1 Conformalized Quantile Regression

This chapter introduces *Conformalized Quantile Regression (CQR)* as the first of two main post-processing procedures that we implemented in the `postforecasts` package.

Section 1.1 explains the original Conformalized Quantile Regression algorithm as proposed in Romano, Patterson, and Candès (2019). There, we highlight potential limitations of the traditional implementation that could potentially be fixed by more flexible modifications, which are discussed in Section 1.2 and Section 1.3.

1.1 Traditional CQR

All derivations in this sections are taken from the original paper (Romano, Patterson, and Candès 2019). The authors motivate Conformalized Quantile Regression by stating two criteria that an ideal procedure for generating prediction intervals should satisfy:

- It should provide valid coverage in finite samples without making strong distributional assumptions
- The resulting intervals should be as short as possible at each point in the input space

According to the authors, CQR performs well on both criteria while being *distribution-free* and adaptive to *heteroscedasticity*.

1.1.1 Statistical Validity

The algorithm that CQR is build upon is statistically supported by the following Theorem:

Theorem 1.1. *If $(X_i, Y_i), i = 1, \dots, n+1$ are exchangeable, then the $(1-\alpha)\cdot 100\%$ prediction interval $C(X_{n+1})$ constructed by the CQR algorithm satisfies*

$$P\left(Y_{n+1} \in C(X_{n+1})\right) \geq 1 - \alpha.$$

Moreover, if the conformity scores E_i are almost surely distinct, then the prediction interval is nearly perfectly calibrated:

$$P\left(Y_{n+1} \in C(X_{n+1})\right) \leq 1 - \alpha + \frac{1}{|I_2| + 1},$$

where I_2 denotes the calibration set.

Thus, the first statement of Theorem 1.1 provides a coverage *guarantee* in the sense that the adjusted prediction interval is *lower-bounded* by the desired coverage level. The second statement adds an *upper-bound* to the coverage probability which gets tighter with increasing sample size and asymptotically converges to the desired coverage level $1-\alpha$ such that lower bound and upper bound are asymptotically identical.

1.1.2 Algorithm

The CQR algorithm is best described as a multi-step procedure.

Step 1:
Split the data into a training and validation (here called *calibration*) set, indexed by I_1 and I_2, respectively.

Step 2:
For a given quantile α and a given quantile regression algorithm $\mathcal{A}$, calculate lower and upper interval bounds on the training set:

$$\{\hat{q}_{\alpha,low},\ \hat{q}_{\alpha,high}\} \leftarrow \mathcal{A}\left(\{(X_i, Y_i) : i \in I_1\}\right).$$

Step 3:
Compute *conformity scores* on the calibration set:

$$E_i := \max\left\{\hat{q}_{\alpha,low}(X_i) - Y_i,\ Y_i - \hat{q}_{\alpha,high}(X_i)\right\} \quad \forall\, i \in I_2$$

For each i, the corresponding score E_i is *positive* if Y_i is *outside* the interval $[\hat{q}_{\alpha,low}(X_i),\ \hat{q}_{\alpha,high}(X_i)]$ and *negative* if Y_i is *inside* the interval.

Step 4:
Compute the *margin* $Q_{1-\alpha}(E, I_2)$ given by the $(1-\alpha)(1+\frac{1}{1+|I_2|})$-th empirical quantile of the scores E_i in the calibration set. For small sample sizes and small quantiles α the quantile above can be greater than 1 in which case it is simply set to 1 such that the maximum value of the score vector is selected.

Step 5:
On the basis of the original prediction interval bounds $\hat{q}_{\alpha,low}(X_i)$ and $\hat{q}_{\alpha,high}(X_i)$, the new *post-processed* prediction interval for Y_i is given by

$$C(X_{n+1}) = [\hat{q}_{\alpha,low}(X_i) - Q_{1-\alpha}(E, I_2),\ \hat{q}_{\alpha,high}(X_i) + Q_{1-\alpha}(E, I_2)]\,.$$

Note that the *same* margin $Q_{1-\alpha}(E, I_2)$ is subtracted from the original lower quantile prediction and added to the original upper quantile prediction. This limitation is adressed in Section 1.2.

1.1.3 Results

We first analyze the Effect of CQR in the UK Covid-19 Forecasting Challenge data and point out common trends of CQR adjustments. Then, we investigate if these trends generalize to the larger European Forecast Hub data set.

The first finding that is valid for almost all feature combinations is that CQR *expands* the original forecast intervals. Due to **step 5** of Section 1.1.2 it moves lower and upper bound in a *symmetric* way by using the same margin for lower and upper corrections. This implies that the interval *midpoint* remains unchanged when applying the traditional CQR algorithm.

