold m and α_j. For each fitted ERF, we use the validation set $(X_i, Y_i), i \in \mathcal{N}_M$ to estimate $\hat{\theta}(X_i; \alpha_j)$. The cross-validation error can now be defined as

$$CV(\alpha_j) = \sum_{m=1}^{M}\sum_{i \in \mathcal{N}_m} \ell_{\hat{\theta}(X_i;\alpha_j)}(Z_i)\, \mathbb{1}\{Z_i > 0\} \tag{7}$$

where $\ell_\theta(Z_i)$ is as in equation (4) and maps Z_i to the deviance of the GPD. The optimal α^* is simply chosen such that $CV(\alpha^*) < CV(\alpha_j), j = 1, \ldots, J$. Gnecco et al. (2022) argue that minimum node size $\kappa \in \mathbb{N}$ is the most influential in the ERF algorithm. κ controls the complexity of a forest. As κ gets smaller, the leafs of trees will contain fewer observations. This directly influences the corresponding similarity weights $w_n(\cdot, \cdot)$ as these observations get assigned a relatively heavier weight. This can make the weights, which are crucial to ERF, sensitive and unstable. For large κ we observe the opposite, the ERF is not able to assign strong enough similarity weights. Therefore the choice of a balanced κ is vital and thus is the first parameter we optimize through cross validation.

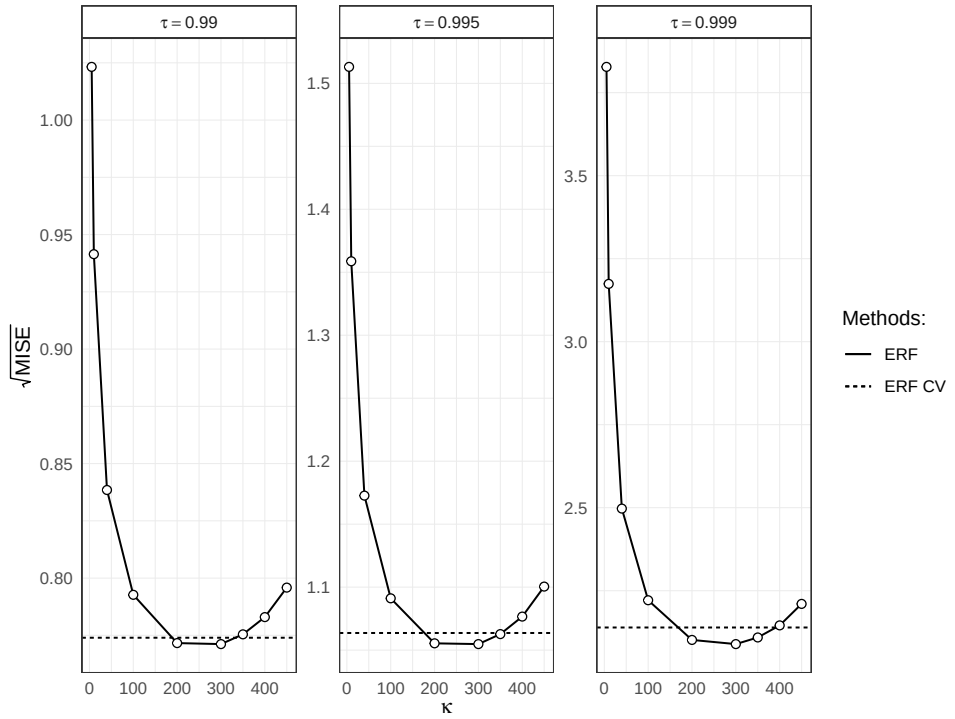

Figure 2: Performance of cross-validated ERF (dotted line) compared to different values of κ for regular ERF (solid line). See 5.1 for the DGP.

Figure 2 displays the results of our cross-validation process compared to regular ERF for multiple κ. The data was generated according to 5.1 with a dimensionality of $p = 40$. We implement $M = 5$-fold cross-validation which is repeated 3 times to decrease variability. Each forest grows 50 trees and the MISE is calculated over $S = 20$ simulations. Note that the MISE of the cross-validated ERF is close to the minimum value of MISE. This suggests that the cross-validation scheme works reasonably well. Therefore, we set $\kappa = 250$ to save future computational load. Important to note is that this cross validation scheme, when sequentially applied on multiple hyperparameters, does not necessarily return optimal parameters. This is due to the one dimensional approach of the optimization problem. However, due to computational limitations, Gnecco et al. (2022) consider this the best feasible option.

