



Verification Functions for Ensemble Forecasts Implemented in the R package SpecsVerification

Stefan Siegert
University of Exeter

Abstract

Keywords: keywords, comma-separated, not capitalized, Java.

1. Introduction

SPECS

Ensemble forecasting general

deterministic, probabilistic, ensemble

Forecast verification general a posteriori comparison of forecasts with their verifying observations; ensembles can be verified by taking the ensemble mean as a deterministic forecast; deriving a probability distribution and use a proper score; here: evaluate the raw ensemble

Finite size effect in ensemble verification Everything else being equal, larger ensembles yield better scores than smaller ensembles.

```
R > data(eurotempforecast)
R > ens <- ens[, 1:8]
R > ens.bin <- ens.bin[, 1:8]
R > ens.cat <- ens.cat[, 1:8]
R > yrs <- as.numeric(names(obs))
R > N <- length(obs)
```

```

R > par(las = 1, cex = 0.7, mgp = c(3, 1, 0), mar = c(2, 4, 1, 1))
R > plot(NA, type = "n", xlim = range(yrs), ylim = range(c(obs, ens)),
+       xlab = "", ylab = "temp [C]")
R > for (i in 1:ncol(ens)) points(yrs, ens[, i], col = "darkgray",
+       cex = 0.7)
R > points(yrs, obs, pch = 15, cex = 1.5)

```

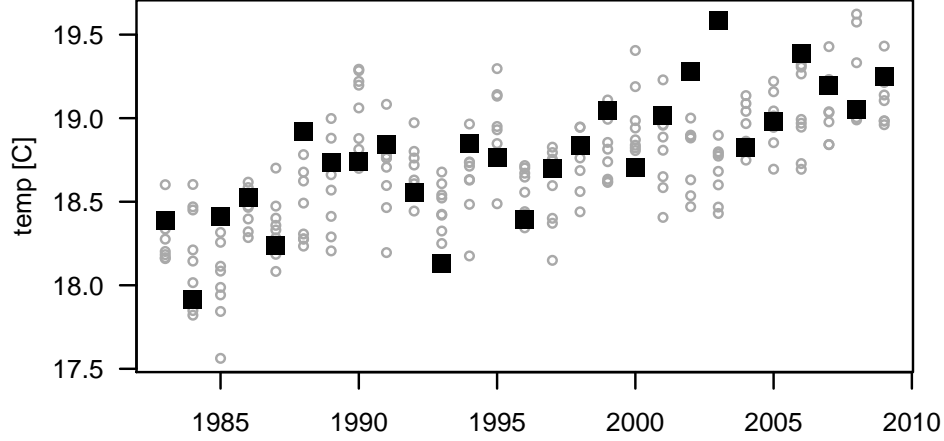


Figure 1: Seasonal European temperature forecasts by NCEP CFSv2, initialised in May, verified in JJA.

2. Ensemble-adjusted verification scores

2.1. Representation of ensemble and observation data

In **SpecsVerification**, archives of N instances of ensemble forecasts, each with R members, are represented by $N \times R$ matrices. The data shown in Figure 1 is a matrix with dimensions

```
R > dim(ens)
```

```
[1] 27 8
```

2.2. Binary forecasts

This section outlines the theory behind ensemble-adjusted verification scores, using probabilistic forecasts of binary events for illustration.

One of the most common verification measures for probabilistic forecasts of binary events is the Brier score (Brier 1950). Suppose a probability forecast $p_t \in [0, 1]$ is issued at time t for a binary (yes/no) event. The occurrence or non-occurrence of the event is coded as $y_t = 1$ or $y_t = 0$, respectively. The Brier score is given by the squared difference between forecast and observation:

$$s_B(p_t, y_t) = (p_t - y_t)^2 \quad (1)$$

The Brier score is negatively oriented - lower scores indicate better forecasts. The Brier score is a strictly proper verification score, meaning that the expected score obtains its minimum value if and only if the observation y_t is a random draw from p_t (Gneiting and Raftery 2007).

Assume next that instead of predicting the probability p_t , we make a prediction based on an ensemble forecast of size R , whose members were sampled identically and independently with probability p_t . That is, each of the R ensemble members is an independent Bernoulli trial with success probability p_t . An unbiased estimator of the success probability p_t is given by the fraction i_t/R , where i_t is the number of successes, i.e. the number of ensemble members that predict the event $y_t = 1$. The Brier score of the estimated probability is equal to

$$s_B\left(\frac{i_t}{R}, y_t\right) = \left(\frac{i_t}{R} - y_t\right)^2 \quad (2)$$

Taking expectation over the random variable $i_t \sim \text{Binomial}(p_t, R)$, it is shown that (Ferro, Richardson, and Weigel 2008)

$$E\left[s_B\left(\frac{i_t}{R}, y_t\right)\right] = s_B(p_t, y_t) + \frac{p_t(1-p_t)}{R} \quad (3)$$

That is, even though the fraction i_t/R is an unbiased estimator of the event probability p_t , the Brier score of i_t/R is not an unbiased estimator of the Brier score of p_t . The bias, given by the additional positive term on the rhs of Equation 3, depends on the ensemble size and vanishes for $R \rightarrow \infty$. The bias can be interpreted as a finite-ensemble penalty: If two ensembles sample their members from the same probability p_t , the one with the larger ensemble size obtains the lower (i.e. better) Brier score on average. This is reasonable since more ensemble members allow for more robust estimation of the “true” probability p_t . But in the analysis of ensemble hindcasts it is sometimes desirable to correct the finite-ensemble bias. For example, if the hindcast ensemble has R members, and future operational forecasts will be made with $R^* > R$ ensemble members, the score calculated for the R -member hindcast ensemble will be a too pessimistic estimate of the score of the R^* -member forecast ensemble.