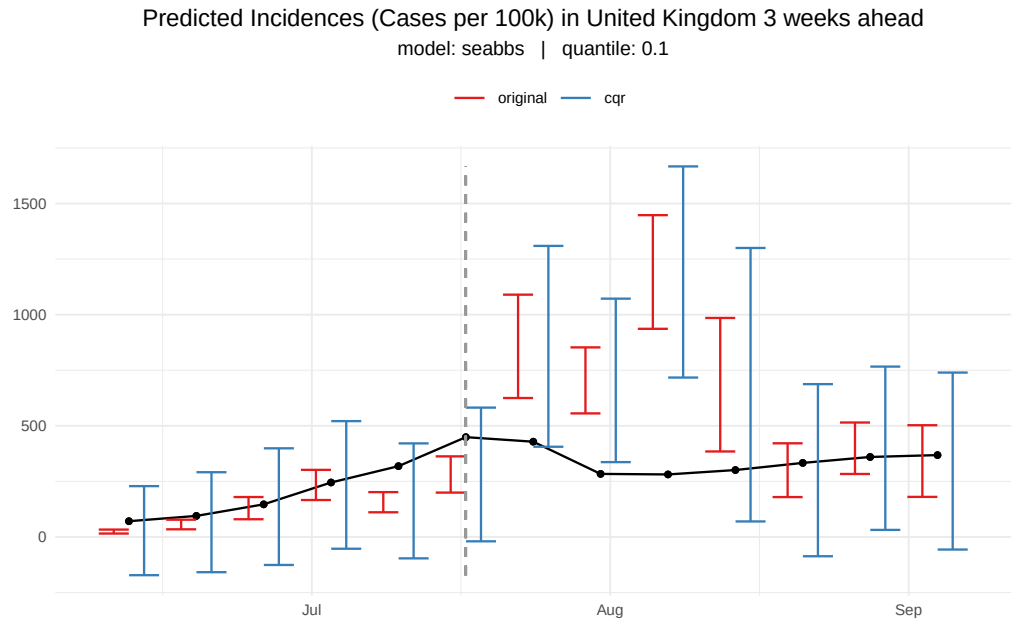

Figure 1: CQR tends to make prediction intervals larger, here for the 'seabbs' forecasting model

One extreme example of this behaviour is given in Figure 1. Since the `seabbs` forecasts are submitted by single individual, we find evidence for the hypothesis stated in **??** that humans tend to be too confident in their own predictions resulting in too narrow uncertainty bounds. By extending the intervals symmetrically CQR maintains *pointwise* information from the original forecasts while simultaneously increasing interval coverage.

Table 1: WIS improvement by CQR for one particular feature combination on the validation set.

method	model	target_type	horizon	quantile	interval_score	dispersion
cqr	seabbs	Cases	3	0.1	157.32	87.49
original	seabbs	Cases	3	0.1	210.62	38.14

Table 2: Overall WIS improvement by CQR on the validation set.

method	interval_score	dispersion
cqr	62.15	24.1
original	65.74	12.0

Yet, the pure effect of increasing coverage does not automatically imply that the Weighted Interval Score has improved as well due to the trade-off between coverage and precision. Thus, we explicitly compute the WIS, once for the specific covariate combination of Figure 1 in Table 1 and once aggregated over all *models*, *target types*, *horizons* and *quantiles* in Table 2.

Both tables confirm the visual impression of Figure 1: CQR improves the Weighted Interval Score by increasing the `dispersion` value, a measure for the prediction interval width. This effect is particularly strong in case if the `seabbs` model but still applies to a more moderate extent to most of the other forecasting models. Since many of the general findings for traditional CQR coincide between the UK data and the EU Forecast Hub data, we jump straight to the latter.

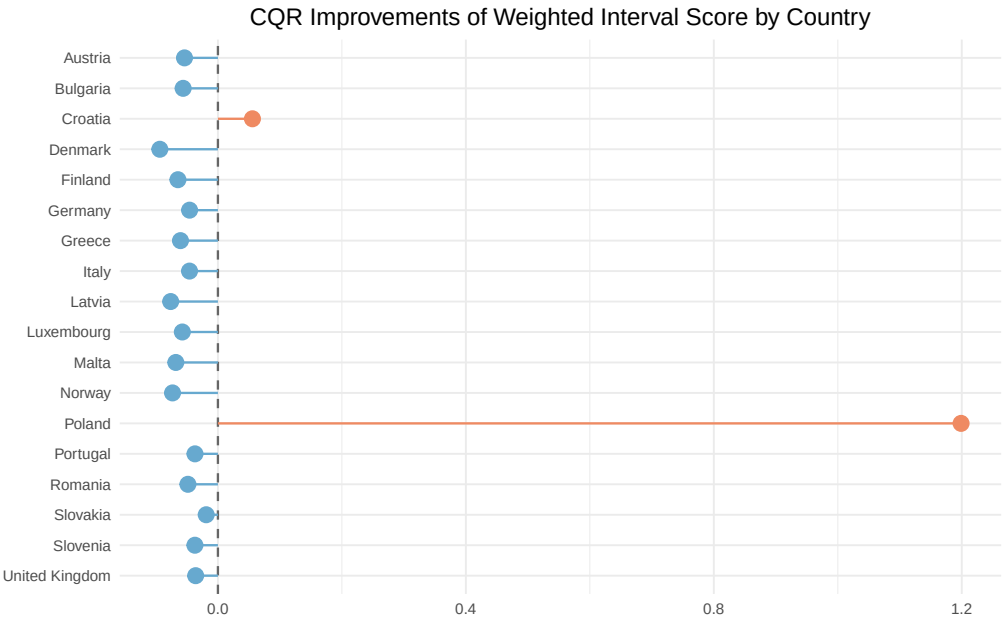

Figure 2: CQR proves to be beneficial for the vast majority of countries, with the major exception of Poland

First, we investigate if CQR is equally effective across all countries. Figure 2 indicates that this is clearly *not* the case: CQR is beneficial on out of sample data in almost all 18 selected countries. The strongest effect in absolute value, however, can be observed for Poland in *negative* direction.

