Forecast Verification: Ensemble-adjusted scoring rules, graphical tools, comparative verification, and uncertainty assessment implemented in the `R` package `SpecsVerification`

Stefan Siegert

September 19, 2016

Abstract

Forecast verification, the retrospective comparison of forecasts to observations, is a common task at institutions that routinely issue forecasts to the public, such as climate centers. Ensemble forecasts, i.e., multiple forecasts initialised from perturbed initial conditions, are routinely issued to generate probabilistic forecasts. In recent years, advances have been made in statistical methodology to verify ensemble forecast, in particular to estimate the finite-ensemble effect on verification scores. This paper summarises statistical methodology to account for finite-ensemble effects in ensemble verification. Forecast verification analyses should be accompanied by an assessment of the sampling uncertainty in the results. This paper summarises existing methods to assess sampling uncertainty for a large number of verification metrics, including scoring rules, correlation coefficients, the area under the ROC curve, the rank histogram, the reliability diagram, and the Reliability-Resolution-Uncertainty decomposition of the Brier score. Forecast verification is often more meaningful by comparing competing forecasting systems for the same observation. A number of methods for uncertainty assessment for comparative verification are summarised in this paper as well. All methods presented are implemented in the `R` package `SpecsVerification`.

1 Introduction

An ensemble forecast is a collection of forecasts for the same target. The ensemble members usually differ due to differences in initial conditions, boundary conditions, model physics and background information [Gneiting, 2005, Leutbecher and Palmer, 2008]. Ensemble forecasting is today operationally used in weather and climate forecasting to explore the chaotic divergence of nonlinear systems, and to estimate forecast uncertainty. To evaluate the reliability and predictive skill of ensemble forecasts, the forecasts have to be compared to their verifying observations of the real world. The comparison of forecasts with their verifying observations is commonly referred to as forecast verification [Jolliffe and Stephenson, 2012]. A verification measure is thus a function that depends on an archive $D = \{x_t, y_t\}_{t=1}^N$ of past forecasts x_t and verifying observations y_t. The archive D is also called a hindcast data set. A variety of verification measures exists to assess the quality of different forecast products, such as deterministic forecasts, probabilistic forecasts, ensemble forecasts, univariate or multivariate forecasts, and gridded or unevenly spaced spatial forecasts.

The `R` package `SpecsVerification` documented in this paper implements specialised verification scores for ensemble forecasts. Buizza and Palmer [1998] and others have shown that verification metrics can depend systematically on the number of ensemble members. Everything else being equal, ensembles with many members achieve better verification results than ensembles with fewer members. This finding is not surprising, because larger ensembles allow for a more comprehensive exploration of forecast uncertainty, and thus more robust probabilistic forecasts. However, hindcast experiments with state-of-the-art climate models are computationally expensive, and therefore new model configurations are often tested with small ensemble sizes first. Large ensembles are only generated in forecast mode, after developers have agreed on the best model configuration. But if measures of forecast skill depend on the ensemble size, the skill calculated for the hindcast ensemble with few members is not representative of the skill that an operational ensemble with more members will likely achieve. A number of verification scores have been proposed in the past to estimate, and correct, this finite ensemble effect. An important contribution of this paper is to summarise existing ensemble verification scores, and to document their implementation in the `R` statistical programming environment [R Core Team, 2015].

Forecast quality can hardly be defined in an absolute sense. Verification results are easier to interpret if verification measures are compared between competing forecasting systems for the same observation. `SpecsVerification` contains a number of functions to enable comparative verification between competing forecasts. Moreover, the value of a verification metric is of little use without an estimate of its sampling uncertainty [Jolliffe, 2007]. Uncertainty assessment is all the more important if the sample size of the hindcast data set is small, as is often the case in climate science. All verification functions implemented in `SpecsVerification` come with built-in uncertainty assessments. Systematically reviewing and implementing these methods is another important contribution of this paper.

There are packages for forecast verification available for download from the Comprehensive `R` Archive Network (CRAN). The most notable package is `verification` [NCAR - Research Applications Laboratory, 2015]. While `verification` contains a limited number of functions for ensemble verification, comparative verification and uncertainty assessment, it does not have the same focus on these topics as does `SpecsVerification`. Since the two packages do not share the same functions, `SpecsVerification` should be regarded as complementary to `verification`. Further recently published `R` packages for forecast verification include `easyverification` [Bhend et al., 2016] and `s2dverification` [Guemas et al., 2016], both of which borrow functions from `SpecsVerification`.

2 Seasonal climate forecast and observation data

`SpecsVerification` includes the data set `eurotempforecast` of seasonal temperature ensemble forecasts and verifying observations. The forecasts 24-member ensembles of near-surface air temperatures produced by the NCEP climate forecast system version 2 [Saha et al., 2014], initialised between 11 April and 6 May each year from 1983–2009 ($N = 27$), and averaged over the region limited by latitudes 30°N – 75°N and longitudes 12.5°W – 42.5°E, and over the months June-July-August, i.e., the lead time is about 1–3 months. Data from the NCEP climate forecast system reanalysis [Saha et al., 2010] was taken as verifying observations. All data was downloaded through the ECOMS user data gateway R-interface [Santander Meteorology Group, 2015]. Ensemble members and observation data are plotted as time series in Figure 1.

```
data(eurotempforecast)
R   <- ncol(ens)
yrs <- as.numeric(names(obs))
N   <- length(obs)
```

All functions in `SpecsVerification` that analyse ensemble forecast data, assume that archives of N instances of R-member ensemble forecasts, are represented by $N \times R$ matrices. The data shown in Figure 1 depicts 27 years or 24-member ensemble forecasts, and is thus a `R` matrix with 27 rows and 24 columns.

Table 1: Overview of verification functions implemented in `SpecsVerification`, and their application.