The ensemble-adjusted Brier score, given by (Ferro *et al.* 2008)

$$s_B^*(i_t, R, R^*, y_t) = \left(\frac{i_t}{R} - y_t\right)^2 - \frac{i_t(R - i_t)}{R(R - 1)} \left(\frac{1}{R} - \frac{1}{R^*}\right) \quad (4)$$

allows to correct the finite-ensemble bias. The ensemble-adjusted Brier score is in expectation equal to the Brier score that would be achieved by an ensemble with R^* members sampled from the same probability p_t , i.e.,

$$E[s_B^*(i_t, R, R^*, y_t)] = (p_t - y_t)^2 + \frac{p_t(1-p_t)}{R^*}. \quad (5)$$

Note that, trivially, $s_B^*(i_t, R, R, y_t) = s_B(i_t/R, y_t)$. Note further that setting $R^* = \infty$ yields the fair Brier score (Ferro 2013) which estimates the score of the underlying probability p_t . The ensemble-adjusted Brier score can be used to compare ensemble forecasting systems with different numbers of members. It further allows for the extrapolation of the average score of an ensemble forecast system to larger ensemble sizes.

The **SpecsVerification** function **EnsBrier** calculates the ensemble-adjusted Brier scores of a collection of N ensemble forecasts and their corresponding binary observations. The argument **R.new** allows for estimation of the score of an arbitrary ensemble size, including **R.new=Inf**.

We transform the continuous ensemble data into binary by addressing the question "Will this year's summer be warmer than last year's"?

```
R > rbind(
+   unadjusted = mean(EnsBrier(ens.bin, obs.bin, R.new=NA)),
+   fair       = mean(EnsBrier(ens.bin, obs.bin, R.new=Inf))
+ )

           [,1]
unadjusted 0.1753472
fair       0.1534392
```

2.3. Categorical forecasts

Assume the ensemble forecasting system produces an ensemble of categorical rather than binary forecasts. That is, each ensemble members and the verifying observation falls into one of K classes. Two types of categorical forecasts can be distinguished: Disjoint categories and nested categories.

Assume the observation assumes on of K possible values, or classes, and a probabilistic forecast $\mathbf{p}_t = (p_{t,1}, \dots, p_{t,K})$, is issued. The verifying observation is vector-valued \mathbf{y}_t , where the k -th element of \mathbf{y}_t is $y_{t,k} = 1$ if the k -th class is observed, and $y_{t,k} = 0$ otherwise. The quadratic score for such a probability forecast is given by

$$s_Q(\mathbf{p}_t, \mathbf{y}_t) = \sum_{k=1}^K (p_{t,k} - y_{t,k})^2 \quad (6)$$

The quadratic score is simply the sum of Brier scores for the individual categories. Or stated differently, the Brier score is one-half the quadratic score of a 2-class categorical forecast.

Now assume an R -member categorical ensemble forecast \mathbf{i}_t is issued at time t , indicating that $i_{t,k}$ out of R ensemble members have predicted the k -th category, for $k = 1, \dots, K$. Using results obtained for the ensemble-adjusted Brier score, the ensemble-adjusted quadratic score is seen to be

$$s_Q^*(\mathbf{i}_t, R, R^*, \mathbf{y}_t) = \sum_{k=1}^K \left\{ \left(\frac{i_{t,k}}{R} - y_{t,k} \right)^2 - \left(\frac{1}{R} - \frac{1}{R^*} \right) \frac{i_{t,k}(R - i_{t,k})}{R(R-1)} \right\} \quad (7)$$

The ensemble adjusted quadratic score is implemented as the function **EnsQs** in **SpecsVerification**.

The quadratic score is insensitive to relabelling the K categories. Sometimes this is undesired in categorical forecasting problems. The forecast categories might have a natural ordering, such as low/medium/high intensity of precipitation. In that case, an order-sensitive score might be desired that penalises forecasts that assign most probability to the "high" category more than it penalises forecasts that assign most probability to the "medium" category, if the low category verifies. A lot of probability in the "medium" category can be considered closer to the observation than a lot of probability in the "high" category, such that the former forecast can be considered "better". An order-sensitive score is also desired if the forecast

categories are nested within one another, e.g., for precipitation amounts categorised into $<1\text{mm}/<5\text{mm}/<20\text{mm}/<\infty$. In that case, a high probability assigned to the $<5\text{mm}$ category should receive less penalty than a high probability assigned to the $<20\text{mm}$ category, if the actual amount is $<1\text{mm}$.