At first sight this finding seems like a data entry error, there is no obvious reason why such a general algorithm like Conformalized Quantile Regression might not work for one specific location. The large negative effect is also interesting in light of Theorem 1.1: We *know* that CQR always improves the forecast intervals on

Table 3: Weighted Interval Score for Poland by Model on Training and Validation Set

method	model	training score	validation score
cqr	epiforecasts-EpiNow2	22.84	1.94
original	epiforecasts-EpiNow2	23.71	1.32
cqr	EuroCOVIDhub-ensemble	29.44	2.34
original	EuroCOVIDhub-ensemble	31.37	0.96
cqr	IEM_Health-CovidProject	56.54	4.68
original	IEM_Health-CovidProject	62.04	0.91

the training set which, in particular, applies to Poland as well. We can confirm this theoretical guarantee empirically by evaluating the Weighted Interval Score for Poland on the training set only. Table 3 summarizes the training and validation scores for three selected forecasting models.

Indeed, CQR improves the WIS for all three models in-sample whereas the performance out-of-sample drops drastically. This finding provides evidence that the observations used for the initial training phase must be *fundamentally different* to those encountered in the Time Series Cross Validation process. More specifically it suggests a *distribution shift* of the true observed values and/or the original quantile predictions right at the split of training and validation phase. Further, the *scale* of training and validation scores is quite different, which usually stems from different magnitudes of the observed incidences in absolute terms.

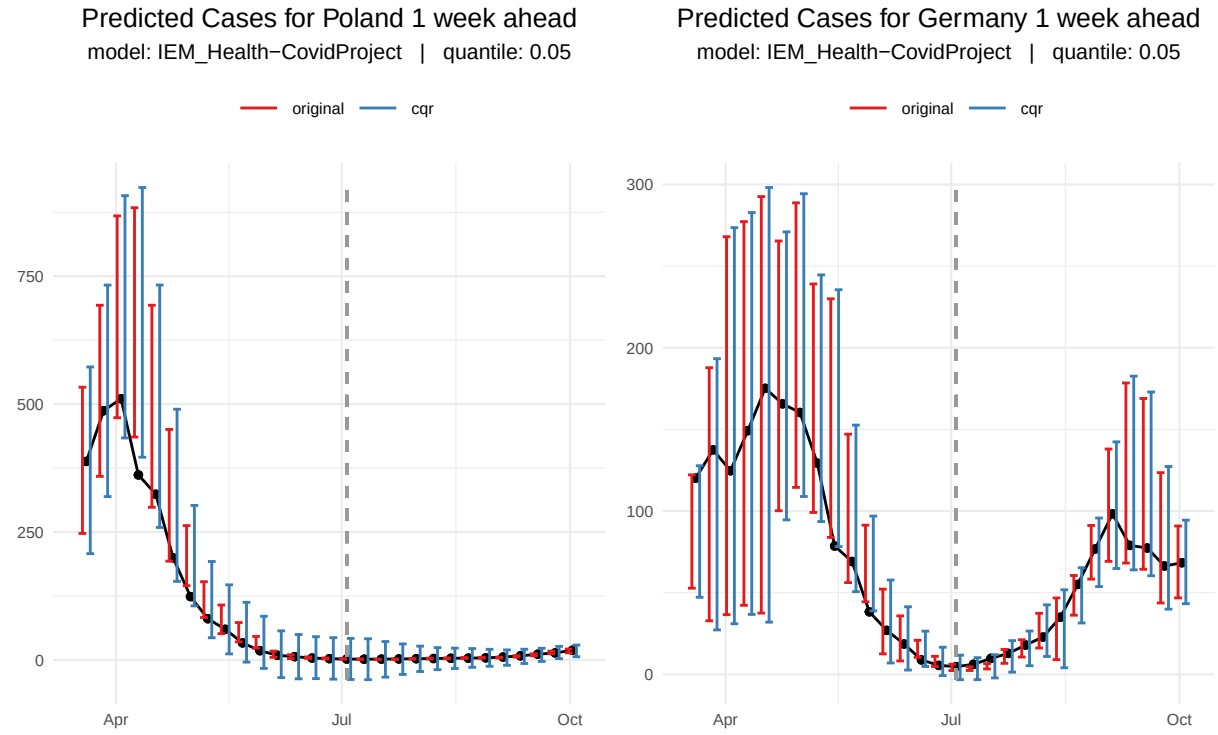

Figure 3: Development of Covid-19 Cases in 2021 in Poland and Germany

Figure 3 confirms our hypothesis for 1 week-ahead forecasts of 90% prediction intervals for Covid-19 Cases. The left plot displays the development of observed and predicted values for the outlier Poland compared to the same setting for Germany where CQR is beneficial. A few weeks before the training-validation split which is highlighted by the grey dashed line, the Case incidences plummeted in Poland. In strong contrast to Germany, where the Covid-19 situation relaxed during the summer months of 2021 as well, the incidences remain low until late autumn in Poland (according to the collected data of the European Forecast Hub),

which also explains the different scales of the interval scores in Table 3.