Function	Long name	Applies to
`AbsErr`	Absolute error Score*	Deterministic forecasts of continuous observations
`Auc`	Area under the ROC curve	Probability forecasts of binary observations
`AucDiff`	Difference between two areas under the ROC curve	Two competing probability forecasts for the same binary observations
`BrierDecomp`	Brier score decomposition into Reliability, Resolution, Uncertainty	Probability forecasts of binary observations
`Corr`	Correlation coefficient	Deterministic forecasts of continuous observations
`CorrDiff`	Difference between two correlation coefficients	Two competing deterministic forecasts for the same continuous observations
`DressCrps`	Continuous ranked probability score for dressed ensembles*	Ensemble forecasts of continuous observations
`DressIgn`	Ignorance score for dressed ensembles*	Ensemble forecasts of continuous observations
`EnsBrier`	Ensemble-adjusted Brier score*	Ensemble forecast of binary observations
`EnsCrps`	Ensemble-adjusted continuous ranked probability score*	Ensemble forecasts of continuous observations
`EnsRps`	Ensemble-adjusted ranked probability score*	Ensemble forecasts of categorical observations
`EnsQs`	Ensemble-adjusted quadratic score*	Ensemble forecasts of categorical observations
`GaussCrps`	Continuous ranked probability score for Normal distributions*	Probability forecasts of continuous observations
`PlotRankhist`	Plot a rank histogram	see `Rankhist`
`Rankhist`	Calculate a rank histogram	Ensemble forecasts of continuous observations
`ReliabilityDiagram`	Calculate and plot a reliability diagram	Probability forecasts of binary observations
`ScoreDiff`	Calculate a score difference and assess uncertainty	All scores marked with a *
`SkillScore`	Calculate a skill score and assess uncertainty	All scores marked with a *
`TestRankhist`	Statistical tests of a rank histogram	see `Rankhist`
`SqErr`	Squared error score*	Deterministic forecasts of continuous observations

```r
par(las=1, cex=0.7, mgp=c(3, 1, 0), mar=c(2,4,1,1))
matplot(yrs, ens, ylab="temp [C]", pch=1, col=gray(.5))
points(yrs, obs, pch=15, cex=1.5)
```

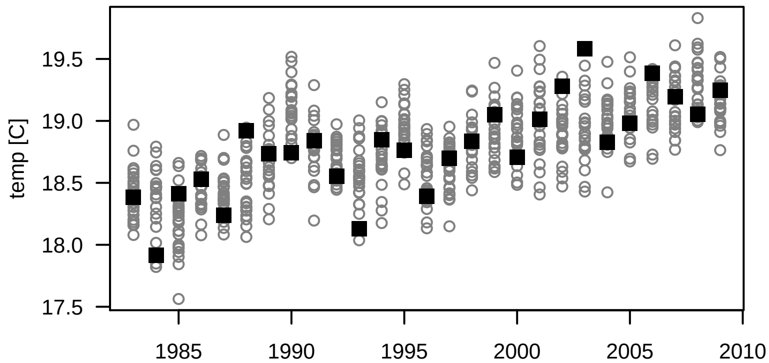

Figure 1: Seasonal European temperature ensemble forecasts (circles) and verifying observations (squares).

Real-valued forecasts and real-valued observations are saved as `R` matrix `ens` and `R` vector `obs`. The continuous data in `ens` and `obs` was used to derive binary and categorical forecasts and observations (matrices `ens.bin` and `ens.cat`, and vectors `obs.bin` and `obs.cat`). The binary forecasts, were generated by asking "Will this year's summer be warmer than last year's"? The binary observation at time index t is equal to one if the temperature in the corresponding year exceeds the temperature of the previous year, and zero otherwise. The individual ensemble members were transformed equivalently, by comparing the rth ensemble member in year t to the observation in year $t-1$. Categorical forecasts and observations were generated by asking "Will this year's temperature be similar to last year's temperature (within a 0.5K range), colder, or warmer?" Ensemble members and the observation thus fall into one of 3 categories: Category 1 if the temperature that is less than $0.25K$ colder than the previous year, category 2 if the temperature is within $\pm 0.25K$ of the previous year, and category 3 if it is more than $0.25K$ warmer. In addition, the vector `obs.lag` is provided, containing 1-year lagged observed temperatures. For example, continuous, binary and categorical observations and forecasts for the year 2001 are

```r
rbind(
continuous = c(obs=obs["2001"], ens["2001", 1:5]),
binary     = c(obs=obs.bin["2001"], ens.bin["2001", 1:5]),
categorical = c(obs=obs.cat["2001"], ens.cat["2001", 1:5]))

##             obs.2001 Member_1 Member_2 Member_3 Member_4 Member_5
## continuous     19.01    18.65    18.96    19.23    18.41    18.81
## binary          1.00     0.00     1.00     1.00     0.00     1.00
## categorical     3.00     2.00     3.00     3.00     1.00     2.00
```

3 Ensemble-adjusted verification scores

3.1 Binary ensemble forecasts

This subsection outlines the theory behind ensemble-adjusted verification scores, using probabilistic forecasts of binary events. 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 (non-occurrence) of the event is coded as $y_t = 1$ ($y_t = 0$). The Brier score is given by the squared difference between forecast and observation:

$$s_B(p_t, y_t) = (p_t - y_t)^2 \tag{1}$$

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].

Suppose that the probability p_t is unknown, but an ensemble forecast of size R drawn from p_t is available. Each of the R ensemble members is interpreted as an independent Bernoulli trial with success probability p_t. An unbiased estimator of p_t is given by the fraction i_t/R, where i_t is the number of ensemble members that predict the event $y_t = 1$. The Brier score of the probability forecast i_t/R is equal to

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

and taking expectation over the random variable $i_t \sim Binomial(p_t, R)$, it is shown that [Ferro et al., 2008]

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

That is, the Brier score of i_t/R is a biased estimator of the Brier score of p_t. The expected Brier score of i_t/R is larger (i.e. worse) than 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 $R \to \infty$. 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) expected Brier score. This is reasonable since more ensemble members allow for more robust estimation of the true probability p_t. But it is sometimes desirable to correct the finite-ensemble bias. Suppose, for example, a hindcast ensemble has R members, but future operational forecasts will be made with $R^* > R$ ensemble members, using the same ensemble system. Due to the finite-ensemble bias, the score calculated for the R-member hindcast ensemble will be a too pessimistic estimate of the expected score of the R^*-member forecast ensemble. An adjustment is desirable that allows to use an R member ensemble to unbiasedly estimate the score of an $R^* \neq R$ member ensemble.

The ensemble-adjusted Brier score, given by [Ferro et al., 2008]

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

includes a correction of 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, whose members are sampled from the same probability p_t, i.e.,

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

Setting $R^* = \infty$ yields the fair Brier score [Ferro, 2013] which is an unbiased estimator of the Brier score of the underlying (unknown) probability p_t. The ensemble-adjusted Brier score can be used to compare ensemble forecasting systems (e.g. from different climate centers) that use different ensemble sizes. The score further allows for the extrapolation of the average score of an ensemble forecast system to larger ensemble sizes, e.g., to estimate forecast skill with R^* members based on hindcast skill with $R < R^*$ members. 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`.