The ranked probability score (RPS) is an order-sensitive quadratic score for categorical probability forecasts. The forecast vector \mathbf{i}_t is transformed to the K -element cumulated forecast vector \mathbf{j}_t , with k -th element equal to $j_{t,k} = \sum_{l=1}^k i_{t,l}$. Likewise, the cumulated observation vector \mathbf{z}_t has its k -th element equal to $z_{t,k} = \sum_{l=1}^k y_{t,l}$. The RPS is the quadratic score achieved by the cumulative forecast \mathbf{j}_t for the cumulative observation \mathbf{z}_t . Accumulating the elements of \mathbf{i}_t and \mathbf{y}_t nests the K forecast categories within each other. The forecast is transformed from “ $i_{t,k}$ out of R ensemble members predict category k ” to the forecast “ $j_{t,k}$ out of R ensemble members forecast category k or less”. The nesting of forecast categories ensures order-sensitivity of the score. Using results from the previous section, we get the ensemble-adjusted RPS

$$s_R^*(\mathbf{i}_t, R, R^*, \mathbf{y}_t) = \sum_{k=1}^K \left\{ \left(\frac{j_{t,k}}{R} - z_{t,k} \right)^2 - \left(\frac{1}{R} - \frac{1}{R^*} \right) \frac{j_{t,k}(R - j_{t,k})}{R(R-1)} \right\} \quad (8)$$

The ensemble adjusted RPS is implemented as the function **EnsRps** in **SpecsVerification**.

We transform the continuous ensemble forecasts into categorical forecasts by addressing the question “Will this year’s summer temperature be similar to last year’s temperature (within a 0.5K range), colder, or warmer?”

```
R > cbind(
+ qs = c(
+   unadjusted = mean(EnsQs(ens.cat, obs.cat, R.new=NA)),
+   fair       = mean(EnsQs(ens.cat, obs.cat, R.new=Inf))
+ ),
+ rps = c(
+   unadjusted = mean(EnsRps(ens.cat, obs.cat, R.new=NA)),
+   fair       = mean(EnsRps(ens.cat, obs.cat, R.new=Inf))
+ ))
```

	qs	rps
unadjusted	0.6388889	0.3842593
fair	0.5873016	0.3752205

2.4. Continuous forecasts

If the forecast target is a continuous variable, such as temperature or pressure, the continuous ranked probability score (Matheson and Winkler 1976) can be used for forecast verification. If the forecast for the continuous target y_t is given as a cumulative distribution function $F_t(x)$, the CRPS is given by

$$s_C(F_t, y_t) = \int_{-\infty}^{\infty} dz [F_t(z) - H(z - y_t)]^2 \quad (9)$$

where $H(x)$ is the Heaviside step-function, satisfying $H(x) = 1$ for all $x \geq 0$ and $H(x) = 0$ otherwise. Suppose an ensemble forecast x_t with R real-valued members $x_t = \{x_{t,1}, x_{t,2} \dots, x_{t,R}\}$ is issued for the real-valued verifying observation y_t . The ensemble can be transformed into a cdf by taking the empirical distribution function given by

$$\hat{F}_t(z) = \frac{1}{R} \sum_{r=1}^R H(z - x_{t,r}). \quad (10)$$

Using properties of the Heaviside function, it is straightforward to show that the CRPS of the empirical distribution \hat{F} is given by

$$s_C(\hat{F}_t, y_t) = \frac{1}{R} |x_{t,r} - y_t| - \frac{1}{2R^2} \sum_{r=1}^R \sum_{r'=1}^R |x_{t,r} - x_{t,r'}|. \quad (11)$$

Fricker, Ferro, and Stephenson (2013) show that the CRPS is sensitive with respect to the ensemble size, and propose the ensemble-adjusted CRPS

$$s_C^*(x_t, R, R^*, y_t) = \frac{1}{R} \sum_{r=1}^R |x_{t,r} - y_t| - \frac{1}{2R(R-1)} \left(1 - \frac{1}{R^*}\right) \sum_{r=1}^R \sum_{r'=1}^R |x_{t,r} - x_{t,r'}|. \quad (12)$$

The ensemble-adjusted CRPS is, in expectation, equal to the CRPS that the empirical distribution function calculated from an ensemble of size R^* would achieve. This includes the case $R^* = \infty$, for which the fair CRPS is obtained. The ensemble-adjusted CRPS is implemented in the **SpecsVerification** function `EnsCrps`.

```
R > rbind(
+   unadjusted = mean(EnsCrps(ens, obs, R.new=NA)),
+   fair       = mean(EnsCrps(ens, obs, R.new=Inf))
+ )

      [,1]
unadjusted 0.1650061
fair       0.1499564
```

The ensemble adjusted Ignorance score has recently been proposed (?).

3. Comparative verification and uncertainty quantification

3.1. Reference forecast

The value of a verification score by itself is meaningless. In order to evaluate the skill of a forecast, its verification score has to be compared to the score achieved by a reference forecast. For example, if the skill of a state-of-the-art high resolution climate model is evaluated, it is reasonable to compare its verification score to the score achieved by an older climate model, possibly with lower resolution and less physical detail.