These consistently low observed values are connected to low uncertainty of the original forecasts displayed in Figure 3 that were submitted only one week in advance. The forecasters were well aware of the current situation and could quickly react with lower point forecasts and narrower prediction intervals. CQR is not able to compete with this flexibility and requires a long time span to adapt to irregular behaviour. The reasons for these slow adjustments, which reveal a major downside of CQR, follow immediately from the underlying algorithm and are explained in detail in Section 1.2.3.

At the end of this section, we briefly summarize the performance of vanilla CQR across different *quantiles*, *target types* and *horizons*. To obtain more informative visual illustrations we exclude Poland from the further analysis.

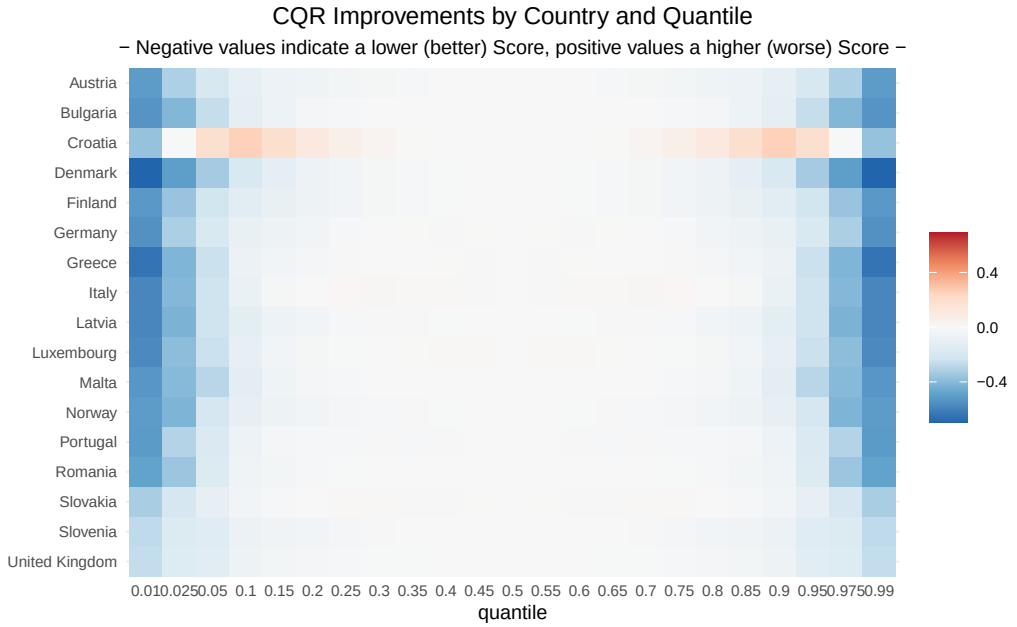

Figure 4: CQR is most beneficial for extreme Quantiles

Figure 4 shows the performance of CQR for all 23 quantile levels in the data set. Although the effect size varies by country, the general trend holds unanimously: Extreme quantiles in the tails of the predictive distribution benefit most from Post-Processing with a gradual decline towards centered quantiles. The same trend can be observed to an even larger extent for non-expert forecasts in the UK Covid-19 Forecasting Challenge Data.

Similar to quantiles there exist obvious tendencies for different forecast horizons as well. Figure 5 shows the performance of CQR across horizons, again stratified by country. Although the effects are more diverse compared to Figure 4, CQR generally works better for larger forecast horizons. Exceptions of this rule are Croatia, which is the only country besides Poland with a negative effect of CQR, and Malta, where the trend is actually reversed and CQR corrections are most beneficial for short-term forecasts.

Both of the previous figures suggest that Post-Processing with Conformalized Quantile Regression is worthwhile whenever the uncertainty is comparably high, which is the case for both quantiles in the tails and large forecast horizons.

Lastly, Table 4 aggregates Weighted Interval Scores on the validation set by target type. Interestingly, the observed effects disagree for the first time: While forecasts for Covid-19 Cases benefit significantly, forecasts for Deaths become slightly worse through CQR adjustments.