To illustrate the finite ensemble effect, we randomly split the 24-member forecast ensemble `ens` into a small 5-member subensemble, and a larger 19-member subensemble, and calculate their unadjusted Brier scores (Equation 2):

```
i.small <- sample(1:R, 5)
i.large <- setdiff(1:R, i.small)
c(small.ens=mean(EnsBrier(ens.bin[, i.small], obs.bin)),
  large.ens=mean(EnsBrier(ens.bin[, i.large], obs.bin)))

## small.ens large.ens
##    0.1689    0.1465
```

The large subensemble obtains a better average Brier score than the small subensemble, even though both ensembles were produced by the same ensemble forecasting system. We next adjust the Brier score for the finite ensemble size, by calculating s_B^* using $R^* = 19$ for both ensembles:

```
c(small.ens=mean(EnsBrier(ens.bin[, i.small], obs.bin, R.new=19)),
  large.ens=mean(EnsBrier(ens.bin[, i.large], obs.bin, R.new=19)))

## small.ens large.ens
##    0.1454    0.1465
```

After adjusting for the finite-ensemble bias, the scores are very similar. Using the ensemble-adjusted Brier score, we were able to extrapolate the Brier score of a 5-member ensemble to the Brier score that a 19-member ensemble would achieve, produced by the same ensemble forecasting system.

3.2 Categorical ensemble forecasts

Suppose a forecast target that always falls into exactly one out of K disjoint categories. A probabilistic forecast for this target is the vector $\mathbf{p}_t = (p_{t,1}, \cdots, p_{t,K})$, whose kth element equals the forecast probability that the observation will materialise in class k. The verifying observation is vector-valued $\mathbf{y}_t$, where the kth element of $\mathbf{y}_t$ is $y_{t,k} = 1$ if the kth class is observed, and $y_{t,j} = 0$ for all $j \neq k$. The quadratic score (QS) of the probability forecast $\mathbf{p}_t$ with verifying observation $\mathbf{y}_t$ is given by

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

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 kth category, for $k = 1, \cdots, K$. Using results obtained for the ensemble-adjusted Brier score, [see also Ferro et al., 2008], the ensemble-adjusted QS is seen to be

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

The ensemble-adjusted QS is implemented as the function `EnsQs` in `SpecsVerification`. The function expects the argument `ens` to be provided as a $N \times R$ matrix of categorical ensemble forecasts, where `ens[t,r]` indicates the category predicted by the rth ensemble members at time t.

The QS is invariant under relabelling of the K categories, and therefore insensitive to distance. If the observation materialises in class 1, say, the QS penalises a forecast that puts all probability mass into class 2 as much as a forecast that puts all probability mass into class 3. In forecast forecast problems where the categories convey a natural ordering of the forecast target, such as {1="no rain", 2="light rain"; 3="heavy rain"}, a forecast that predicts class 2 might seem preferable to a forecast that predicts class 3, if class 1 occurs.

The ranked probability score (RPS) is a version of the QS that is sensitive to distance. The ensemble forecast vector $\mathbf{i}_t$ is transformed to the cumulated forecast vector $\mathbf{j}_t$, with kth element equal to $j_{t,k} = \sum_{l=1}^{k} i_{t,l}$, and the cumulated observation vector $\mathbf{z}_t$ has $z_{t,k} = \sum_{l=1}^{k} y_{t,l}$. The RPS is the QS of the forecast $\mathbf{j}_t$ evaluated on the observation $\mathbf{z}_t$. Accumulating the elements of $\mathbf{i}_t$ and $\mathbf{y}_t$ nests the K forecast categories within each other. Essentially, 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*". Nesting of forecast categories enables order-sensitivity of the score. Using results from the previous section, we get the ensemble-adjusted RPS

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

The ensemble-adjusted RPS is implemented as the function `EnsRps` in `SpecsVerification`. The argument `ens` should be provided as for `EnsQs`; the nesting of forecast categories is performed internally.

To illustrate the finite-ensemble effect and its adjustment in the QS and RPS, we split the ensemble randomly into a 5-member ensemble and a 19-member ensemble. We evaluate the unadjusted QS of the full 24-member ensemble, the unadjusted scores of the small and large subensembles, as well as the QS of the two subensembles adjusted for the size $R^* = 24$ of the full ensemble:

```
i.small <- sample(1:R, 5)
i.large <- setdiff(1:R, i.small)
cbind(
 QS = c(
  ens           = mean(EnsQs(ens.cat,            obs.cat)),
  small.ens     = mean(EnsQs(ens.cat[, i.small], obs.cat)),
  large.ens     = mean(EnsQs(ens.cat[, i.large], obs.cat)),
  small.ens.adj = mean(EnsQs(ens.cat[, i.small], obs.cat, R.new=24)),
  large.ens.adj = mean(EnsQs(ens.cat[, i.large], obs.cat, R.new=24))
 ),
 RPS = c(
  ens           = mean(EnsRps(ens.cat,            obs.cat)),
  small.ens     = mean(EnsRps(ens.cat[, i.small], obs.cat)),
  large.ens     = mean(EnsRps(ens.cat[, i.large], obs.cat)),
  small.ens.adj = mean(EnsRps(ens.cat[, i.small], obs.cat, R.new=24)),
  large.ens.adj = mean(EnsRps(ens.cat[, i.large], obs.cat, R.new=24))
 )
)

##                     QS    RPS
## ens             0.5782 0.3344
```

```
## small.ens     0.6519 0.3719
## large.ens     0.5856 0.3386
## small.ens.adj 0.5815 0.3355
## large.ens.adj 0.5810 0.3362
```

The full-ensemble obtains a better score than either of the sub-ensembles and the large subensemble outperforms the small subensemble. When adjusting scores of the subensembles to $R^* = 24$, the scores of the subensembles are more similar to each other, and closer to the score of the full ensemble. Note that the equality of ensemble-adjusted scores for different ensemble sizes holds only in expectation. Empirical averages over finite numbers of hindcasts differ, which explains the differences in the previous analyses. The reader is referred to Ferro [2013] for illustrations of convergence by random number experiments.