In the absence of a dynamical climate model to which the score can be compared, simple statistical benchmark predictions can be used. A popular simple reference forecast is the climatological forecast, which is only based on the known record of observations, without reference to any numerical forecast model. **SpecsVerification** includes the function **ClimEns** which transforms a vector of observations into a matrix of climatological ensemble forecasts, including the possibility to leave out the t -th observation in the t -th climatological ensemble:

```
R > ens.ref      <- ClimEns(obs,      leave.one.out=TRUE)
R > ens.cat.ref  <- ClimEns(obs.cat, leave.one.out=TRUE)
R > ens.bin.ref  <- ClimEns(obs.bin, leave.one.out=TRUE)
```

The new data set of climatological ensembles can be used as a reference ensemble to which the numerical forecast ensemble can be compared. We recommend also considering statistical reference forecasts such as a linear trend or an auto-regressive model, which might be more suitable than the climatological forecast.

3.2. Mean scores and mean score differences

Suppose we have calculated two time series $\{s_{1,1}, s_{1,2}, \dots, s_{1,N}\}$ and $\{s_{2,1}, s_{2,2}, \dots, s_{2,N}\}$ of verification scores for two competing forecast systems for the same observation. [Diebold and Mariano \(1995\)](#) suggest to test the null-hypothesis of equal forecast accuracy using the time series d_1, \dots, d_N of loss differentials $d_t = s_{1,t} - s_{2,t}$. Under the assumption of temporal independence of d_t , and zero mean of the loss-differential, the test statistic

$$T = \bar{d} \sqrt{\frac{N}{\text{var}(d_t)}} \quad (13)$$

is asymptotically Normally distributed with mean zero and variance one. This test is implemented in **SpecsVerification** in the function **ScoreDiff**. The function includes the option to account for autocorrelation of the loss-differential by specifying an effective sample size **N.eff**.

```
R > rbind(
+   brier = ScoreDiff(EnsBrier(ens.bin, obs.bin), EnsBrier(ens.bin.ref, obs.bin)),
+   qs    = ScoreDiff(EnsQs(ens.cat, obs.cat),   EnsQs(ens.cat.ref, obs.cat)),
+   rps   = ScoreDiff(EnsRps(ens.cat, obs.cat),   EnsRps(ens.cat.ref, obs.cat)),
+   crps  = ScoreDiff(EnsCrps(ens, obs),          EnsCrps(ens.ref, obs))
+ )
```

	score.diff	score.diff.sd	p.value	ci.L	ci.U
brier	0.08500781	0.05950433	0.07655982	-0.031618542	0.2016342
qs	0.05933596	0.09682528	0.26999942	-0.130438107	0.2491100
rps	0.04769340	0.07476910	0.26177724	-0.098851334	0.1942381
crps	0.06697894	0.03125735	0.01606370	0.005715656	0.1282422

How do the score differences change if adjust the ensemble size to infinity?

```
R > rbind(
+   brier = ScoreDiff(EnsBrier(ens.bin,      obs.bin, R.new=Inf),
+                       EnsBrier(ens.bin.ref, obs.bin, R.new=Inf)),
+   qs    = ScoreDiff(EnsQs(ens.cat,      obs.cat, R.new=Inf),
+                       EnsQs(ens.cat.ref, obs.cat, R.new=Inf)),
+   rps   = ScoreDiff(EnsRps(ens.cat,      obs.cat, R.new=Inf),
+                       EnsRps(ens.cat.ref, obs.cat, R.new=Inf)),
+   crps  = ScoreDiff(EnsCrps(ens,      obs, R.new=Inf),
+                       EnsCrps(ens.ref, obs, R.new=Inf))
+ )
```

	score.diff	score.diff.sd	p.value	ci.L	ci.U
brier	0.09727310	0.05979417	0.051889887	-0.01992132	0.2144675
qs	0.08506309	0.09664353	0.189382628	-0.10435475	0.2744809
rps	0.05139946	0.07489848	0.246276159	-0.09539886	0.1981978
crps	0.07343666	0.03128020	0.009444761	0.01212859	0.1347447

3.3. Skill scores

It is common practice to compare scores of competing forecasts by a so-called skill score, which is a normalised mean score difference (Wilks 2011). Denote by S the mean score of the forecast under evaluation, by S_{ref} the mean score of a reference forecast, and by S_{perf} the mean score that would be achieved by the perfect forecaster. The skill score is then given by the score difference between the reference forecast and the evaluated forecast, normalised by the difference between the reference forecast and the perfect forecast:

$$SS = \frac{S_{ref} - S}{S_{ref} - S_{perf}} \quad (14)$$

The variance of the skill score can be estimated by error propagation as follows

$$var(SS) \approx \frac{1}{(S_{ref} - S_{perf})^2} var(S) + \frac{(S - S_{perf})^2}{(S_{ref} - S_{perf})^2} var(S_{ref}) - 2 \frac{S - S_{perf}}{(S_{ref} - S_{perf})^3} cov(S, S_{ref}) \quad (15)$$