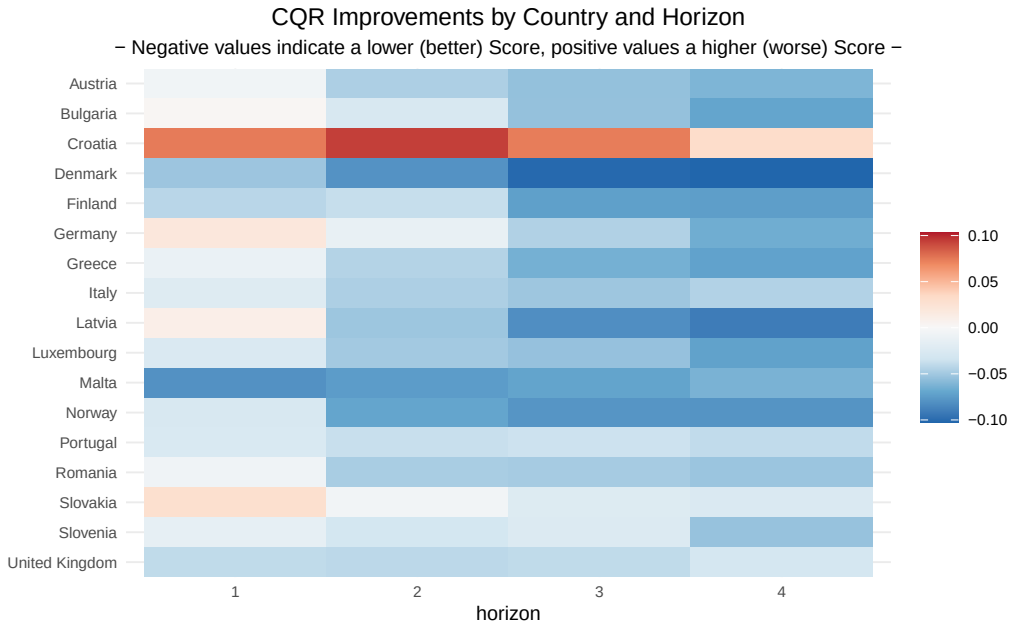

Figure 5: CQR is most beneficial for large Forecast Horizons

Table 4: CQR Improvements by Target Type for European Forecast Hub Data excluding Poland

method	target_type	interval_score	dispersion
cqr	Cases	59.26	19.55
original	Cases	62.69	15.49
cqr	Deaths	0.32	0.15
original	Deaths	0.32	0.13

1.2 Asymmetric CQR

1.2.1 Theory

As noted in Section 1.1 this section suggests a first extension to the original CQR algorithm. Instead of limiting ourselves to choose the *same* margin $Q_{1-\alpha}(E, I_2)$ for adjusting the original lower and upper quantile predictions, we allow for individual and, thus, generally different margins $Q_{1-\alpha,low}(E, I_2)$ and $Q_{1-\alpha,high}(E, I_2)$ such that the post-processed prediction interval is given by

$$C(X_{n+1}) = [\hat{q}_{\alpha,low}(X_i) - Q_{1-\alpha,low}(E_{low}, I_2),\ \hat{q}_{\alpha,high}(X_i) + Q_{1-\alpha,high}(E_{high}, I_2)]\,.$$

This asymmetric version additionally requires a change in the computation of the conformity scores. Instead of considering the elementwise maximum of the differences between observed values Y_i and original bounds, we simply compute two separate score vectors:

$$E_{i,low} := \hat{q}_{\alpha,low}(X_i) - Y_i \quad \forall\, i \in I_2$$
$$E_{i,high} := Y_i - \hat{q}_{\alpha,high}(X_i) \quad \forall\, i \in I_2$$

1.2.2 Results

To avoid repetitions with respect to Section 1.1 we only briefly mention where asymmetric and vanilla CQR have similar effects and rather focus on the differences.

Figure 6 nicely demonstrates the key characteristics of asymmetric CQR: Adjustments of the lower and upper interval bounds are independent from each other. Considering the last interval on the far right the

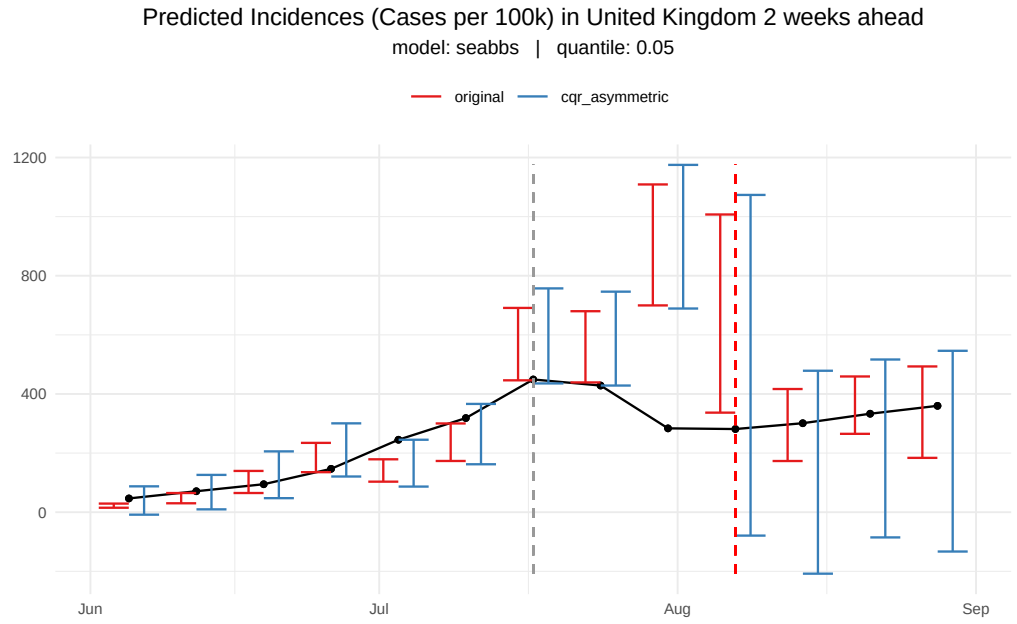

Figure 6: Illustration of CQR's slow reaction process

lower bound is adjusted downwards by a large amount whereas the upper bound is increased only slightly. This behaviour implies that, in contrast to traditional CQR, original and corrected prediction intervals are generally *not* centered around the same midpoint.