5.2.2 Scale and shape parameters

Equation (1) shows the highly non-linear influence of ξ on the estimation of extreme quantiles. Its value is crucial because, ξ, interpreted as the shape parameter of the GPD dictates the assumed behaviour in the tails of the noise distributions. Gnecco

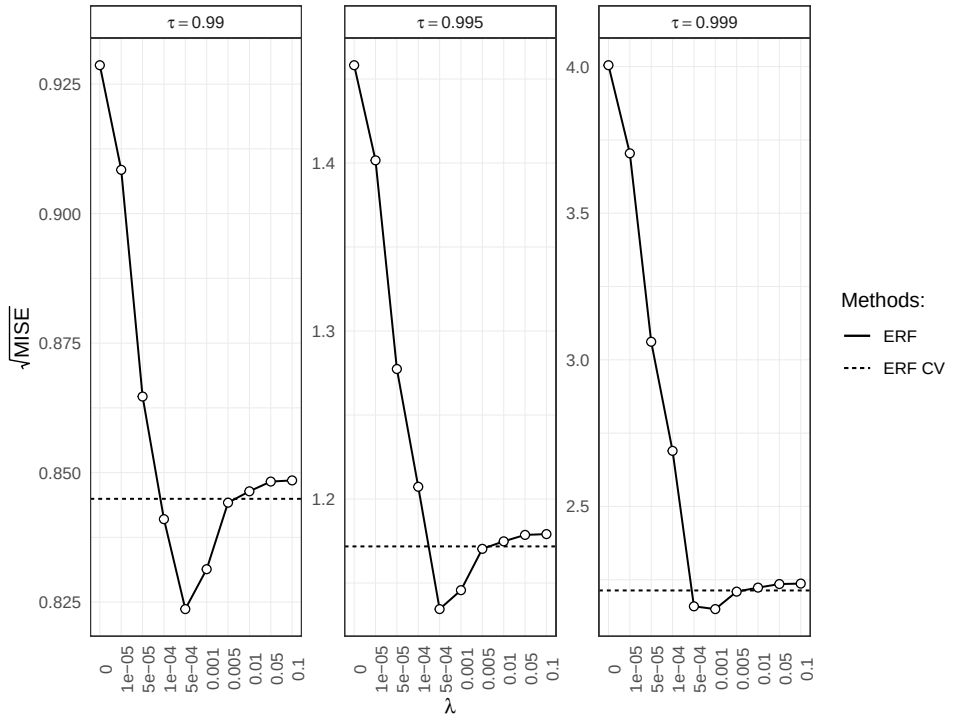

Figure 3: Performance of cross-validated ERF (dotted line) compared to different values of λ for regular ERF (solid line). See 5.1 for the DGP.

et al. (2022) propose a penalization scheme based on the weighted GPD deviance plus the squared distance between ξ and a constant (unconditional) ξ, generally given by empirical experience. They derive the following estimator for $\theta(x) = (\xi, \sigma)$:

$$\hat{\theta}(x) = \underset{\theta=(\sigma,\xi)}{\arg\min} \frac{1}{(1-\tau_0)} L_n(\theta; x) + \lambda \left(\xi - \xi_0\right)^2 \tag{8}$$

where $\lambda \geq 0$ is a tuning parameter. For $\lambda \to \infty$, ξ will be a constant equal to ξ_0. However, we can create a more complex model when λ is small as this allows for a varying ξ over predictor space $\mathcal{X}$. Further motivation of this penalization method is beyond the scope of this paper. We apply a similar cross-validation scheme (with the same hyperparameters) on λ as we did in 5.2.1, the results of which are shown in Figure 3. In this application ξ_0 is chosen to be the estimated unconditional GPD shape parameter. The minimum node size is 250, as determined in 5.2.1. Similar to 5.2.1, we conclude that the cross-validation is adequately able to find a value close to the optimal value.

5.3 Model performance comparison

This section explores the performance of ERF in comparison to other extreme quantile estimation methods, specifically in the context of $\tau \to 1$, high predictor dimensionality p and robustness to tail heaviness of the noise distributions. The models we compare ERF with include Meinshausen (2006)'s quantile regression forest (QRF), Athey et al. (2019)'s general random forest (GRF) and an unconditional GPD approach. These methods are chosen as they are fundamental to the ERF algorithm and it would therefore be interesting to see if a combination of the 3, along with the added computational intensity, yields significantly better performance.

5.3.1 Experiment 1

In this experiment, we generate a dataset with size $n = 2000$, again according to the DGP in 5.1, and go through several values for both τ and p. The results of all 4 models are displayed in Figure 4 where we use the MISE again to make a comparison. In the left subfigure, we keep $p = 10$ constant and incrementally increase τ. Note that all methods are close for intermediate quantile $\tau_0 = 0.8$. At this point ERF and GRF are equal since the intermediate quantile is calculated by a GRF. However, when looking at higher quantile levels, we see the power of extrapolation as the MISE of the ERF method is lower than all other methods for $\tau > 0.99$. The choice of a (conditional) GPD to model the tails of our response variable is also supported as the unconditional GPD is able to model extreme quantiles better than the forest based methods.