where the variances and covariances of the mean scores are approximated by the variances and covariances of the scores, divided by the sample size. The skill score is implemented in **SpecsVerification** in the function **SkillScore**, which takes as inputs two vectors of verification scores of the evaluated and the reference forecast, the constant score achieved by a perfect forecaster, as well as a possibly user-defined effective sample size.

```
R > rbind(
+   brier = SkillScore(EnsBrier(ens.bin, obs.bin), EnsBrier(ens.bin.ref, obs.bin)),
+   qs    = SkillScore(EnsQs(ens.cat, obs.cat), EnsQs(ens.cat.ref, obs.cat)),
+   rps   = SkillScore(EnsRps(ens.cat, obs.cat), EnsRps(ens.cat.ref, obs.cat)),
+   crps  = SkillScore(EnsCrps(ens, obs), EnsCrps(ens.ref, obs))
+ )
```



```

      skillscore skillscore.sd
brier 0.32650726      0.2152194
qs     0.08498117      0.1400237
rps    0.11041350      0.1728616
crps   0.28872095      0.1182740

R > rbind(
+   brier = SkillScore(EnsBrier(ens.bin, obs.bin, R.new=Inf),
+                         EnsBrier(ens.bin.ref, obs.bin, R.new=Inf)),
+   qs    = SkillScore(EnsQs(ens.cat, obs.cat, R.new=Inf),
+                         EnsQs(ens.cat.ref, obs.cat, R.new=Inf)),
+   rps   = SkillScore(EnsRps(ens.cat, obs.cat, R.new=Inf),
+                         EnsRps(ens.cat.ref, obs.cat, R.new=Inf)),
+   crps  = SkillScore(EnsCrps(ens, obs, R.new=Inf),
+                         EnsCrps(ens.ref, obs, R.new=Inf))
+ )

      skillscore skillscore.sd
brier 0.3879870      0.2225080
qs     0.1265133      0.1458629
rps    0.1204807      0.1753339
crps   0.3287330      0.1214782

```

3.4. Correlation and correlation difference

The Pearson correlation coefficient is one of the most popular verification criteria, and can be calculated with the built-in R function `cor`. Since uncertainty quantification is often of interest, **SpecsVerification** provides the function `Corr`, which returns a correlation coefficient, a p-value and a confidence interval. The user can provide the confidence level for the confidence interval, an effective sample size to account for possible auto-correlation in the data:

```

R > ens.mean <- rowMeans(ens)
R > Corr(ens.mean, obs)

      corr      p.value      L      U
6.791888e-01 4.900893e-05 4.032555e-01 8.419058e-01

```

It is often of interest to compare the correlation coefficients between two forecasts that were issued for the same observation. The actual difference in correlation is of interest, as well as an estimation of the statistical significance of the correlation difference. **SpecsVerification** implements the function `CorrDiff` that returns the difference between the correlation of the forecast ensemble `ens` and the correlation of a reference forecast ensemble `ens.ref`, both of which were issued for the same observation `obs`. The function calculates a p-value using the test by [Steiger \(1980\)](#) and a confidence interval based on [Zou \(2007\)](#) are calculated. Both methods take into account correlation between the two competing forecasts. For illustration, we evaluate the difference in correlation between the ensemble mean forecast and the persistence forecast:

```
R > CorrDiff(ens.mean, obs.lag, obs)
```

```
corr.diff    p.value          L          U
0.1011145    0.1667036 -0.1118385  0.3532482
```

4. Rank histogram analysis for ensemble forecasts

The verification rank histogram ([Hamill 2001](#)) is a non-parametric graphical tool to assess the reliability of an ensemble forecasting system. For each pair of ensemble forecast and verifying observation, the rank of the observation among the ordered ensemble members is calculated. In a R -member ensemble, the rank is between 1 and $R + 1$. If the ensemble is a reliable representation of the uncertainty in the observation, the observation should statistically behave like “just another ensemble member”. Each verification rank should therefore be equally likely on average, and the histogram over verification ranks should be flat. **SpecsVerification** contains the function `Rankhist` to calculate the verification rank counts for an archive of ensembles and observations.

```
R > rh <- Rankhist(ens, obs)
R > rh
```

```
[1] 2 3 4 1 2 2 3 5 5
```

The function `PlotRankhist` plots the rank histogram. Two plotting modes are available: `mode="raw"` simply plots the rank counts as a bar plot histogram. `mode="prob.paper"` plots the rank counts on probability paper following [Bröcker \(2008\)](#). Assuming that each rank count has a binomial distribution with success probability $1/(R + 1)$ and sample size N . The observed rank count c_i is transformed to the cumulative probability ν_i under the Binomial distribution. To test the null-hypothesis of a flat rank histogram, 90-, 95-, and 99-percent prediction intervals are included, corrected for multiple testing. The interpretation is, given that the null-hypothesis is true, on average 9 out of 10 rank histograms should lie completely inside the 90% prediction interval.

The function `TestRankhist` implements different statistical tests of the null-hypothesis of flat rank histogram. Flatness of the rank histogram can be assessed by a Pearson χ^2 -test ([Pearson 1900](#)). Suppose rank i was observed r_i times for $i = 1, \dots, R + 1$, and define $e_i = N/(R + 1) \forall i$ the expected number of counts if each verification rank were equally likely. Define further

$$x_i = \frac{r_i - e_i}{\sqrt{e_i}}. \quad (16)$$

Under the null-hypothesis of equally likely verification ranks, the test statistic

$$\chi^2 = \sum_{i=1}^{R+1} x_i^2 \quad (17)$$

has a χ^2 -distribution with R degrees of freedom.

```
R > PlotRankhist(rh, mode = "raw")
```