3.3 Continuous ensemble forecasts

If the forecast target is a continuous variable, such as temperature or pressure, the continuous ranked probability score [CRPS; 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 (cdf) $F_t(x)$, the CRPS is given by

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

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 $x_t = \{x_{t,1}, x_{t,2} \ldots, x_{t,R}\}$ 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

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

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

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

Fricker et al. [2013] show that the CRPS is sensitive to the ensemble size, and propose the ensemble-adjusted CRPS

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

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 [Fricker et al., 2013]. The ensemble-adjusted CRPS is implemented in the `SpecsVerification` function `EnsCrps`. We calculate the unadjusted CRPS ($R^* = 24$) and the fair CRPS ($R^* = \infty$):

```
cbind(
 CRPS=c(
  unadjusted = mean(EnsCrps(ens, obs)),
  fair       = mean(EnsCrps(ens, obs, R.new=Inf))
 )
)
```

```
##               CRPS
## unadjusted 0.1381
## fair       0.1329
```

4 Comparative verification and uncertainty assessment

4.1 Reference forecast

The value of a verification score by itself is often not easily interpretable. To evaluate the usefulness (or "skill") of a forecast, its verification score is compared to the score achieved by a suitable reference forecast. For example, the score of a state-of-the-art high resolution climate model should be compared to the score achieved by an older climate model version with lower resolution and less physical detail. In the absence of a reference forecast generated by a dynamical climate model, a simple statistical benchmark prediction can be used. A popular statistical reference forecast is the climatological forecast, which is only based on the known record of observations. To benchmark ensemble forecasts in particular, the climatological reference forecast can be generated by sampling randomly from the record of known observations, or by treating all previously available observations as an exchangeable ensemble forecast drawn from the climatological distribution. `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 tth observation on the tth forecast instance:

```
ens.ref     <- ClimEns(obs)
ens.cat.ref <- ClimEns(obs.cat)
ens.bin.ref <- ClimEns(obs.bin)
```

In addition to the climatological forecast, it is advisable to also consider statistical reference forecasts such as a linear trend or an auto-regressive model, which are often more suitable than the climatological forecast.

4.2 Score differences

Suppose we have calculated two sets of N verification scores, $\{s_1^{(1)}, s_2^{(1)}, \ldots, s_N^{(1)}\}$ for forecast 1, and $\{s_1^{(2)}, s_2^{(2)}, \ldots, s_N^{(2)}\}$ for forecast 2, using the same set of observation. Diebold and Mariano [1995] suggest to test the null-hypothesis of equal forecast accuracy using the time series $d_1, \ldots, d_N$ of loss differentials, $d_t = s_t^{(1)} - s_t^{(2)}$. Define $\bar{d}$ to be the empirical average over $d_1, \cdots, d_N$. Under the assumption of temporal independence of successive d_t, the test statistic

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

is asymptotically Normally distributed with mean zero and variance one, if the population score difference is zero. The test is implemented in `SpecsVerification` in the function `ScoreDiff`. The function includes the option to account for temporal dependency of the loss-differential by specifying an effective sample size `N.eff`.

```
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        L      U
## Brier    0.10292       0.04156 0.0066342  0.02147 0.1844
## QS       0.06927       0.09024 0.2213483 -0.10760 0.2461
## RPS      0.06612       0.06380 0.1500227 -0.05893 0.1912
## CRPS     0.07705       0.02233 0.0002801  0.03328 0.1208
```

4.3 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 (often we have $S_{\mathrm{perf}} = 0$). The skill score is then given by the average score difference between the reference forecast and the evaluated forecast, normalised by the average score difference between the reference forecast and the perfect forecast:

$$SS = \frac{S_{\mathrm{ref}} - S}{S_{\mathrm{ref}} - S_{\mathrm{perf}}} \tag{14}$$

The variance of the skill score can be estimated by error propagation (also known as the delta-method) as follows:

$$\begin{aligned} var(SS) \approx & \frac{1}{(S_{\mathrm{ref}} - S_{\mathrm{perf}})^2} var(S) + \frac{(S - S_{\mathrm{perf}})^2}{(S_{\mathrm{ref}} - S_{\mathrm{perf}})^2} var(S_{\mathrm{ref}}) \\ & - 2 \frac{S - S_{\mathrm{perf}}}{(S_{\mathrm{ref}} - S_{\mathrm{perf}})^3} cov(S, S_{\mathrm{ref}}) \end{aligned} \tag{15}$$

where the variances and covariances of the mean scores S and S_{ref} are approximated by the variances and covariances calculated for the individual scores, divided by the sample size. Calculation of skill scores and their approximate standard deviation is implemented in `SpecsVerification` in the function `SkillScore`, which takes as inputs two series of verification scores, as well as a possibly user-defined effective sample size.

```
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     0.4263       0.16147
## QS        0.1070       0.14064
## RPS       0.1651       0.15771
## CRPS      0.3582       0.07919
```

4.4 Correlation and correlation difference

The Pearson (product-moment) correlation coefficient is one of the most popular verification criteria, and can easily be calculated with the built-in `R` function `cor`. Since uncertainty assessment is often of interest in forecast verification, `SpecsVerification` provides the function `Corr`, which returns the sample correlation coefficient r_{xy}, a one-sided p-value, and a confidence interval based on standard methods presented in, e.g., Von Storch and Zwiers [2001]. The p-value is calculated based on the test statistic

$$T_{\mathrm{cor}} = \sqrt{(N-2)\frac{r_{xy}^2}{1-r_{xy}^2}} \tag{16}$$

which has a Student's t-distribution with $N-2$ degrees of freedom under the null-hypothesis of zero population correlation. The $(1-\alpha)\times 100\%$ confidence interval $[L,U]$ is calculated by

$$[L,U] = \left[\tanh\left(z_{xy} + \frac{Z_{\alpha/2}}{\sqrt{N-3}}\right), \tanh\left(z_{xy} + \frac{Z_{1-\alpha/2}}{\sqrt{N-3}}\right)\right] \tag{17}$$

where $z_{xy} = \mathrm{atanh}(r_{xy})$ is the Fisher transformation, and Z_p is the p-quantile of the standard Normal distribution.

```
ens.mean <- rowMeans(ens)
Corr(ens.mean, obs, conf.level=0.95)

##        corr     p.value           L           U
## 0.757095576 0.000002427 0.529391069 0.883049977
```