The right subfigure sets $\tau = 0.9995$ and evaluates performance for increasing values of dimensionality (and noise) p. Note that although all methods seem relatively robust to increasing dimension p, none of the methods are competitive in MISE compared with the ERF algorithm. These findings are similar in nature to the findings of Gnecco et al. (2022).

5.3.2 Experiment 2

This experiment analyses the robustness to different tail heavy noise distributions in a large dimension. We again simulate data according to 5.1 with $p = 40$ and the noise distributions are chosen to have shape parameters $\xi = 0, \frac{1}{4}, \frac{1}{3}$. Note that $\xi = 0$ corresponds to a Gaussian distribution and $\xi = \frac{1}{\nu}$ corresponds to a Student's t distribution with ν

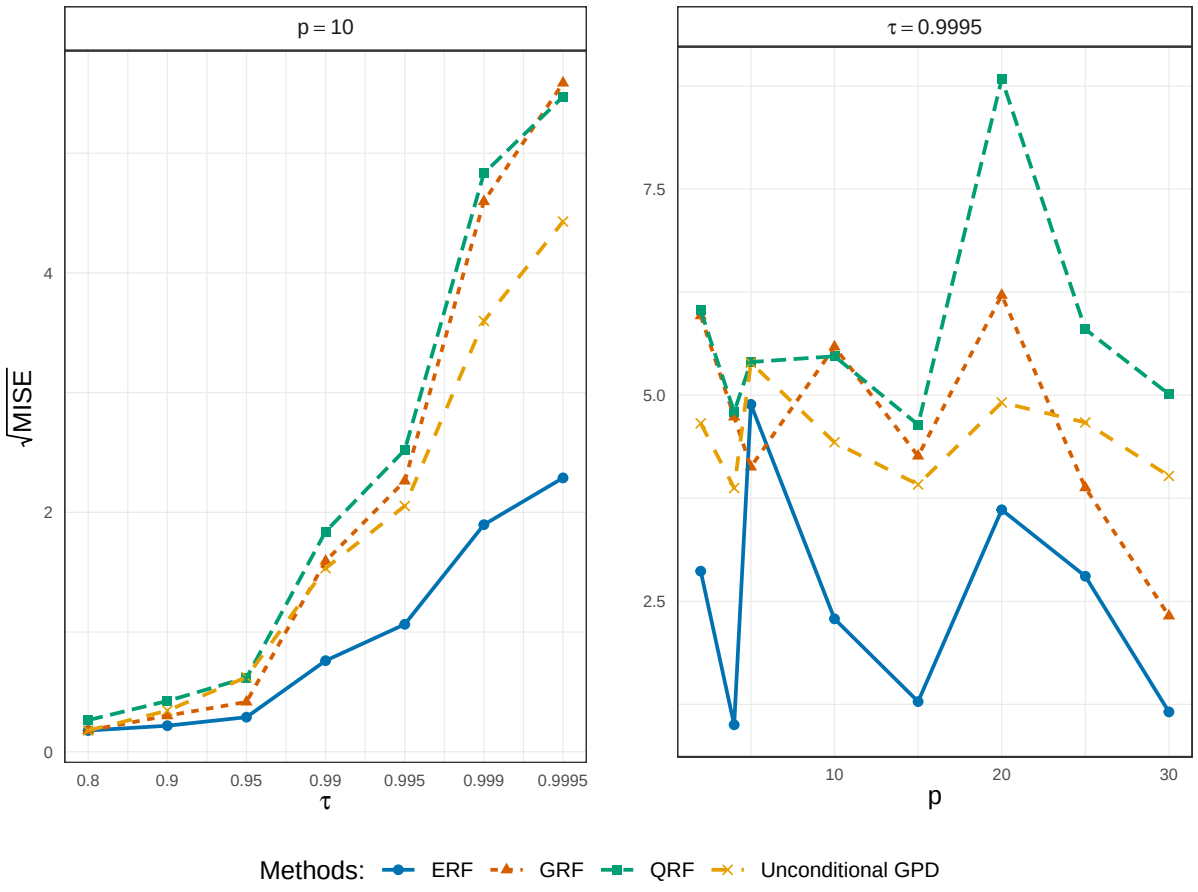

Figure 4: Mean Integrated Squared error as a function of τ (left) and p (right) for all four methods. Data Generating Process as in 5.1.

degrees of freedom. We fix extreme quantile $\tau = 0.9995$, Figure 5 shows the results for all methods and values of ξ, note that the figure displays ISE and the triangles correspond to the MISE. Logically, as the tails of the noise get heavier, the problem gets more difficult and thus all methods perform worse. The main result following from Figure 5 is the outperformance of the ERF algorithm, with the lowest MISE for all noise distributions. Furthermore, note the extreme outliers, respresented as dots, that progressively become larger in the QRF and GRF methods, highlighting the importance of extrapolation methods for extreme quantile regressions.