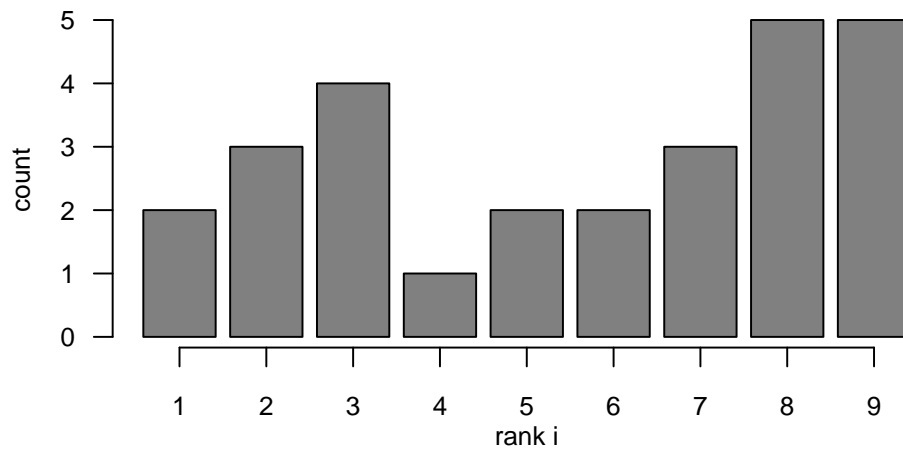


Figure 2: Rank histogram

```
R > PlotRankhist(rh, mode = "prob.paper")
```

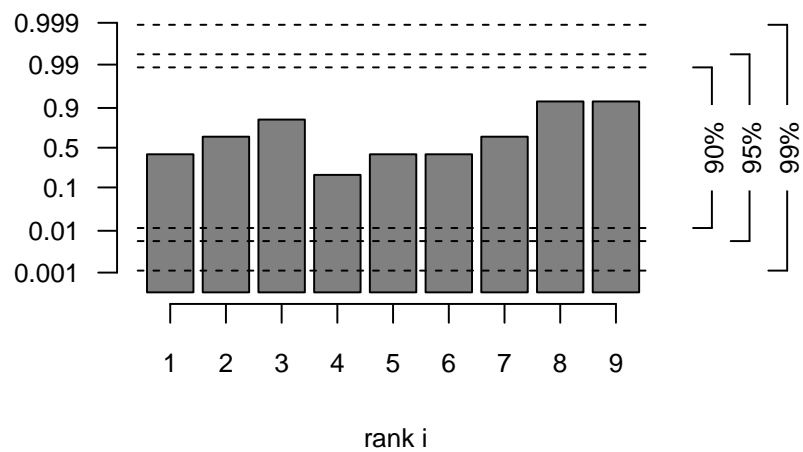


Figure 3: Rank histogram on probability paper

Hamill (2001) showed that certain types of violation of ensemble reliability are visible as different patterns in the rank histogram. In particular, a constant bias of the mean produces sloped rank histograms, and ensembles with insufficient (excessive) ensemble spread produce U-shaped (\cap -shaped) rank histogram. Jolliffe and Primo (2008) showed that the χ^2 -test statistic can be decomposed to test for sloped and convex rank histograms specifically, thus increasing the power of the test. The test requires the definition of suitable contrast vectors \mathbf{c} of length $R + 1$, that satisfy $\sum_i c_i = 0$, $\sum_i c_i^2 = 1$, and $\sum_i c_i c'_i = 0$ for every pair of contrasts \mathbf{c} and \mathbf{c}' . Assuming a number of up to R contrast vectors $\mathbf{c}^{(1)}$, $\mathbf{c}^{(2)}$, \dots , the test statistics $(\sum_i c_i^{(k)} x_i)^2$ are independently χ^2 distributed with one d.o.f. The function **TestRankhist** uses a linear and a squared contrast. Defining $J = R + 1$, the i -th element of the contrast vectors $\mathbf{c}^{(lin)}$ and $\mathbf{c}^{(sq)}$, for $i = 1, \dots, J$ are given by

$$c_i^{(lin)} = -\sqrt{\frac{3(J+1)}{J(J-1)}} + i\sqrt{\frac{12}{J^3 - J}}, \text{ and} \quad (18)$$

$$c_i^{(sq)} = -\frac{\sqrt{5}J^2 - \sqrt{5}}{\sqrt{4(J-2)(J-1)J(J+1)(J+2)}} + \left(i - \frac{J+1}{2}\right)^2 \sqrt{\frac{180}{J^5 - 5J^3 + 4J}}. \quad (19)$$

The χ^2 test using the linear contrast is sensitive to sloped rank histograms, i.e. biased ensembles, while the χ^2 based on the squared contrast is sensitive to convex rank histograms, i.e. over- or under-dispersed ensembles. **TestRankhist** returns the test-statistics and one-sided p-values of the Pearson χ^2 test, and of the two tests based on the contrasts $\mathbf{c}^{(lin)}$ and $\mathbf{c}^{(sq)}$:

`R > TestRankhist(rh)`

	pearson.chi2	jp.slope	jp.convex
test.statistic	5.3333333	1.6055556	1.3257576
p.value	0.7214269	0.2051177	0.2495614

The rank histogram of the temperature ensemble forecast provides no evidence against the null-hypothesis of a reliable ensemble.