The plot also illustrates what we have already seen in Section 1.1: Once the true value is not contained in the prediction interval and there is a large discrepancy towards the closest boundary, all CQR versions tend to *overcompensate* in the next time step. This jump can be observed from time step 9 to time step 10, where the latter is highlighted by the red dashed line. Even more problematic, the large correction margin only vanishes very gradually afterwards even if the observed Time Series has stabilized. In Figure 6 the lower quantile prediction of asymmetric CQR approaches the original lower quantile forecast very slowly after the jump in observed Cases. The following paragraphs aim to explain this inflexibility in detail and draw the connection to the underlying algorithm that was developed ealier in this chapter.

1.2.3 CQR Downsides

Going back to Section 1.2.1 asymmetric CQR computes two separate score vectors based on the original lower and upper quantile forecasts and the vector of observed values. To confirm our findings visually we select the data subset of Figure 6.

Consider the intervals one step prior to the red vertical line. At this point in time the training set includes the first 9 elements of true values and predicted quantiles which are then used to compute a list of lower and upper scores:

```
scores_list <- compute_scores_asymmetric(
  true_values[1:9], quantiles_low[1:9], quantiles_high[1:9]
)
scores_list$scores_lower
## [1]  -31.443366  -40.808821  -29.765120  -11.289450 -141.757533 -145.173165
## [7]   -2.839344   10.514219  415.998372
```

The vector of lower scores is calculated by the difference of true values and lower predicted quantiles at each time step. Due to the jump from time point 9 to 10 the final element of the lower score vector has a large value of 416.

Next, the (scalar) lower margin is computed:

```
margin <- compute_margin(scores_list$scores_lower, quantile)
margin
```

```
##     100%
## 415.9984
```

Due to the small sample size of 9 observations and the relatively small quantile level of 0.05 the margin is simply the *maximum* or 100% quantile of the lower scores. The *updated* lower quantile prediction for the 10th time point is thus the original lower quantile prediction at time point 10 minus the margin, i.e.

```
quantiles_low[10] - margin
```

```
## [1] -79.18
```

which coincides with Figure 6.

The procedure now continues by consecutively adding the next elements to the vector of true values and original quantile predictions. Since the differences in observed incidence of Cases and predicted lower bound are all much *smaller* for the remaining time steps, the **same** value 416 remains the maximum of the lower score vector until the end! Thus, if like in the case above, the margin always equaled the maximum score, the adjustments would stay that large independent of the future development of the time series.

In fact, the only difference from that scenario to **Step 4** of Section 1.1.2 is that the *quantile* of the score vector that determines the value of the margin depends on the *size* of the score vector. Since the size increases by ine with each time step during the Time Series Cross Validation process, this quantile slowly declines. For instance, the margin which is responsible for adjusting forcasts at time point 11 is not simply the maximum anymore:

```
scores_list <- compute_scores_asymmetric(
  true_values[1:10], quantiles_low[1:10], quantiles_high[1:10]
)
margin <- compute_margin(scores_list$scores_lower, quantile)
margin
```

```
##      99%
## 383.5547
```

In this case the 99% quantile is an interpolation of the largest and second largest score, as implemented by the `stats::quantile()` function.

The cycle proceeds in this way until the end. The conclusion of this brief case study is that all modifications of the traditional CQR algorithm suffer from a slow reaction time towards distribution shifts and particularly sudden jumps within observed values and original forecasts. This major downside of Conformalized Quantile Regression is an immediate consequence of the **margin** computation which finally determines the magnitude of forecast adjustments.

TODO: Add Global Results that show differences to traditional CQR

1.3 Multiplicative CQR

1.3.1 Theory

On top of the asymmetric CQR version introduced in Section 1.2, we can extend the CQR algorithm further. So far, the adjustments to the original prediction interval were always chosen in *additive* form. It may be useful to leverage the *magnitude* of the original bounds more explicitly by using *relative* or *multiplicative* adjustments.