6 Analysis of flight delays

We now apply the ERF algorithm to our dataset of flight delays in 2013 as elaborated on in 3. We compare the results against the GRF and Unconditional GPD methods

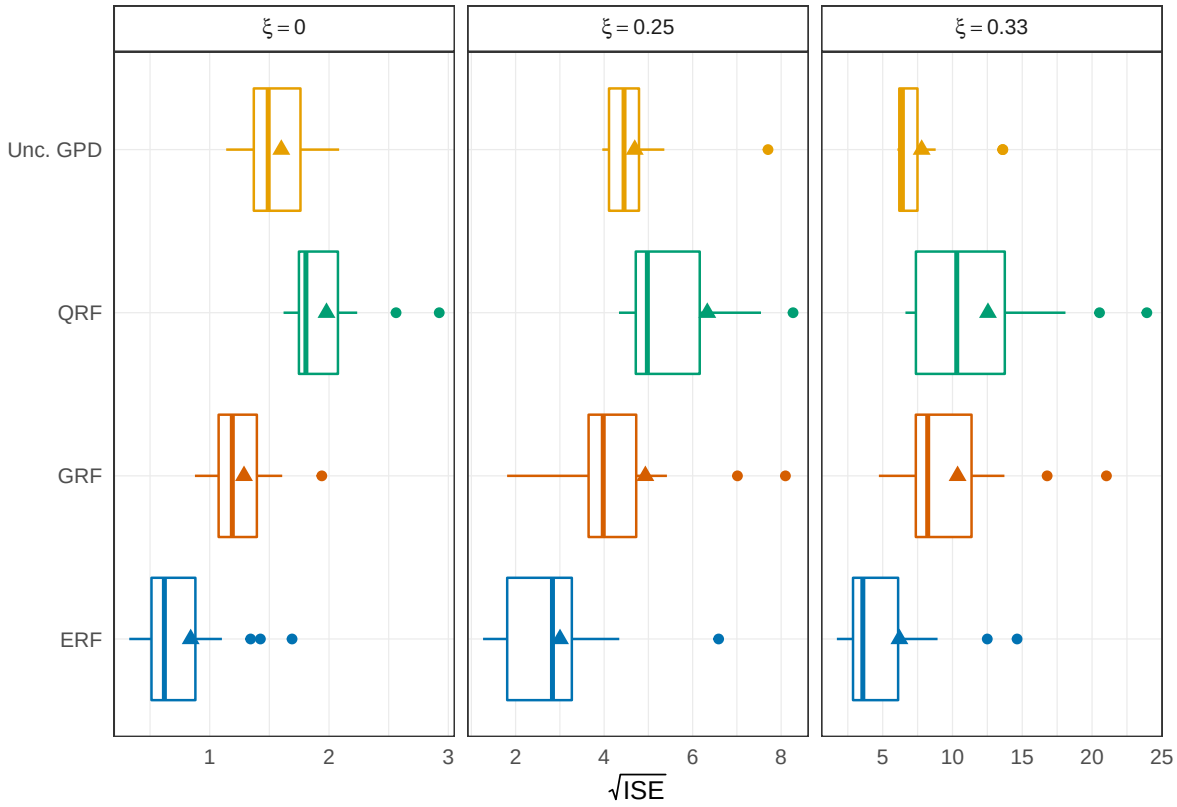

Figure 5: Integrated errors of the four methods for $\tau = 0.9995$ and varying ξ. Note that the triangles denote the MISE.

introduced in 5.3. To ease computational load and improve visualisation, we select the 9 carriers that performed at least 10.000 flights and apply our methods on a randomly sampled subset of 100.000 observations. This number of observations is plentiful for the algorithms to be able to provide their extreme quantile estimate. Furthermore, under the assumption that the results of the simulation study are valid for our dataset, we do not implement another cross-validation scheme for the minimum node size and the penalizing factor. We set $\kappa = 250$ and $\lambda = 4e^{-5}$. Similar to Gnecco et al. (2022), we split our dataset in a 50/50 training and validation set and use 10% of the training set to fit the ERF and 90% to estimate the GPD parameters $\hat{\theta} = \left(\hat{\xi}, \hat{\sigma}\right)$. Figure 6 displays the estimated GPD parameters for different hours of departure. The values of the estimated shape parameter ξ range between 0.2 and 0.3, indicating heavy tails conditional on the predictor space, which supports our implementation of ERF. Note that there is an increasing relation between the scale parameter $\hat{\sigma}$ and the specific hour and a decreasing relation for the shape parameter $\hat{\xi}$. This inverse relation is typical behaviour between these 2 parameters as a larger (smaller) ξ must be compensated to retain a valid probability density function. This

