

IMF and Benchmark Forecasts

1 Extracting error quantiles

Consider a forecast that stems from a source s for a specific target k in a country j , for target year t and with forecast horizon h :

$$\hat{y}_{s,k,j,t,h}$$

For example, this could be a forecast stemming from the International Monetary Fund World Economic Outlook ($s = IMF$) for real GDP growth ($k = gdp$) in Canada ($j = Canada$) for the year 2022 ($t = 2022$). h then indexes the forecast horizon, where we code:

$$h = \begin{cases} 0, & \text{for forecasts made in October of the same year} \\ 0.5, & \text{for forecasts made in April of the same year} \\ 1, & \text{for forecasts made in October of the previous year} \\ 1.5, & \text{for forecasts made in April of the previous year} \end{cases}$$

After the target year has completed, we obtain the realized value for the quantity of interest. For these, the WEO updates publishes biannual updates for two years, yielding 4 versions of the realized value. In accordance with previous literature (*cite Timmermann 2008*), we use the version that is published in October of the following year and thereby don't index the true value by its publishing date (*rephrase*). We thus write the true value as

$$\hat{y}_{k,j,t}$$

Given the forecast and the realized value for the quantity of interest, we can calculate the respective forecast error as

$$e_{s,k,j,t,h}^d = y_{k,j,t} - \hat{y}_{s,k,j,t,h}$$

15 for the “directional” error method and as

$$e_{s,k,j,t,h}^a = |y_{k,j,t} - \hat{y}_{s,k,j,t,h}|$$

16 for the “absolute” error method.

17 The objective is to extract quantiles from sets of errors $\mathcal{E}_{s,k,j,t,h}$ constructed of certain years, depending on the
 18 estimation method m , to be able to quantify the uncertainty inherent in the forecasts via central prediction intervals
 19 of level $\alpha = \{0.5, 0.8\}$. For the estimation method, we consider a “rolling window” method, an “expanding window”
 20 method, and a “leave-one-out” method. For the rolling window method ($m = rw$), the errors of the last nine years
 21 enter into the estimation. For the expanding window method ($m = ew$), all previous years are considered, leaving
 22 a nine year window up front for the first estimation. For the leave-one-out method, all years except the current
 23 target year enter the estimation set. The latter is of course equivalent to the expanding window method in a real
 24 time setting and is considered in the scope of this analysis as a mere check *rephrase*. As an example, the error set
 25 for the “directional” error method and the rolling window approach is

$$\mathcal{E}_{s,k,j,t,h}^{d,rw} = \{e_{s,k,j,t^*,h}^d | t - 9 \leq t^* < t\}$$

26 Insert reasoning to use the past 9 errors.

27 To now obtain the lower l and upper u values for a central prediction interval of level α , we take quantiles of these
 28 sets and add them to the current prediction:

29 For the directional method:

$$l_{t,h,v,l,j}^{\alpha,d} = \hat{y}_{t,h,l,j} + q^{0.5-\alpha/2} \left(\mathcal{E}_{t,h,v,l,j}^{d,m} \right)$$

30

$$u_{t,h,v,l,j}^{\alpha,d} = \hat{y}_{t,h,l,j} + q^{0.5+\alpha/2} \left(\mathcal{E}_{t,h,v,l,j}^{d,m} \right)$$

31 And for the absolute method:

$$l_{t,h,v,l,j}^{\alpha,a} = \hat{y}_{t,h,l,j} - q^{\alpha} \left(\mathcal{E}_{t,h,v,l,j}^{m,a} \right)$$

32

$$u_{t,h,v,l,j}^{\alpha,a} = \hat{y}_{t,h,l,j} + q^{\alpha} \left(\mathcal{E}_{t,h,v,l,j}^{m,a} \right)$$