It is often of interest to compare the skill of two forecasts by calculating their correlation difference. `SpecsVerification` implements the function `CorrDiff` that returns the difference between the correlation r_{by} of the forecast B and the correlation r_{ay} of a reference forecast A, both of which were issued for the same observation Y. The function calculates a one-sided p-value, using the test for differences between overlapping correlation coefficients by Steiger [1980]: Denote by r_{ab} the correlation between forecast A and forecast B. The determinant of the sample correlation matrix of the two forecasts and the observation is calculated:

$$R = (1 - r_{ay}^2 - r_{by}^2 - r_{ab}^2) + (2r_{ay}r_{by}r_{ab}) \tag{18}$$

The test statistic

$$T_{\mathrm{cordiff}} = (r_{by} - r_{ay})\sqrt{\frac{(N-1)(1+r_{ab})}{2\left(\frac{N-1}{N-3}\right)R + \frac{1}{4}(r_{ay}+r_{by})^2(1-r_{ab})^3}} \tag{19}$$

has a Student's t-distribution with $N-3$ degrees of freedom under the null-hypothesis that A and B have equal correlations with Y. Furthermore, a confidence interval for the correlation difference $r_{by} - r_{ay}$ is calculated, based on Zou [2007]. We first estimate the correlation between the correlation coefficients r_{ay} and r_{by} by

$$c_{ab} = \frac{\left(r_{ab} - \frac{1}{2} r_{ay} r_{by}\right)\left(1 - r_{ay}^2 - r_{by}^2 - r_{ab}^2\right) + r_{ab}^3}{(1 - r_{ay}^2)(1 - r_{by}^2)}. \tag{20}$$

We then calculate $(1-\alpha)\times 100\%$ confidence intervals (l_a, u_a) for r_{ay} and (l_b, u_b) for r_{by}, using the Fisher transformation as in Equation 17. An approximate $(1-\alpha)\times 100\%$ confidence interval (L, U) for the correlation difference $r_{by} - r_{ay}$ is then given by

$$\begin{aligned} L &= (r_{by} - r_{ay}) - \sqrt{(r_{by} - l_b)^2 + (u_a - r_{ay})^2 - 2c_{ab}(r_{by} - l_b)(u_a - r_{ay})}, \\ U &= (r_{by} - r_{ay}) + \sqrt{(u_b - r_{by})^2 + (r_{ay} - l_a)^2 - 2c_{ab}(u_b - r_{by})(r_{ay} - l_a)}. \end{aligned} \tag{21}$$

For illustration, we evaluate the difference in correlation between the ensemble mean forecast and the persistence forecast, i.e., the observation at one year lag:

```
CorrDiff(fcst=ens.mean, fcst.ref=obs.lag, obs=obs, conf.level=0.95)

## corr.diff   p.value         L         U
##  0.179021  0.029082 -0.005417  0.440518
```

The one-sided p-value is small and the value of zero correlation is close to the boundary of the 95% confidence interval, which provides ample evidence that the ensemble mean seasonal temperature forecast has higher correlation skill than the persistence forecast.

4.5 Area under the curve (AUC) and AUC differences

Relative operating characteristics [ROC, Mason and Graham, 2002] analysis is a method from signal detection theory to evaluate the quality of forecasts for binary events. Consider the two competing forecasts $x_t^{(1)}$ and $x_t^{(2)}$ for the same binary observations $y_t \in \{0, 1\}$ for $t = 1, \ldots, N$. (The forecasts are allowed to take values on the real line, and need not be probabilitities.) For ROC analysis, the forecasts are grouped into two sets: $C_0^{(r)}$ is the set of all forecasts $x^{(r)}$ for which an event did not happen ($y_t = 0$), and $C_1^{(r)}$ is the set of all forecasts for which an event did happen ($y_t = 1$). The area under the ROC curve (AUC) for the rth forecast ($r = 1, 2$) is equal to the probability that a randomly drawn forecast from $C_1^{(r)}$ is larger than a randomly drawn forecast from $C_0^{(r)}$. The AUC is thus a measure of the ability of the forecast system to distinguish events from non-events.

DeLong et al. [1988] suggest a nonparametric method to estimate the variance of AUC, and of differences in AUC. Denote by $X_i^{(r)}$, $i = 1, \cdots, m$ the elements of $C_1^{(r)}$ and by $Y_i^{(r)}$, $i = 1, \cdots, n$ the members of $C_0^{(r)}$. Define the function Ψ as

$$\Psi(x, y) = \mathbb{1}(x > y) + \frac{1}{2}\mathbb{1}(x = y) \tag{22}$$

where $\mathbb{1}(\cdot)$ is the indicator function which equals one if its argument is true, and zero otherwise. The AUC of the rth forecast is estimated by

$$\hat{\theta}^{(r)} = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \Psi\left(X_i^{(r)}, Y_j^{(r)}\right). \tag{23}$$

For variance estimation, first define the quantities $V_i^{(r)}$ and $W_i^{(r)}$ by

$$V_i^{(r)} = \frac{1}{n}\sum_{j=1}^{n}\Psi\left(X_i^{(r)}, Y_j^{(r)}\right) \quad \text{and} \quad W_j^{(r)} = \frac{1}{m}\sum_{i=m}^{n}\Psi\left(X_i^{(r)}, Y_j^{(r)}\right), \tag{24}$$

and $v_{r,s}$ and $w_{r,s}$ by

$$\begin{aligned} v_{r,s} &= \frac{1}{m-1}\sum_{i=1}^{m}\left[V_i^{(r)} - \hat{\theta}^{(r)}\right]\left[V_i^{(s)} - \hat{\theta}^{(s)}\right] \\ w_{r,s} &= \frac{1}{n-1}\sum_{j=1}^{n}\left[W_j^{(r)} - \hat{\theta}^{(r)}\right]\left[W_j^{(s)} - \hat{\theta}^{(s)}\right] \end{aligned} \tag{25}$$

where $r = 1, 2$ and $s = 1, 2$. Finally, the estimated variance of the rth AUC estimate $\hat{\theta}^{(r)}$ is given by

$$var\left(\hat{\theta}^{(r)}\right) = \frac{1}{m}v_{r,r} + \frac{1}{n}w_{r,r} \tag{26}$$

and the variance of the AUC difference $\theta^{(2)} - \theta^{(1)}$ is approximated by

$$var\left(\hat{\theta}^{(2)} - \hat{\theta}^{(1)}\right) = \frac{1}{m}(v_{1,1} + v_{2,2} - 2v_{1,2}) + \frac{1}{n}(w_{1,1} + w_{2,2} - 2w_{1,2}). \tag{27}$$

Note that the AUC is asymptotically Normally distributed. The estimated variance can therefore be used to construct a confidence interval, i.e., $\hat{\theta} \pm 1.96\sqrt{var(\hat{\theta})}$ is a central 95% confidence interval.