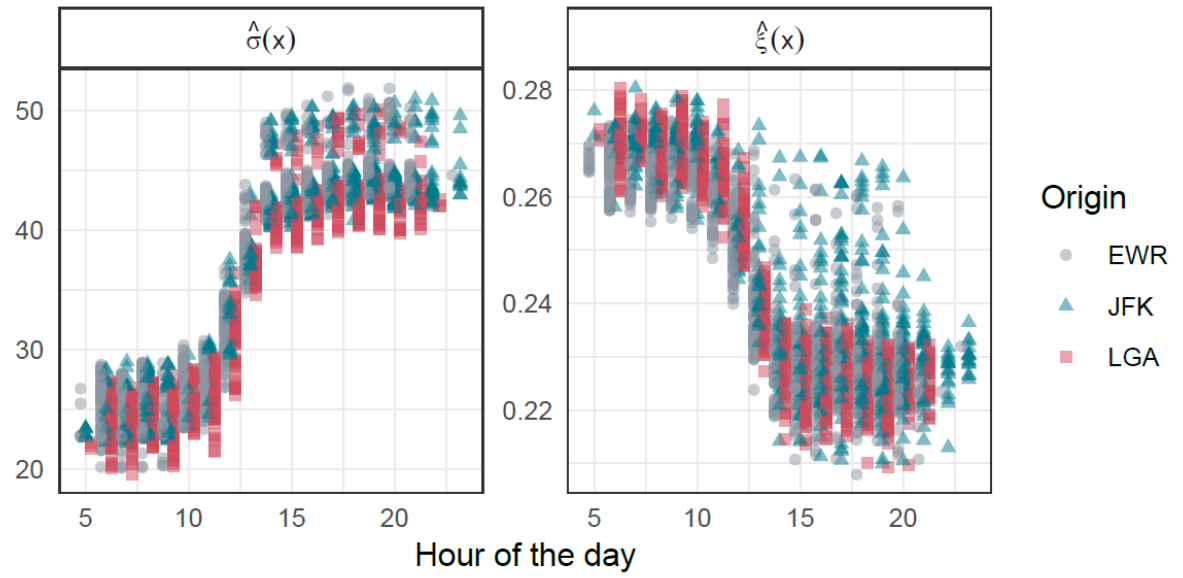

Figure 6: Estimated GPD parameters for different hours of departure. Hours were obtained by rounding the scheduled departure times down to full hours.

is usually achieved by decreasing (increasing) σ. The jump in both parameter estimates around 11am is of great interest and will be elaborated on later. Lastly, there is a clear distinction between the airports concerning the shape parameter estimates. La Guardia Airport (LGA) shows a clear decreasing relation while this relation is more distorted for the other two airports. A reasonable explanation follows from the fact that almost all flights departing LGA are domestic. This means that there is no extensive customs procedure during check-in and in general less bottlenecks that can create (unexpected) delays. This is a possible argument for why the tails of the delay distribution conditional on LGA as origin are less heavy and better defined than for EWR and JFK.

Figure 7 shows the predicted quantiles of the ERF, GRF and Unconditional GDP methods for different τ as a function of the predictor "hour of the day". Note that all methods clearly indicate an increasing relationship, the later the hour of departure, the higher the estimated quantile. For ERF and GRF (at $\tau = 0.9$), Figure 7 shows a noticeable jump around 11am, similar to what the scale and (negative) shape parameter estimates display in Figure 7. A possible explanation for this is the fact that the New York Port Authority imposes a (voluntary) curfew from 12pm - 6am on the NYC airports to reduce noise and allow for maintenance on the airport/runways. Therefore, flights leaving in the early hours of the day, are likely to have stayed the night at the airport and are thus