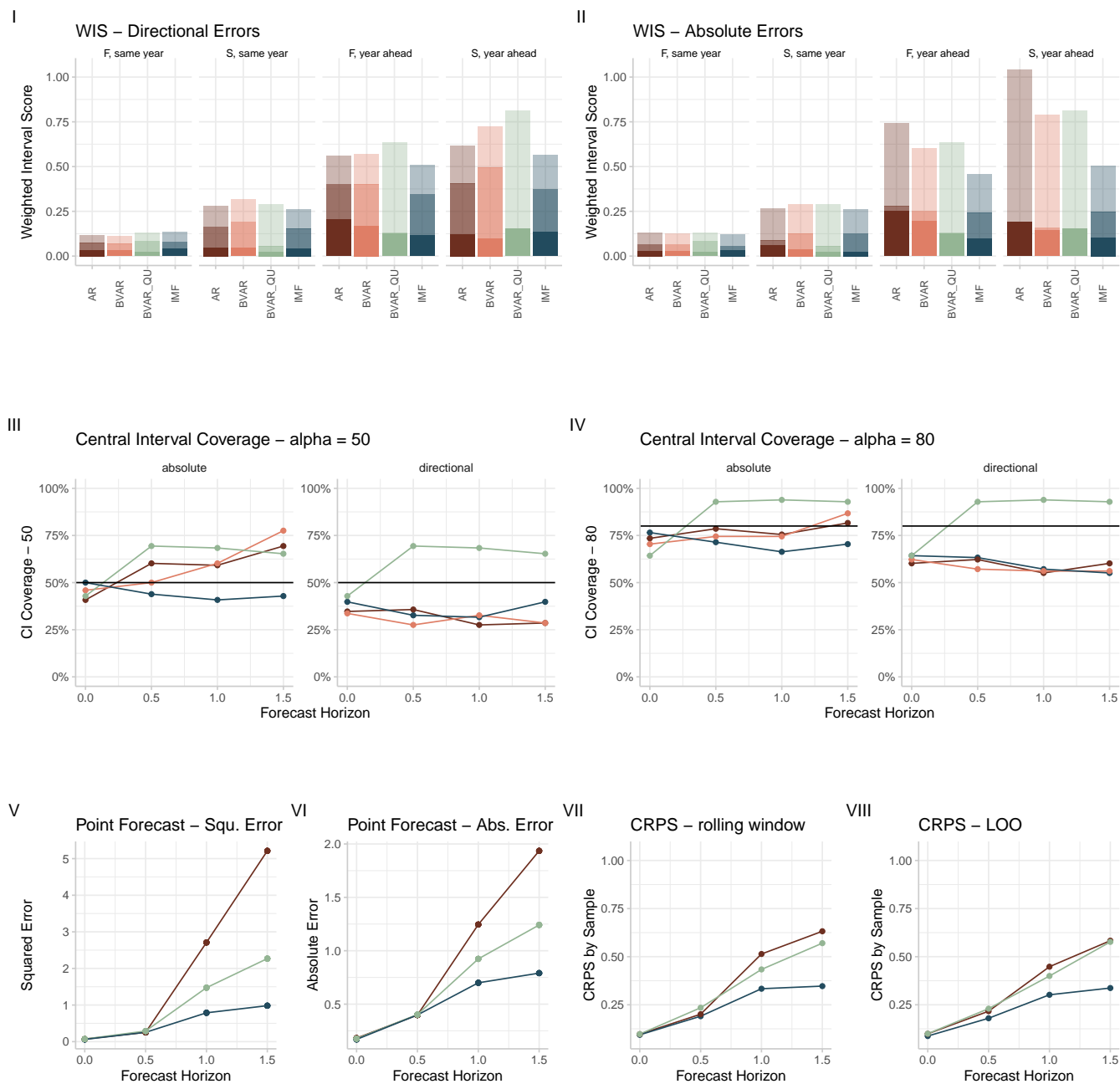
33 Two different philosophies.

34 The absolute method will always yield symmetric central prediction intervals around the forecast value, while the
35 directional method will in general yield asymmetric intervals. They thus result in different central intervals, unless
36 the errors in \mathcal{E} are perfectly symmetric around zero¹. In fact, the directional method can yield central prediction
37 intervals that do not even contain the forecast value, in cases where the $(0.5 - \alpha/2)$ -quantile is positive or the
38 $(0.5 + \alpha/2)$ -quantile is negative.

¹Not totally correct, actually. For this to hold exactly, the error set would need to be augmented with one zero value.