`SpecsVerification` provides the functions `Auc` and `AucDiff` that implement calculation of AUC and AUC differences and the corresponding variance estimates. The following calculates AUCs for the ensemble mean of the binary ensemble, using a large and a small subensemble:

```
rbind(
  large.ens = Auc(rowMeans(ens.bin[, i.large]), obs.bin),
  small.ens = Auc(rowMeans(ens.bin[, i.small]), obs.bin)
)

##              auc  auc.sd
## large.ens 0.8892 0.06944
## small.ens 0.8523 0.07179
```

The AUC of both ensembles is significantly larger than 0.5, which is a sign of forecast skill. The large subensemble has a slightly higher AUC than the small subensemble. The following evaluates the AUC difference between the large and a small ensemble:

```
AucDiff(rowMeans(ens.bin[, i.large]), rowMeans(ens.bin[, i.small]), obs.bin)

##    auc.diff auc.diff.sd
##     0.03693     0.06346
```

The AUC difference between the large and small ensemble is well within sampling variability, so we remain uncertain about whether using a larger ensemble improves the AUC. In general, we expect ensemble size to effect AUC, but we are not aware of any ensemble-adjustments for the AUC.

5 Rank histogram analysis for ensemble forecasts

The verification rank histogram [Talagrand et al., 1997, 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 reliable, i.e., statistically exchangeable with the verifying observation, the observation should 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 histogram counts for an archive of ensembles and observations:

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

##  [1] 0 2 1 0 2 4 1 1 0 0 0 0 1 2 2 1 3 1 1 0 1 1 0 2 1
```

The function `PlotRankhist` plots the rank histogram. Two plotting modes are available: The option `mode="raw"` plots the rank histogram counts as a simple bar plot, and `mode="prob.paper"` plots the rank counts on probability paper following Bröcker [2008], as follows. Assuming a reliable ensemble, each rank count should be approximately Binomially distributed with success probability $1/(R+1)$ and sample size N. To assess the plausibility of individual rank counts, each 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. If the null-hypothesis of a reliable ensemble is true, on average 9 out of 10 rank histograms are expected to lie entirely inside the 90% prediction interval. A rank histogram on probability paper is shown in Figure 2.

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

$$q_i = \frac{r_i - e_i}{\sqrt{e_i}}. \tag{28}$$

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

$$\chi^2 = \sum_{i=1}^{R+1} q_i^2 \tag{29}$$

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

Hamill [2001] showed that certain types of violation of ensemble reliability are visible as different patterns in the rank histogram. In particular, a systematic bias of the ensemble mean produces sloped rank histograms, and ensembles with insufficient (excessive) ensemble spread produce ∪-shaped (∩-shaped) rank histograms. 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 χ^2-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 $\mathbf{c}^{(1)}$, $\mathbf{c}^{(2)}$, $\ldots$, the test statistics $(\sum_i c_i^{(k)} q_i)^2$ are independently χ^2-distributed with one d.o.f. The function `TestRankhist` applies this test, using a linear and a squared contrast. Defining $J = R+1$, the ith element of the contrast vectors $\mathbf{c}^{(lin)}$ and

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

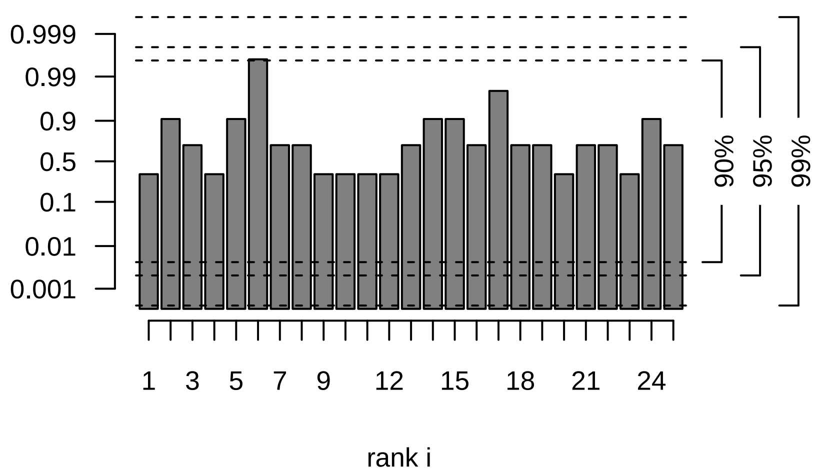

Figure 2: Rank histogram on probability paper

$\mathbf{c}^{(sq)}$, for $i = 1, \cdots, J$ are given by

$$c_i^{(lin)} = -\sqrt{\frac{3(J+1)}{J(J-1)}} + i\sqrt{\frac{12}{J^3 - J}}, \text{ and} \tag{30}$$

$$c_i^{(sq)} = -\frac{\sqrt{5}J^2 - \sqrt{5}}{\sqrt{4(J-2)(J-1)J(J+1)(J+2)}} + \left(i - \frac{J+1}{2}\right)^2 \sqrt{\frac{180}{J^5 - 5J^3 + 4J}}. \tag{31}$$

The χ^2-test using the linear contrast is sensitive to sloped rank histgrams, i.e. biased ensembles, while the χ^2-test using the squared contrast is sensitive to convex rank histograms, i.e. over- or under-dispersed ensemble forecasts. `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 $\mathbf{c}^{(sq)}$:

```
TestRankhist(rh)