5. Reliability diagrams for probability forecasts

The reliability diagram is a classical tool to compare probability forecasts of binary events to the verifying binary observations (Jolliffe and Stephenson 2012). The reliability diagram compares the forecast probability to the conditional frequency of the observation, given the forecast. A forecast is reliable if the forecast probability and conditional event frequency coincide. Forecast reliability is a reasonable criterion that probability forecasts should satisfy; over all instances that the forecast issued a probability p , the event should happen $p \times 100\%$ of the time.

If the forecast issues probabilities that take any value on the unit interval, most forecast probabilities will be issued only once. To estimate the conditional event frequency in this case the forecasts can be grouped into a finite number of non-overlapping bins. The average event frequency taken over all instances where the forecast is in a given bin is then taken as an average of the conditional event frequency. The reliability diagram is a plot of the conditional event frequency over the in-bin average of the forecast probabilities. **SpecsVerification** provides the

function `ReliabilityDiagram` that takes as inputs a collection of probability forecasts and binary verifying observations, and calculates the reliability diagram for a specified number of equidistant bins, or a user-defined non-equidistant binning. The consistency resampling method proposed by [Bröcker and Smith \(2007\)](#) is used to estimate the likely spread of the reliability diagrams around the diagonal if the given forecast were, in fact, reliable.

If the `plot` argument is set to `FALSE`, the `ReliabilityDiagram` function returns the quantities necessary to plot the reliability diagram.

```
R > p.bin <- rowMeans(ens.bin)
R > ReliabilityDiagram(p.bin, obs.bin, plot=FALSE, bins=3)
```

	p.avg	cond.probs	cbar.lo	cbar.hi	p.counts	bin.lower	bin.upper
1	0.1000000	0.2000000	0.0000000	0.3000000	10	0.0000000	0.3333333
2	0.5227273	0.7272727	0.2222222	0.8333333	11	0.3333333	0.6666667
3	0.8541667	1.0000000	0.5000000	1.0000000	6	0.6666667	1.0000000

If the argument `plot=TRUE` the reliability diagram is plotted, as shown in Figure 4. The logical argument `plot.refin` controls the refinement diagram, i.e. the histogram over the forecast probabilities. The logical argument `attributes` controls plotting of the polygon defined by the vertical no-resolution line at $\bar{y} = 1/n \sum_t y_t$, where y_t is the binary observation at time t , and the no-skill line defined by the linear equation $f(x) = (x + \bar{y})/2$, to produce the attributes diagram ([Hsu and Murphy 1986](#)). Points that fall into the shaded area of the attributes diagram contribute positively to forecast skill defined by the Brier skill score.

6. Additional functions

The focus of **SpecsVerification**, and of the present paper, is on verification functions for ensemble and probability forecasts, uncertainty quantification, and graphical display. **SpecsVerification** includes a number of additional functions for recalibration and verification of forecasts. For verification of deterministic forecasts, e.g. the ensemble mean forecast, the squared error (function `SqErr`) and the Absolute Error (function `AbsErr`) have been implemented.

7. Conclusion

Acknowledgments

References

- Brier GW (1950). “Verification of forecasts expressed in terms of probability.” *Monthly Weather Review*, **78**(1), 1–3. doi:10.1175/1520-0493(1950)078<0001:vofeit>2.0.co;2.

```
R > rd <- ReliabilityDiagram(p.bin, obs.bin, plot = TRUE, bins = 3,
+   attributes = TRUE)
```