2 Scores, by error method, Horizon and forecast source

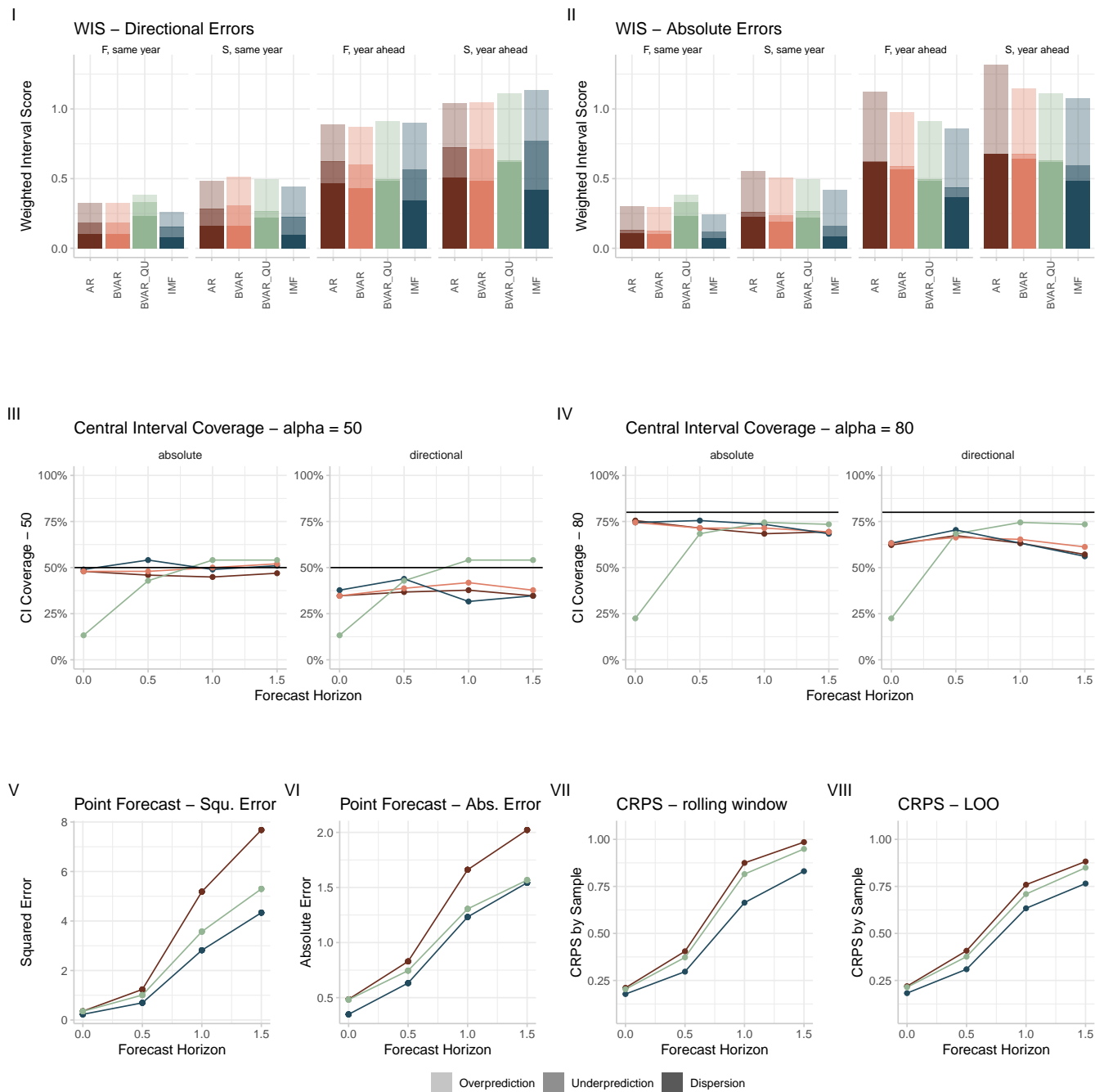
2.1 Inflation



Overprediction Underprediction Dispersion

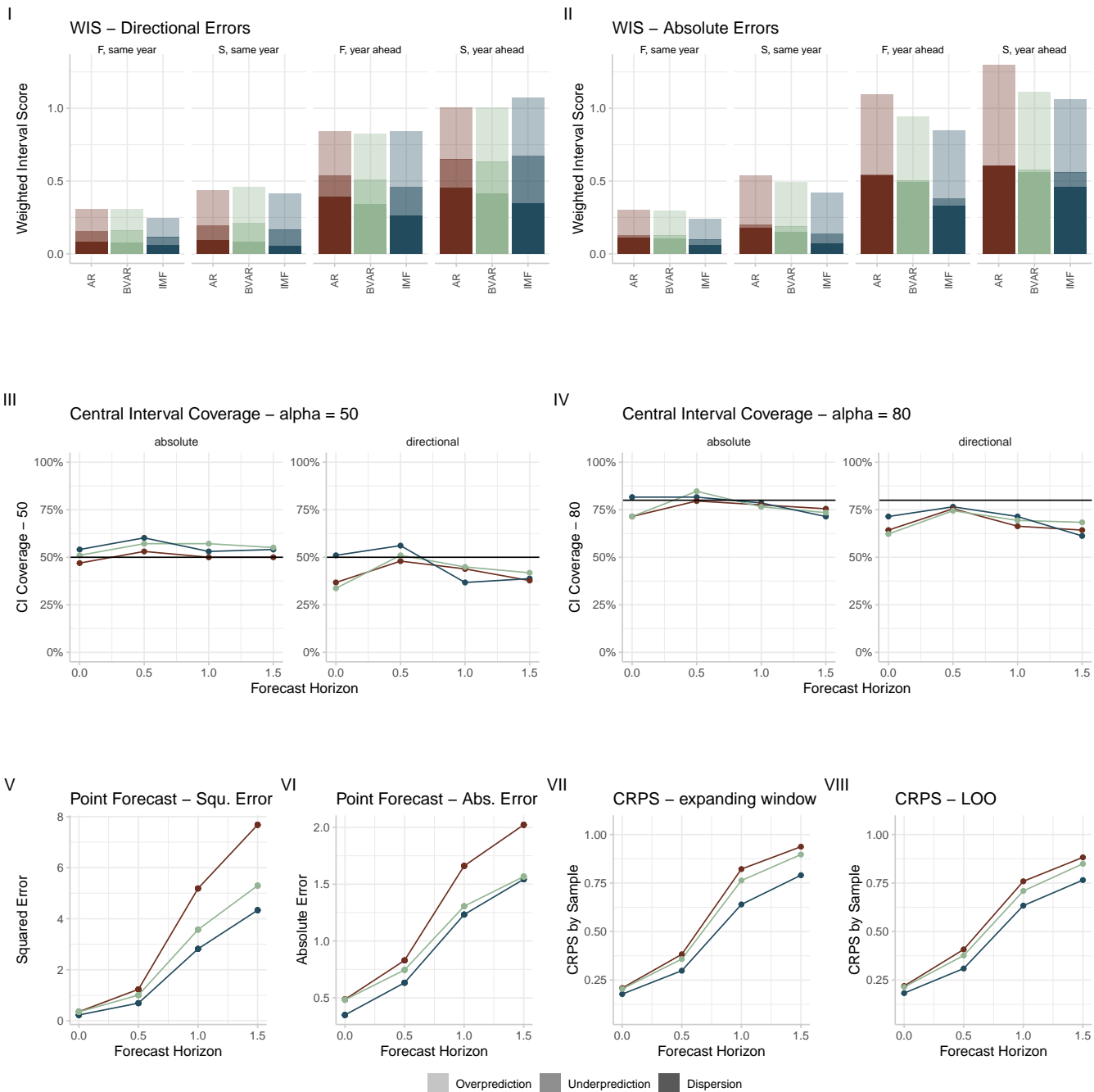
42 Some notes:

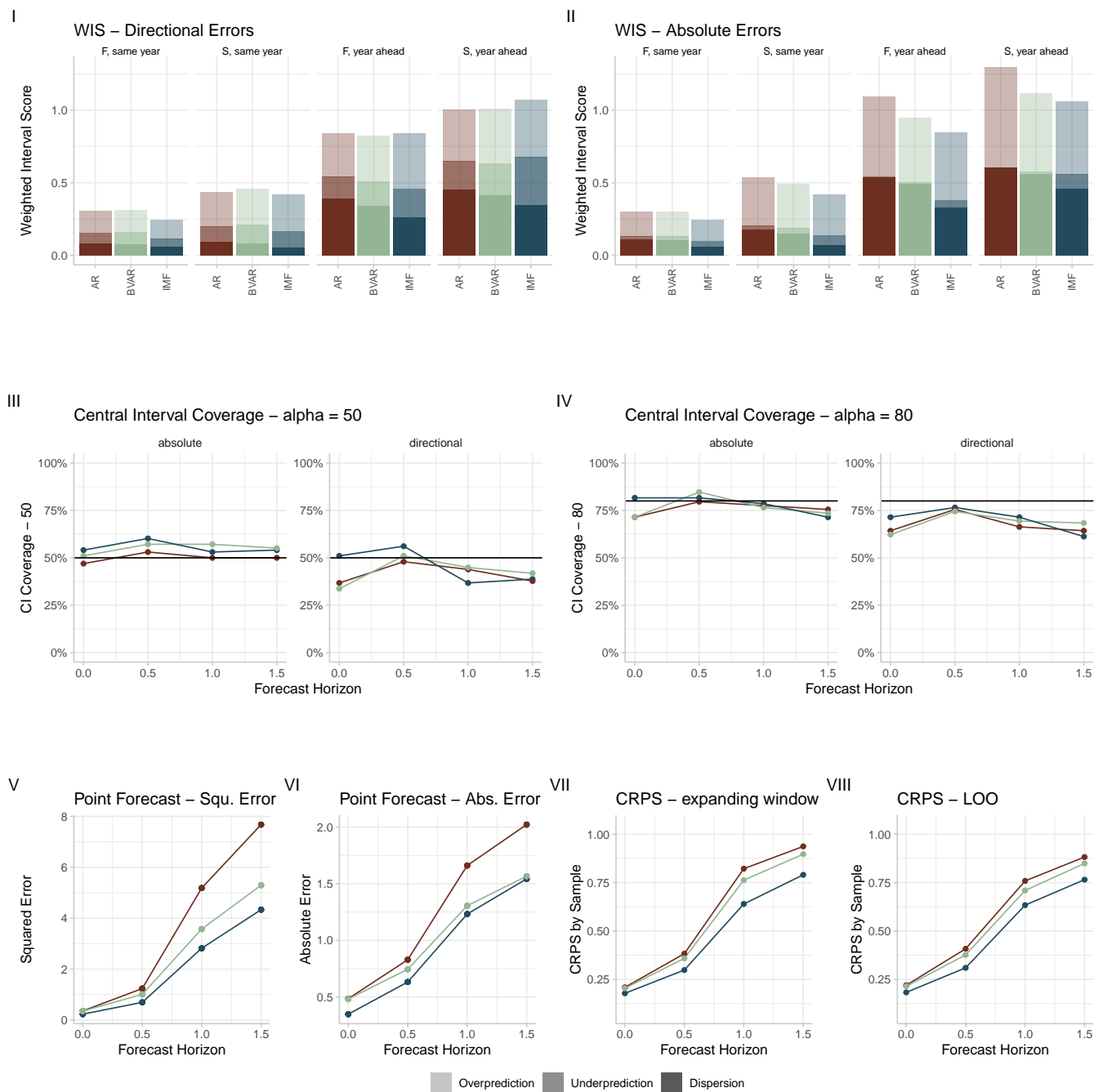
- 43 • Inflation: directional vs. absolute errors:
 - 44 – difference small for IMF method, absolute slightly better, likely due to longer central intervals
 - 45 – AR and BVAR profit more from directional correction (upward bias)
 - 46 – for expanding window method, difference in coverage is smaller (-> structural breaks)
- 47 • Inflation overall scores: IMF forecasts outperform others
 - 48 – lower scores for point forecasts
 - 49 – lower WIS
 - 50 – lower bias (compute directly?)
- 51 • GDP Growth: more similar results for different sources
 - 52 – lower scores at shorter horizons, more similar at larger horizons
 - 53 – IMF forecasts better only for absolute error method