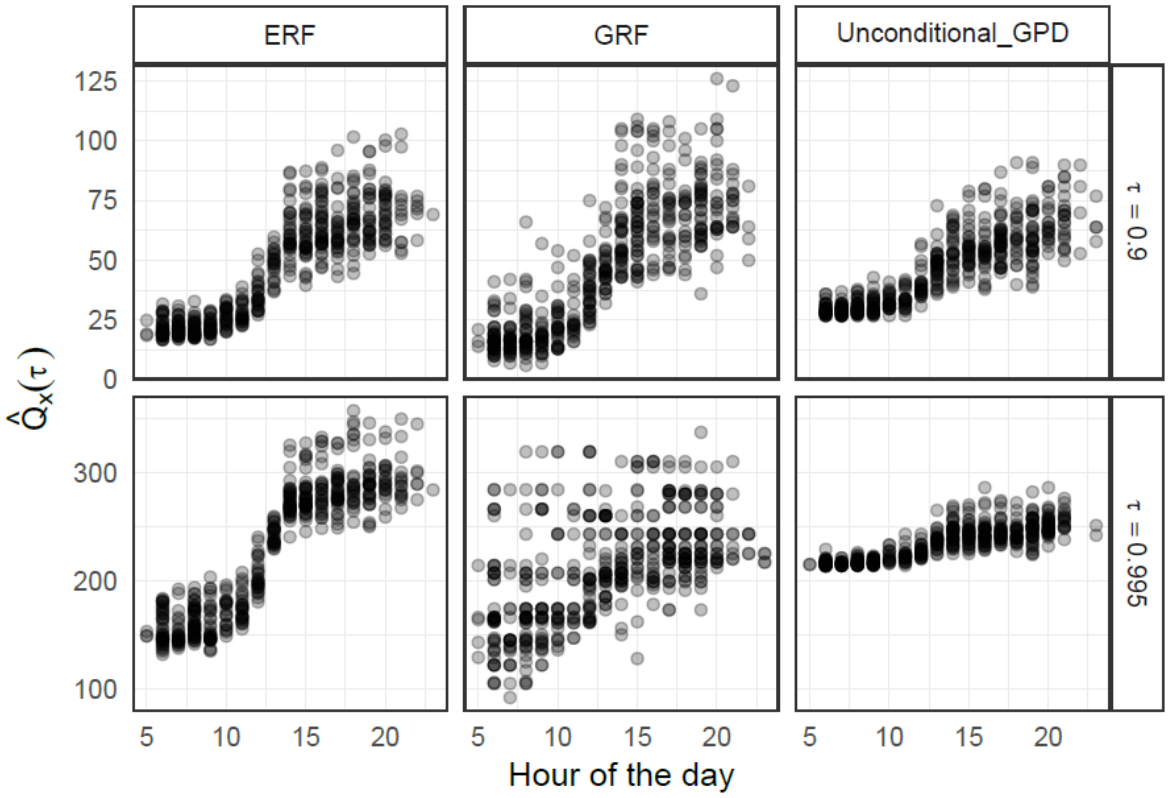

Figure 7: Predicted quantiles compared between ERF, GRF and Unconditional GPD for different τ.

"ready to go". After these early hours, airplanes often have to arrive from another airport, increasing the possibility of (accumulated) delay. Note that the ERF quantiles retain their shape even for extreme quantiles while the GRF and Unconditional GDP methods, due to their lack of extrapolation, lose most of their shape and therefore interpretability. Another reason is that the two latter methods are not able to re-estimate scale and shape parameter conditional on the predictor space while Figure 6 clearly indicates the necessity for this ability. Other predictors, such as the operating carrier of a flight, did not show a clear and interpretable relation between parameter estimates and predictor. 2 examples are included in Appendix A. We continue this research focusing on the specific "hour of the day" predictor.

To formally asses the performance of the applied methods, we use the same metric as Gnecco et al. (2022):

$$\mathcal{R}_n(\hat{Q}_\cdot(\tau)) := \frac{\sum_{i=1}^{w} \mathbb{1}\left\{Y_i < \hat{Q}_{X_i}(\tau)\right\} - w\tau}{\sqrt{w\tau(1-\tau)}} \tag{9}$$

where w is the number of observations in the validation set. $\mathcal{R}_n(\hat{Q}_\cdot(\tau))$ compares the empirical proportion of observations $Y_i < \hat{Q}_{X_i}(\tau)$ with the theoretically expected proportion τ. Gnecco et al. (2022) reasons that by the Central Limit Theorem, $\hat{Q}_{X_i}(\tau)$ is asymptotically $N(0,1)$. We implement an atypical cross-validation scheme that creates 10 folds and uses 1 fold to fit and 9 methods to validate on. This is necessary to ensure we have enough observations to check performance on extreme quantiles. Figure 8 shows the performance over these 10 repetitions. The grey area indicates the 95% confidence interval of the absolute value of a standard Gaussian, which is assumed to be consistent with the 95% confidence interval of $R_n(\hat{Q}_\cdot(\tau))$ as $n \to \infty$. Note that the outperformance of ERF is not as evident as in Figure 5. A possible explanation is the choice of ξ and κ, these were assumed to be generalizable from the simulation study to the plane delay dataset. As the nature of the predictor space is very different, uniformly sampled versus real-life variables such as wind-speed and month, this is a possible explanation for the poorer performance of EFR. This suggestion should be further examined in further research.