##                pearson.chi2 jp.lin    jp.sq
## test.statistic      23.9259 0.0114 0.005574
## p.value              0.4658 0.9150 0.940485
```

All p-values are considerably larger than zero. The rank histogram of the temperature ensemble forecast provides no evidence against the null-hypothesis of a reliable ensemble. However, due to the small sample size, all tests are expected to have low power.

6 Reliability diagram and Brier Score decomposition

The reliability diagram is a classical tool to compare probability forecasts of binary events to the verifying binary observations [Wilks, 2011, Jolliffe and Stephenson, 2012]. The reliability diagram compares the issued forecast probabilities to the conditional average frequencies of the observation, given the forecast. A probability forecast is called reliable if the issued forecast probabilities and their conditional event frequencies are equal.

To estimate the conditional event frequencies, the forecast probabilities are grouped into a finite number of non-overlapping bins. The reliability diagram is a plot of the conditional event frequency per bin 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 user-defined bins, that do not have to be equidistant. The consistency resampling method proposed by Bröcker and Smith [2007] is used to estimate the expected spread of the reliability diagrams around the diagonal to assess the null-hypothesis that the forecast is reliable. If the `plot` argument is set to `FALSE`, the `ReliabilityDiagram` function returns the quantities necessary to plot the reliability diagram.

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

##   p.avgs cond.probs cbar.lo cbar.hi p.counts bin.lower bin.upper
## 1 0.1713     0.2222  0.0000  0.4784        9    0.0000    0.3333
## 2 0.5833     0.4286  0.1667  1.0000        7    0.3333    0.6667
## 3 0.8447     1.0000  0.6000  1.0000       11    0.6667    1.0000
```

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

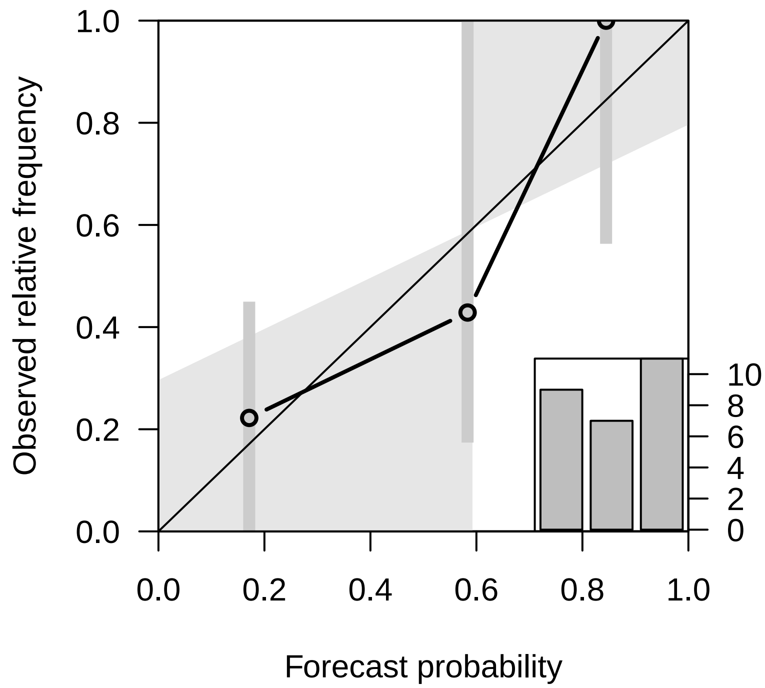

Figure 3: Reliability diagram of temperature exceedance forecasts `p.bin`, using 3 equidistant bins to estimate the conditional event frequencies.

If the argument `plot=TRUE` the reliability diagram is plotted, as shown in Figure 3. 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).

The additive decomposition of the Brier Score into Reliability, Resolution, and Uncertainty is closely related to the Reliability diagram, and allows for a quantitative assessment of different forecast attributes. The empirical mean of the Brier Score of the probability forecasts $p_1, \cdots, p_N$ for the binary observations $y_1, \cdots, y_N$ (cf. Equation 1) can be decomposed additively into three terms, as follows

$$\frac{1}{N}\sum_{t=1}^{N}(y_t - p_t)^2 = REL - RES + UNC. \tag{32}$$

Assume that the forecast probabilities were categorised into D disjoint bins, and a reliability curve was calculated. The in-bin average of the forecast probabilities in the dth bin is denoted by $\bar{p}_d$, the total number of cases in the dth bin is denoted by n_d, the conditional event frequency in the dth bin is denoted by $\bar{y}_d$, and the overall average of the observations is denoted by $\bar{y}$. Then the three components of the Brier Score decomposition are given by [Murphy, 1973]:

$$REL = \frac{1}{N}\sum_{d=1}^{D} n_d(\bar{p}_d - \bar{o}_d)^2, \tag{33}$$

$$RES = \frac{1}{N}\sum_{d=1}^{D} n_d(\bar{o}_d - \bar{y})^2, \tag{34}$$

$$UNC = \bar{y}(1 - \bar{y}). \tag{35}$$

Note that all components can be calculated from the data returned by the `ReliabilityDiagram` function. The Reliability term REL quantifies the weighted squared distance of the reliability curve from the diagonal. The Resolution term RES quantifies the weighted squared distance of the reliability curve from the climatological mean forecast. The Uncertainty term equals the mean Brier Score of the climatological forecast. A bias-corrected Brier score decomposition was proposed by Ferro and Fricker [2012], and approximate variance estimators were derived by Siegert [2013]. The biased and bias-corrected decompositions and their variance estimators are implemented in `SpecsVerification` in the function `BrierDecomp`. For example, the bias-corrected Brier Score of the temperature exceedance forecasts has components and standard deviations equal to

```
BrierDecomp(p.bin, obs.bin, bins=3, bias.corrected=TRUE)

##                    REL    RES     UNC
## component    0.0004808 0.1125 0.25000
## component.sd 0.0160577 0.0428 0.01818
```

The Brier Score decomposition suggests that the forecasts are reliable ($REL \approx 0$), and have non-trivial forecast skill ($RES > 0$).

7 Additional functions

The focus of `SpecsVerification`, and of the present paper, is on verification functions for ensemble and probability forecasts, uncertainty assessment, and graphical display of the results. `SpecsVerification` includes a number of functions for post-processing and forecast verification that were included following user requests.

For verification of deterministic forecasts x_t , e.g. the ensemble mean forecast, the squared error score $(x_t - y_t)^2$ (function `SqErr`), and the Absolute Error score $|x_t - y_t|$ (function `AbsErr`) have been implemented. The function `GaussCrps` calculates the CRPS (eq. 9) where the forecast cdf $F_t(x)$ is a Normal distribution with mean μ_t and standard deviation σ_t^2. The CRPS integral can then be solved analytically [Gneiting et al., 2005] thus eliminating the need for expensive numerical integration. The output from these score functions can further be analysed by the functions `ScoreDiff` and `SkillScore`.