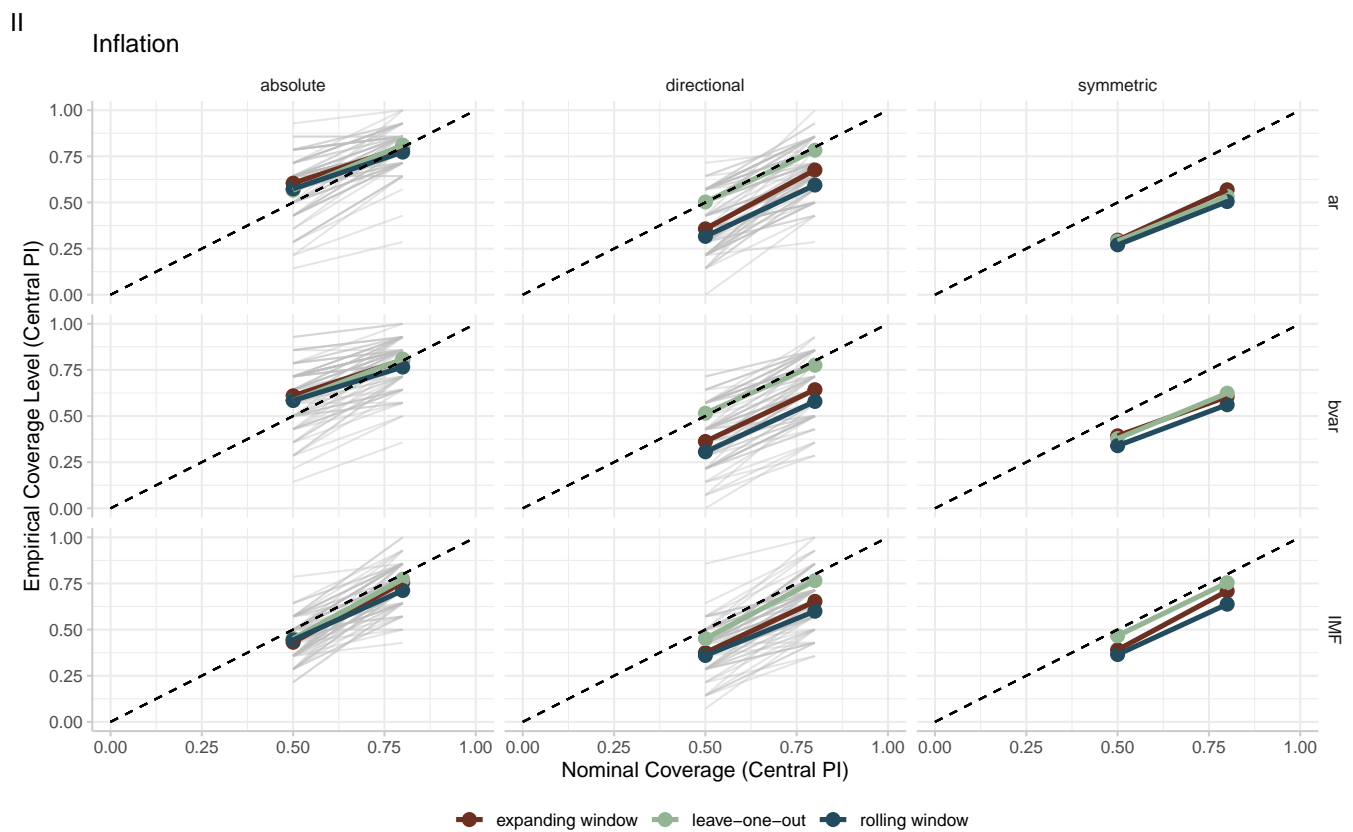
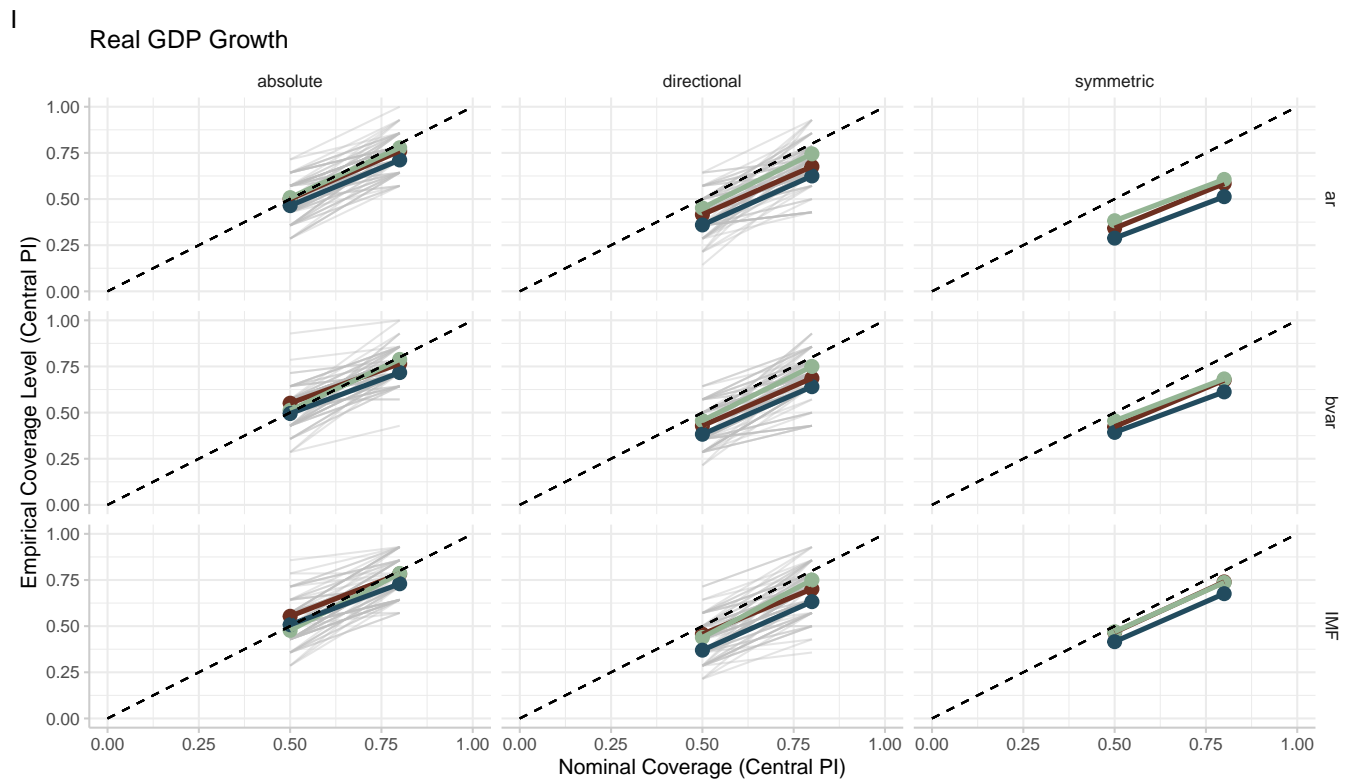
3 Expanding Window - Scores, by error method, Horizon and forecast source