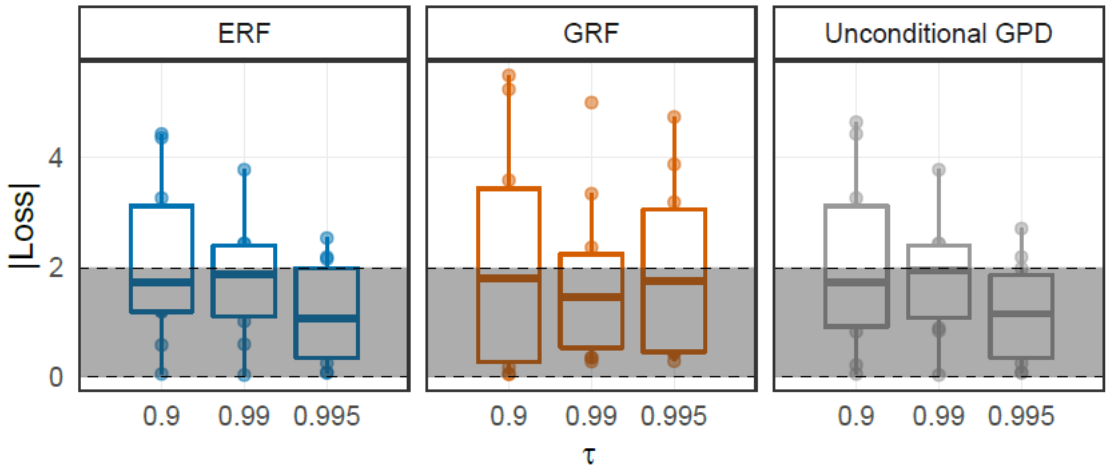

Figure 8: Absolute loss as defined in 9 for the different methods where the grey area represents the 95% interval of $|N(0,1)|$.

7 Conclusion

To answer our research question: "Does an Extremal Random Forest model significantly improve the estimation of extreme quantiles in flight delays?", we implemented the ERF method in a simulation study to evaluate its performance against the more conventional methods, Generalized Random Forests, Quantile Regression Forests and an Unconditional Generalized Pareto Distribution fit. This simulation study highlighted the potential of the ERF method as it proved to be suitable for cross-validation schemes to determine its parameters and showed lower Mean Integrated Square Errors than the alternative methods. Furthermore, the ERF method showcased its robustness to large predictor spaces and varying hyperparameters, which is a desirable property. When applied to our dataset on plane delays for flights departing New York in 2013, the estimates of the GPD parameters as a function of the predictors indicated that such parameters should not be constant. Additionally, the estimated extreme quantiles of the three methods suggested that extrapolation is necessary for a method to retain its shape in extreme contexts that occur outside of the training sample's range. Both these characteristics theoretically suit the ERF algorithm more than its alternatives. However, the last cross-validation scheme, aimed at quantifying performance on our own dataset, indicated that the performance of ERF was not evidently better than the aforementioned alternative methods. The proposed cause of the under-performance relates back to the chosen hyperparameters which were based on the simulation study for computational convenience. As our predictor space is

vastly different in nature to the simulation study, the choice of hyperparameters may be the limiting factor in the performance of ERF, further research into this is needed. So, to answer our research question: No, although the ERF method displays a high potential and desirable properties, it does not improve the estimation of extreme quantiles in flight delays enough to justify its higher computational load over methods such as GRF and Unc. GPD.

8 Discussion

This paper is not short of shortcomings. The main limitation of our research was the computational load of the ERF algorithm and especially the cross-validation schemes. Therefore some decisions were made based on heuristics or assumptions. We would define the choice of hyperparameters κ and λ as the main potential issue of the results in this paper. These were assumed to be equal to the simulation study to relieve computational load, while the nature of the covariates was not similar. The ERF algorithm has shown its potential in the simulation study but, possibly due to these choices, was not able to replicate this performance on our dataset. Therefore, further research is needed to identify the consequences of this assumed generalizability of simulation results.

References

Athey, S., J. Tibshirani, and S. Wager (2019). Generalized random forests. *Annals of Statistics vol. 47*, p. 1179–1203.

Darroch, G. (04-05-2022). Duizenden schadeclaims van reizigers om chaos op schiphol. *Het Parool*.

Gnecco, N., E. Terefe, and S. Engelke (2022). Extremal random forests. https://arxiv.org/pdf/2201.12865.pdf.

Lukacs, M. (2020). Cost of delay estimates. *Federal Aviation Authority (FAA)*.

Manley, B. (2008). Minimizing the pain in air transportation: Analysis of performance and equity in ground delay programs. *PH. D. thesis at George Mason University*.

Meinshausen, N. (2006). Quantile regression forests. *Journal of Machine Learning Research 7*, 983–999.

Wickham, H. (2021). Airline on-time data for all flights departing new york city in 2013.

Zhou, F., G. Jiang, Z. Lu, and Q. Wang (2022). Evaluation and analysis of the impact of airport delays. *Scientific Programming vol. 2022*.