Hence, we again compute separate margins $Q_{1-\alpha,low}(E, I_2)$ and $Q_{1-\alpha,high}(E, I_2)$ which are now *multiplied* with the existing forecasts. The post-processed prediction interval is then given by

$$C(X_{n+1}) = \left[\hat{q}_{\alpha,low}(X_i) \cdot Q_{1-\alpha,low}(E_{low}, I_2),\ \hat{q}_{\alpha,high}(X_i) \cdot Q_{1-\alpha,high}(E_{high}, I_2)\right].$$

Just like the asymmetric version, the computation of the score vectors is changed accordingly to respect the new multiplicative relationship:

$$E_{i,low} := \frac{Y_i}{\hat{q}_{\alpha,low}(X_i)} \quad \forall\, i \in I_2$$
$$E_{i,high} := \frac{Y_i}{\hat{q}_{\alpha,high}(X_i)} \quad \forall\, i \in I_2,$$

where we have to exclude original predictions with the value 0. Since in our application of Covid-19 Cases and Deaths all values are non-negative, we threshold the scores at zero such that $E_{i,low}$ equals 0 whenever $\hat{q}_{\alpha,low}(X_i) \leq 0$.

1.3.2 Regularization

While the idea of multiplicative correction terms is appealing, it turns out that the approach above is flawed in two ways:

1. Recall that the (lower) margin $Q_{1-\alpha,low}(E, I_2)$ basically *picks* a value of the score vector E_{low} at a given quantile level. The score vectors are computed for each combination of *location*, *model*, *target type*, *horizon* and *quantile*, i.e. the number of values in the score vector is identical to the number of distinct time points in the training set. For short time series such as our small UK data set, the margin selects the *largest* value in the score vector for small levels of α such as 0.01 or 0.05, where each such value represents a *ratio* of observed Y_i and original prediction $\hat{q}_{\alpha,low}(X_i)$.

   As one might guess, these factors frequently get very large for small initial quantile predictions $\hat{q}_{\alpha,low}(X_i)$ such that the selected margin for post-processing is unreasonably large. In fact, the margin can remain huge if there exists a *single* outlier in the score vector. In particular, this naive multiplicative version frequently adjusts the lower quantile prediction to a higher value than its upper quantile counterpart.

   We counteract this extreme sensitivity to outliers by *reducing the spread* inside of the score vector to make it more well behaved. Since we deal with multiplicative factors it makes no sense to standardize them to zero mean and unit variance. Instead, we regularize the score vector by pulling all values closer to 1, while keeping all values nonnegative and respecting their *directions*, i.e. values smaller than 1 that reduce the interval width keep doing so but to a lesser extent than before and, analogously, prior values greater than one remain to be greater than 1.

   This goal is achieved by a *root transformation*. Since a greater spread of the score vector should lead to larger regularization we settled on the corrections

$$E_{i,low}^{reg} = E_{i,low}^{\left(\frac{1}{\sigma_{E_{low}}}\right)}, \quad E_{i,high}^{reg} = E_{i,high}^{\left(\frac{1}{\sigma_{E_{high}}}\right)},$$

   where σ_E denotes the standard deviation of the corresponding score vector.

2. Chances are high that at least *one* of the original true values Y_i is larger than its corresponding lower quantile prediction $\hat{q}_{\alpha,low}(X_i)$ such that the maximum of the (regularized) score vector is still larger than 1. Thus, the lower bound for small quantiles α is almost *always* pushed upwards. The same logic applies to the upper bound in which case the entire interval is shifted to the top. This behaviour is usually not desired.

   To prevent interval shifts, we add the additional constraint that the lower and upper margin must multiply to 1, i.e.

$$Q_{1-\alpha,low} \cdot Q_{1-\alpha,high} \overset{!}{=} 1.$$

Table 5: Performance of Multiplicative CQR on the Training Set

method	interval_score	dispersion	underprediction	overprediction
cqr_multiplicative	24.49	5.98	18.01	0.51
original	23.62	4.16	18.88	0.58

Hence, when the lower bound is adjusted upwards ($Q_{1-\alpha,low} > 1$), the upper bound *must* decrease ($Q_{1-\alpha,high} < 1$) and the interval becomes smaller. Similarly, when the upper bound is adjusted upwards ($Q_{1-\alpha,high} > 1$), the lower bound must decrease ($Q_{1-\alpha,low} < 1$) leading to larger intervals overall after post-processing.

1.3.3 Results

As noted in the previous section, *naive* multiplicative Conformalized Quantile Regression without any regularization is useless for post-processing quantile predictions. Typically, one can observe strong overfitting on the training set such that the training performance indicates promising effects, yet the scores on the validation set are *much* worse than the original forecasts. Further, the adjusted intervals are shifted upwards and usually too large.

Before numerically evaluating the performance of *regularized* CQR, it is instructive to look at a visual comparison of the original and post-processed forecasts of all three CQR modifications for one specific feature combination, which is shown in Figure 7.