3.1 Inflation





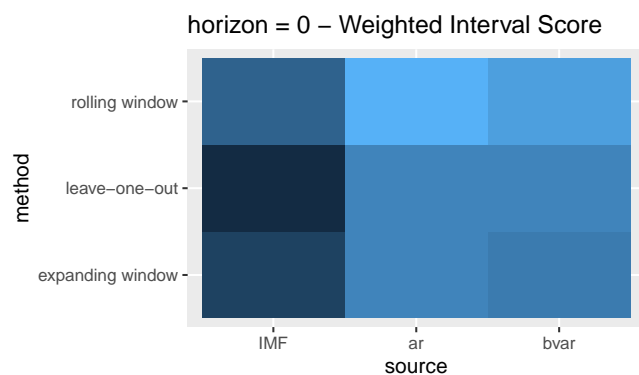
62 4 Coverage, by target, methods and source



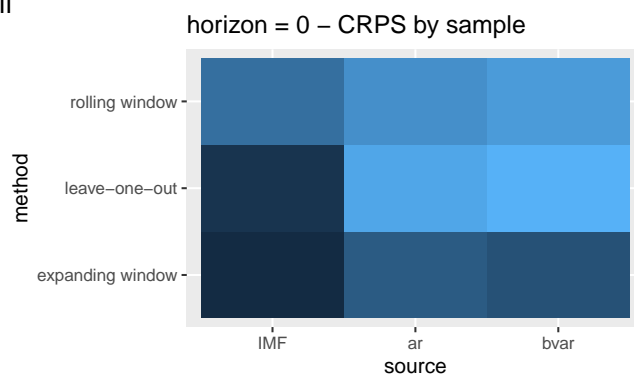
	IMF	ar	bvar
horizon = 0			
expanding window_interval_score	0.115	0.123	0.122
expanding window_sample_crps	0.087	0.092	0.091
leave-one-out_interval_score	0.112	0.123	0.123
leave-one-out_sample_crps	0.088	0.099	0.100
rolling window_interval_score	0.119	0.128	0.126
rolling window_sample_crps	0.094	0.097	0.098
horizon = 0.5			
expanding window_interval_score	0.258	0.272	0.296
expanding window_sample_crps	0.182	0.208	0.241
leave-one-out_interval_score	0.257	0.257	0.276
leave-one-out_sample_crps	0.180	0.217	0.230
rolling window_interval_score	0.261	0.266	0.288
rolling window_sample_crps	0.191	0.201	0.235
horizon = 1			
expanding window_interval_score	0.447	0.740	0.595
expanding window_sample_crps	0.327	0.504	0.426
leave-one-out_interval_score	0.443	0.746	0.587
leave-one-out_sample_crps	0.302	0.448	0.400
rolling window_interval_score	0.458	0.744	0.601
rolling window_sample_crps	0.333	0.514	0.434
horizon = 1.5			
expanding window_interval_score	0.503	1.051	0.812
expanding window_sample_crps	0.346	0.627	0.602
leave-one-out_interval_score	0.488	1.021	0.756

leave-one-out_sample_crps	0.337	0.583	0.577
rolling window_interval_score	0.506	1.043	0.790
rolling window_sample_crps	0.347	0.632	0.570