A Further results

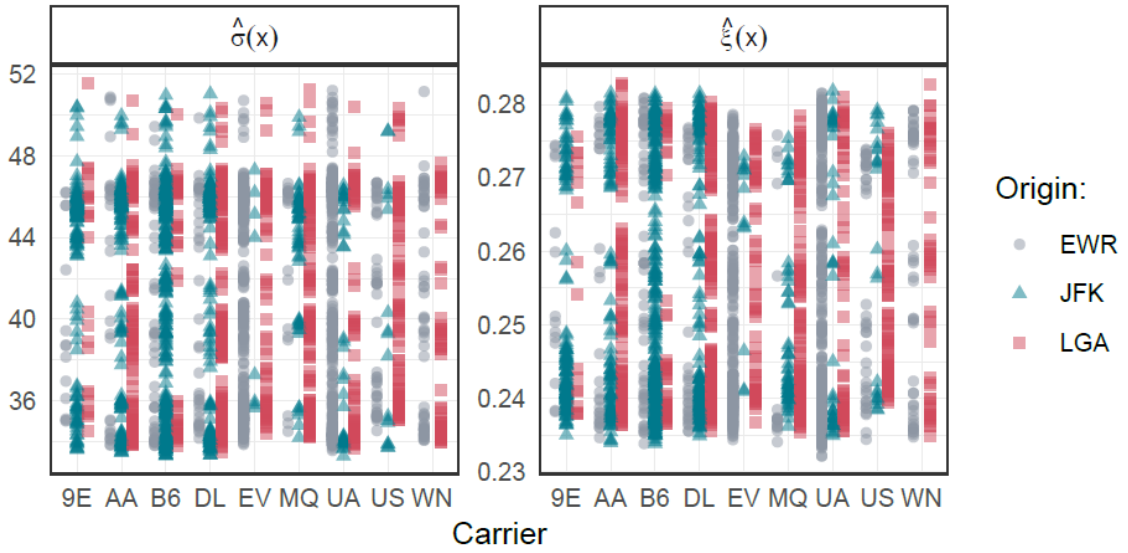

Figure 9: Estimated GPD parameters for different carriers.

B Notes on code

This research has used the ERF package provided by Gnecco et al. (2022) extensively. This package can be found and downloaded at https://github.com/nicolagnecco/erf. Furthermore, for the analysis of plane delays, the package containing all helper functions of Gnecco et al. (2022) was used. This package can be found at https://github.com/nicolagnecco/erf-numerical-results. Regarding our own code, this consisted primarily of tuning the code from Gnecco et al. (2022) to fit our data and intentions. There were also quite some bugs to be fixed within the helper functions. The main files I used for the analysis of plane delays:

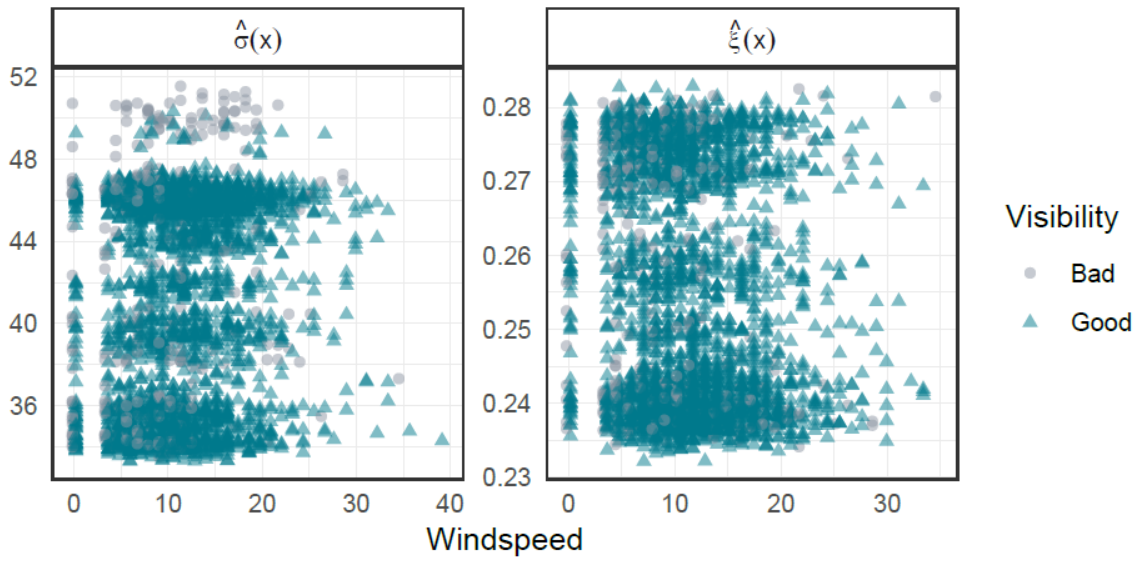

Figure 10: Estimated GPD parameters for different windspeeds (mph). The visibility is on a scale of 1-10 and is classified good if ≥ 8. Note that visibility < 8 is a rare event.

- `dependencies.R` installs all necessary packages for the ERF algorithm.
- `prepare_flight_data.R`, a file that loads the dataset, selects all relevant variables, merges the weather data into it and defines random subset of a certain size.
- `flight_analysis.R`, this file prepares some constants and feeds the our own data in both the ERF and CV schemes of Gnecco et al. (2022). Results are stored on the C: drive
- `plot_flights.R`, which loads the results stored by `flights_analysis.R` and plots all relevant results in a clear way.

Note that the programming efforts of my research thus mainly consisted of understanding the codes provided by Gnecco et al. (2022) and ensuring it worked reliably for my dataset. Presumably due to the young age of the ERF algorithm, the code contained many small bugs that needed debugging. This has taken a considerable amount of our available time as well.