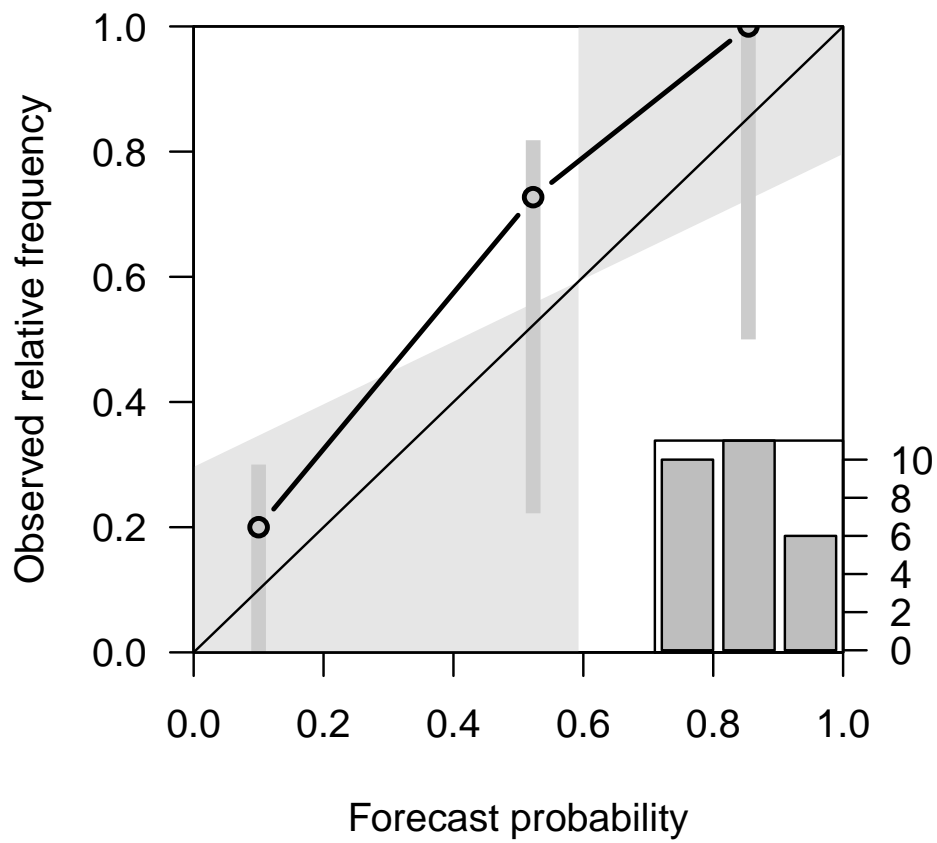


Figure 4: Reliability diagram

- Bröcker J (2008). “On reliability analysis of multi-categorical forecasts.” *Nonlinear Processes in Geophysics*, **15**(4), 661–673. ISSN 1607-7946. doi:10.5194/npg-15-661-2008. URL <http://dx.doi.org/10.5194/npg-15-661-2008>.
- Bröcker J, Smith LA (2007). “Increasing the Reliability of Reliability Diagrams.” *Weather and Forecasting*, **22**(3), 651–661. ISSN 1520-0434. doi:10.1175/waf993.1. URL <http://dx.doi.org/10.1175/WAF993.1>.
- Diebold FX, Mariano RS (1995). “Comparing Predictive Accuracy.” *Journal of Business & Economic Statistics*, **13**(3), 253. doi:10.2307/1392185.
- Ferro CAT (2013). “Fair scores for ensemble forecasts.” *Quarterly Journal of the Royal Meteorological Society*, **140**(683), 1917–1923. doi:10.1002/qj.2270.
- Ferro CAT, Richardson DS, Weigel AP (2008). “On the effect of ensemble size on the discrete and continuous ranked probability scores.” *Meteorological Applications*, **15**(1), 19–24. doi:10.1002/met.45.
- Fricker TE, Ferro CAT, Stephenson DB (2013). “Three recommendations for evaluating climate predictions.” *Meteorological Applications*, **20**(2), 246–255. doi:10.1002/met.1409.
- Gneiting T, Raftery AE (2007). “Strictly Proper Scoring Rules, Prediction, and Estimation.” *Journal of the American Statistical Association*, **102**(477), 359–378. doi:10.1198/016214506000001437.
- Hamill TM (2001). “Interpretation of Rank Histograms for Verifying Ensemble Forecasts.” *Monthly Weather Review*, **129**(3), 550–560. ISSN 1520-0493. doi:10.1175/1520-0493(2001)129<0550:iorhfv>2.0.co;2. URL [http://dx.doi.org/10.1175/1520-0493\(2001\)129<0550:IORHfV>2.0.CO;2](http://dx.doi.org/10.1175/1520-0493(2001)129<0550:IORHfV>2.0.CO;2).
- Hsu Wr, Murphy AH (1986). “The attributes diagram A geometrical framework for assessing the quality of probability forecasts.” *International Journal of Forecasting*, **2**(3), 285–293. ISSN 0169-2070. doi:10.1016/0169-2070(86)90048-8. URL [http://dx.doi.org/10.1016/0169-2070\(86\)90048-8](http://dx.doi.org/10.1016/0169-2070(86)90048-8).
- Jolliffe IT, Primo C (2008). “Evaluating Rank Histograms Using Decompositions of the Chi-Square Test Statistic.” *Monthly Weather Review*, **136**(6), 2133–2139. ISSN 1520-0493. doi:10.1175/2007mwr2219.1. URL <http://dx.doi.org/10.1175/2007MWR2219.1>.
- Jolliffe IT, Stephenson DB (2012). *Forecast verification: a practitioner’s guide in atmospheric science*. John Wiley & Sons.
- Matheson JE, Winkler RL (1976). “Scoring Rules for Continuous Probability Distributions.” *Management Science*, **22**(10), 1087–1096. doi:10.1287/mnsc.22.10.1087.
- Pearson K (1900). “X. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling.” *Philosophical Magazine Series 5*, **50**(302), 157–175. ISSN 1941-5990. doi:10.1080/14786440009463897. URL <http://dx.doi.org/10.1080/14786440009463897>.

Steiger JH (1980). “Tests for comparing elements of a correlation matrix.” *Psychological Bulletin*, **87**(2), 245–251. ISSN 0033-2909. doi:10.1037/0033-2909.87.2.245. URL <http://dx.doi.org/10.1037/0033-2909.87.2.245>.

Wilks DS (2011). *Statistical methods in the atmospheric sciences*, volume 100. Academic press.

Zou GY (2007). “Toward using confidence intervals to compare correlations.” *Psychological Methods*, **12**(4), 399–413. ISSN 1082-989X. doi:10.1037/1082-989X.12.4.399. URL <http://dx.doi.org/10.1037/1082-989X.12.4.399>.

Affiliation:

Stefan Siegert
Exeter Climate Systems
College for Engineering, Mathematics, and Physical Sciences
University of Exeter
Exeter, EX4 4QF, United Kingdom
E-mail: Stefan.Siegert@exeter.ac.uk
URL: <http://emps.exeter.ac.uk/mathematics/staff/ss610>