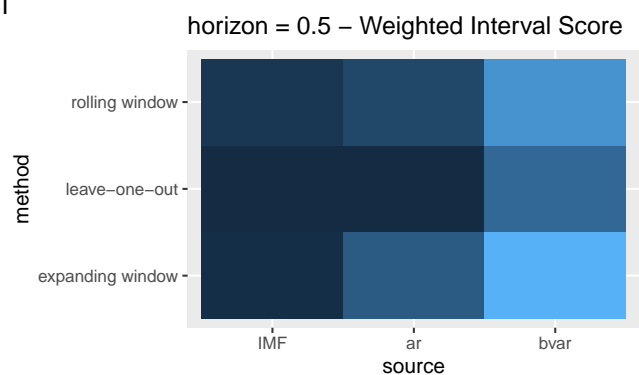
I



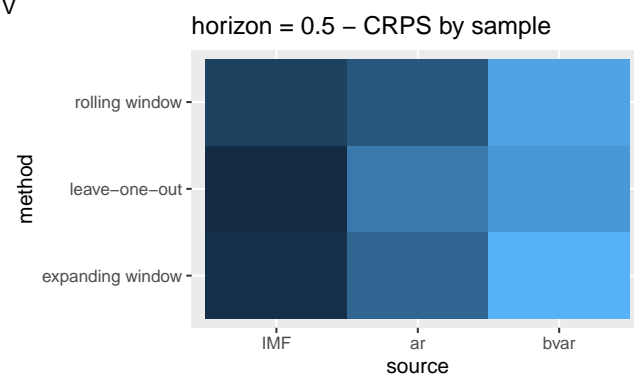
II



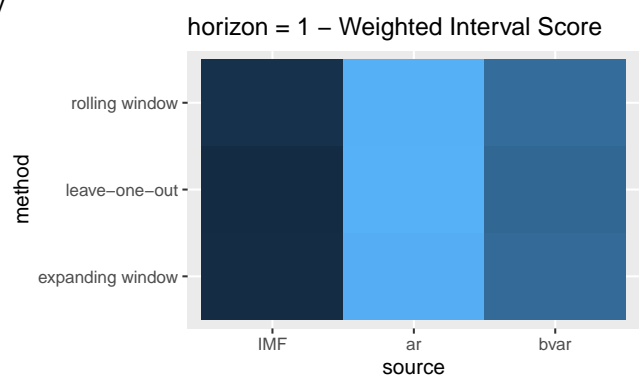
III



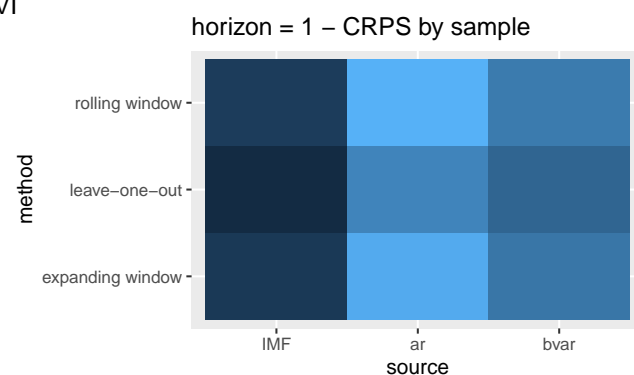
IV



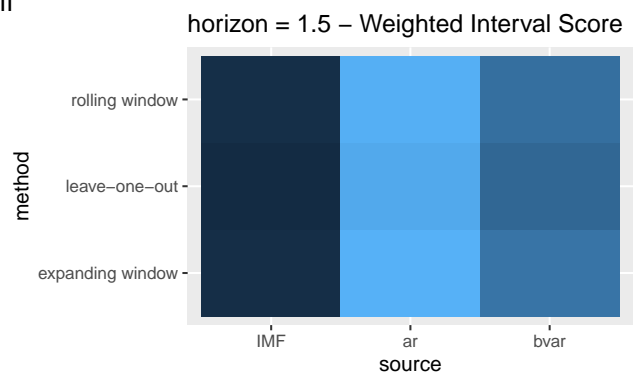
V



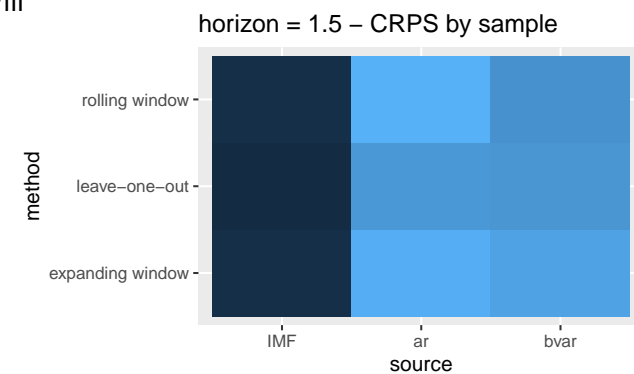
VI



VII



VIII

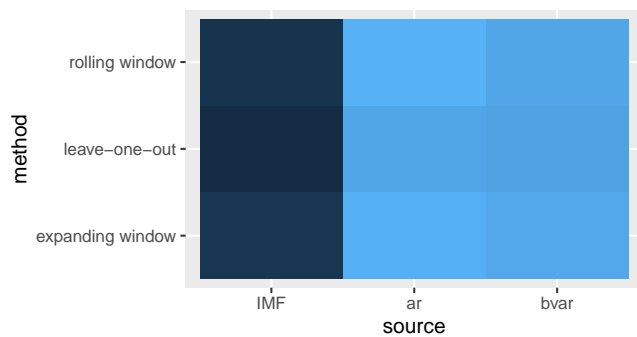


	IMF	ar	bvar
horizon = 0			
expanding window_interval_score	0.241	0.301	0.298
expanding window_sample_crps	0.178	0.209	0.204
leave-one-out_interval_score	0.235	0.297	0.295
leave-one-out_sample_crps	0.183	0.219	0.214
rolling window_interval_score	0.240	0.302	0.297
rolling window_sample_crps	0.178	0.211	0.204
horizon = 0.5			
expanding window_interval_score	0.416	0.540	0.493
expanding window_sample_crps	0.298	0.383	0.358
leave-one-out_interval_score	0.406	0.532	0.482
leave-one-out_sample_crps	0.310	0.408	0.377
rolling window_interval_score	0.416	0.554	0.504
rolling window_sample_crps	0.297	0.405	0.373
horizon = 1			
expanding window_interval_score	0.846	1.094	0.947
expanding window_sample_crps	0.640	0.822	0.763
leave-one-out_interval_score	0.833	1.083	0.933
leave-one-out_sample_crps	0.634	0.759	0.709
rolling window_interval_score	0.858	1.121	0.973
rolling window_sample_crps	0.663	0.875	0.815
horizon = 1.5			
expanding window_interval_score	1.063	1.297	1.113
expanding window_sample_crps	0.790	0.937	0.897
leave-one-out_interval_score	1.030	1.292	1.109

leave-one-out_sample_crps	0.765	0.882	0.849
rolling window_interval_score	1.075	1.313	1.150
rolling window_sample_crps	0.831	0.985	0.949