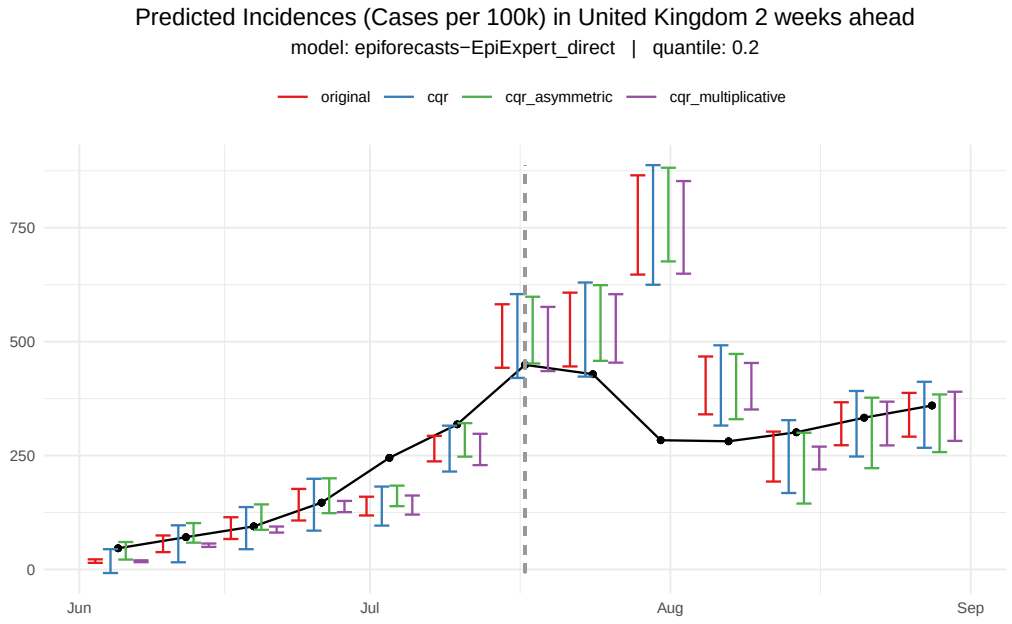

Figure 7: Comparison of CQR variations on the UK data set

The effect of scaling the score vectors in step 1 of the regularization procedure and constraining lower and upper margins in the second step can be detected immediately: Similar to vanilla CQR, the corrected intervals are now centered around the same midpoint as the original forecasts. In strong contrast to the additive CQR versions, however, the issue of interval explosion has not only been diminished by downscaling the scores, but rather *reversed* such that the interval widths now actually *decreased* at most time points and generally appear too narrow.

Moreover, we no longer have any theoretical guarantees of improved forecasts on the training set since

Table 6: Performance of Multiplicative CQR by Model on the Validation Set

method	model	interval_score	dispersion
cqr_multiplicative	epiforecasts-EpiExpert	71.60	10.05
original	epiforecasts-EpiExpert	67.74	12.07
cqr_multiplicative	EuroCOVIDhub-baseline	29.85	17.95
original	EuroCOVIDhub-baseline	29.61	5.92
cqr_multiplicative	EuroCOVIDhub-ensemble	61.24	15.48
original	EuroCOVIDhub-ensemble	56.07	14.00
cqr_multiplicative	seabbs	98.18	9.14
original	seabbs	95.11	14.03

Table 7: Dispersion of Multiplicative CQR by Quantile on the Validation Set

method	0.01	0.025	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45
cqr_multiplicative	8.79	7.40	11.98	16.83	18.26	18.82	18.24	17.53	14.60	10.12	5.22
original	2.82	5.82	9.65	14.86	17.84	18.99	18.83	17.44	14.71	10.97	6.08

Theorem 1.1 only applies to the original additive and symmetric version of CQR. This fact is confirmed empirically by Table 5 which shows the Weighted Interval Score aggregated over all categories of `model`, `target_type`, `horizon` and `quantile`. Indeed, the multiplicative adjustments result in a slightly worse Weighted Interval Score.

Recall that this behaviour is different from the unregularized version, which performed better than the original forecasts across almost all feature combinations. On the flipside, the performance on the validation set improved dramatically compared to the naive implementation, even though it does *not* lead to a score improvement in absolute terms as is shown in Table 6 separated by the forecasting `model`. Interestingly, multiplicative CQR indicates the strongest *relative* performance for the `EuroCOVIDhub-baseline` model where the additive CQR algorithms struggle the most. Overall the score differences across different forecasting models appear to be smoothed out compared to the previous CQR versions which also results from the regularization component that is unique to the multiplicative modification.

The impression of too narrow adjusted intervals does not generalize to the entire data set. The `dispersion` column in Table 6 shows that the intervals are downsized only for some models such as `epiforecasts-EpiExpert` whereas for others like `epiforecasts-ensemble` the distance between lower and upper bound is larger on average.

Table 7 suggests a connection of the `dispersion` change by multiplicative CQR with the `quantile` level. Aggregated over all models, target types and horizons the dispersion value is increased by a large amount for extreme quantiles and remains in a similar range to before towards the quantiles in the center of the predictive distribution. This behaviour is in line with the previously seen additive correction methods and underlines that Figure 7 is not representative for all feature combinations in the UK data set.

Overall, we must conclude that the original CQR algorithm as described in (Romano, Patterson, and Candès 2019) can *not* be modified towards multiplicative margins in any straightforward way. For this reason, we do not extend the analysis of multiplicative CQR to the European Forecast Hub data set and do not include it in the detailed method comparison in **??**.

Romano, Yaniv, Evan Patterson, and Emmanuel J. Candès. 2019. "Conformalized Quantile Regression." *arXiv:1905.03222 [Stat]*, May. http://arxiv.org/abs/1905.03222.