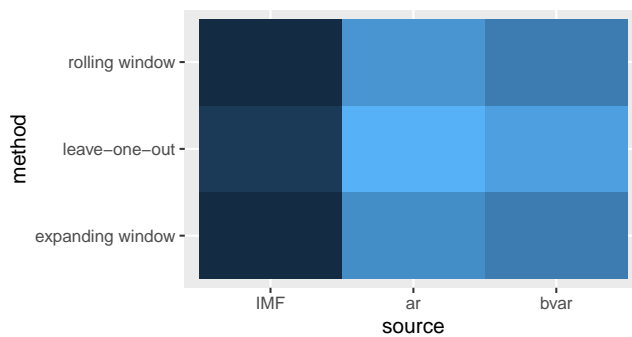
I

horizon = 0 – Weighted Interval Score



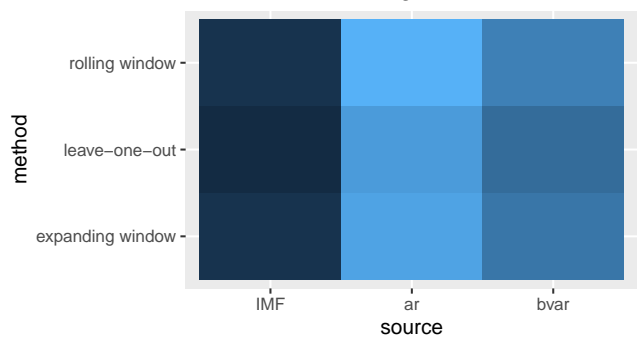
II

horizon = 0 – CRPS by sample



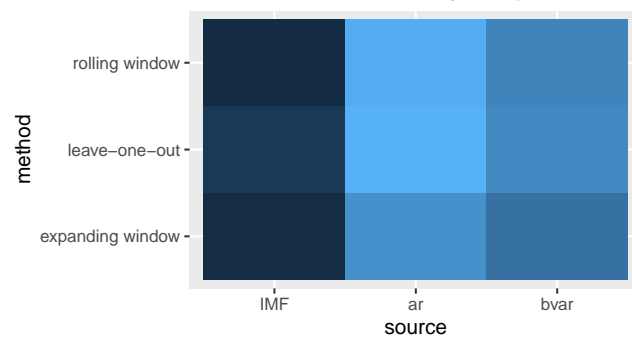
III

horizon = 0.5 – Weighted Interval Score



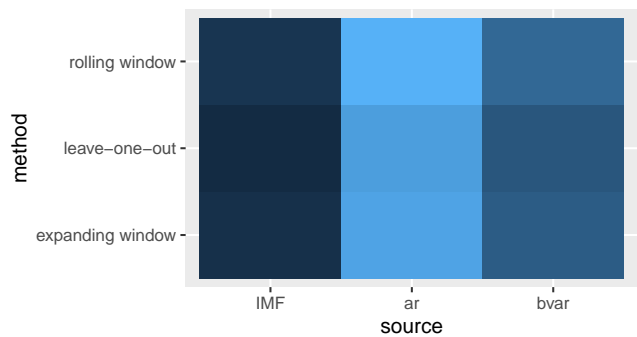
IV

horizon = 0.5 – CRPS by sample



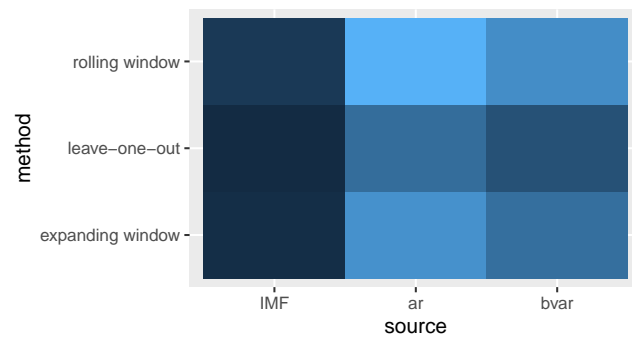
V

horizon = 1 – Weighted Interval Score



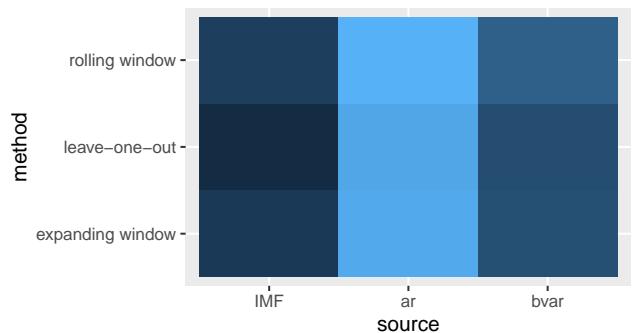
VI

horizon = 1 – CRPS by sample



VII

horizon = 1.5 – Weighted Interval Score



VIII

horizon = 1.5 – CRPS by sample