Ensemble forecasts produced by numerical climate models often contain systematic errors due to numerical approximations, missing physical mechanisms in the model, or coding errors. These biases include a constant bias of the mean, or ensemble dispersion errors. Ensemble dressing is a statistical post-processing method based on kernel density estimation to transform a raw forecast ensemble into a smoothed probability distribution function. In particular, affine kernel dressing [AKD; Bröcker and Smith, 2008] is a method that corrects systematic model errors by an affine transformation of the ensemble, and produces a smooth forecast distribution by dressing the transformed ensemble members with Gaussian kernels. AKD is implemented in the function `DressEnsemble`, and the AKD parameters can be fitted with the function `FitAkdParameters`. The forecast quality of the dressed ensemble can be evaluated by the CRPS using the function `DressCrps` which uses the analytical expression of Grimit et al. [2006], and by the Ignorance Score [Roulston and Smith, 2002] (also known as the Logarithmic score), using the function `DressIgn`. The dressed ensemble can be further analysed by the functions `GetDensity` and `PlotDressedEns`.

Auxiliary functions were added to the package, namely the function `Detrend` to remove a common linear trend from an ensemble of forecasts, and the function `GenerateToyData` that simulates artificial ensemble and observation data using a statistical signal-plus-noise model [Siegert et al., 2016].

8 Conclusion

The package `SpecsVerification` for the `R` statistical programming environment implements a variety of forecast verification functions that are not available in previous verification software packages. The focus of `SpecsVerification` is on comparative verification of ensemble forecasts, and uncertainty assessment by variance estimation, statistical testing, and confidence intervals. Continuous, categorical and binary ensemble forecasts for univariate quantities can be evaluated and compared by a number of verification scores. Additional functions for data transformation and statistical post-processing simplify a number of common verification tasks.

Acknowledgements

This paper has benefitted from numerous discussions with the members of the statistical science group at the University of Exeter, in particular Chris Ferro and David Stephenson. This work was supported by the European Union Programme FP7/2007-13 under grant agreement 3038378 (SPECS). The views expressed herein are those of the author and do not necessarily reflect the views of funding bodies and their subagencies.

References

Jonas Bhend, Jacopo Ripoldi, Claudia Mignani, Irina Mahlstein, Rebecca Hiller, Christoph Spirig, Mark Liniger, Andreas Weigel, Joaquín Bedia Jimenez, and Matteo De Felice. *easyVerification: Ensemble Forecast Verification for Large Data Sets*, 2016. URL `https://CRAN.R-project.org/package=easyverification`. R package version 0.2.0.

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1): 1–3, Jan 1950. doi: 10.1175/1520-0493(1950)078¡0001:vofeit¿2.0.co;2.

J. Bröcker. On reliability analysis of multi-categorical forecasts. *Nonlinear Processes in Geophysics*, 15(4):661–673, 2008. ISSN 1607-7946. doi: 10.5194/npg-15-661-2008. URL `http://dx.doi.org/10.5194/npg-15-661-2008`.

Jochen Bröcker and Leonard A. Smith. Increasing the reliability of reliability diagrams. *Weather and Forecasting*, 22(3):651–661, Jun 2007. ISSN 1520-0434. doi: 10.1175/waf993.1. URL `http://dx.doi.org/10.1175/WAF993.1`.

Jochen Bröcker and Leonard A. Smith. From ensemble forecasts to predictive distribution functions. *Tellus A*, 60(4):663–678, Aug 2008. ISSN 1600-0870. doi: 10.1111/j.1600-0870.2008.00333.x. URL `http://dx.doi.org/10.1111/j.1600-0870.2008.00333.x`.

R. Buizza and T. N. Palmer. Impact of ensemble size on ensemble prediction. *Monthly Weather Review*, 126 (9):2503–2518, Sep 1998. ISSN 1520-0493. doi: 10.1175/1520-0493(1998)126¡2503:ioesoe¿2.0.co;2. URL `http://dx.doi.org/10.1175/1520-0493(1998)126<2503:IOESOE>2.0.CO;2`.

Elizabeth R. DeLong, David M. DeLong, and Daniel L. Clarke-Pearson. Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach. *Biometrics*, 44(3): 837–845, Sep 1988. ISSN 0006-341X. doi: 10.2307/2531595. URL `http://dx.doi.org/10.2307/2531595`.

Francis X. Diebold and Roberto S. Mariano. Comparing predictive accuracy. *Journal of Business & Economic Statistics*, 13(3):253, Jul 1995. doi: 10.2307/1392185.

C. A. T. Ferro. Fair scores for ensemble forecasts. *Quarterly Journal of the Royal Meteorological Society*, 140(683):1917–1923, Dec 2013. doi: 10.1002/qj.2270.

C. A. T. Ferro and T. E. Fricker. A bias-corrected decomposition of the brier score. *Quarterly Journal of the Royal Meteorological Society*, 138(668):1954–1960, Mar 2012. ISSN 0035-9009. doi: 10.1002/qj.1924. URL `http://dx.doi.org/10.1002/qj.1924`.

Christopher A. T. Ferro, David S. Richardson, and Andreas P. Weigel. On the effect of ensemble size on the discrete and continuous ranked probability scores. *Meteorological Applications*, 15(1):19–24, 2008. doi: 10.1002/met.45.

Thomas E. Fricker, Christopher A. T. Ferro, and David B. Stephenson. Three recommendations for evaluating climate predictions. *Meteorological Applications*, 20(2):246–255, Jun 2013. doi: 10.1002/met.1409.

T. Gneiting. Atmospheric science: Weather forecasting with ensemble methods. *Science*, 310(5746):248–249, Oct 2005. ISSN 1095-9203. doi: 10.1126/science.1115255. URL `http://dx.doi.org/10.1126/science.1115255`.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, Mar 2007. doi: 10.1198/016214506000001437.

Tilmann Gneiting, Adrian E. Raftery, Anton H. Westveld, and Tom Goldman. Calibrated probabilistic forecasting using ensemble model output statistics and minimum crps estimation. *Monthly Weather Review*, 133(5):1098–1118, May 2005. ISSN 1520-0493. doi: 10.1175/mwr2904.1. URL `http://dx.doi.org/10.1175/MWR2904.1`.

E. P. Grimit, T. Gneiting, V. J. Berrocal, and N. A. Johnson. The continuous ranked probability score for circular variables and its application to mesoscale forecast ensemble verification. *Q.J.R. Meteorol. Soc.*, 132(621C):2925–2942, Oct 2006. ISSN 1477-870X. doi: 10.1256/qj.05.235. URL `http://dx.doi.org/10.1256/qj.05.235`.

Virginie Guemas, Nicolau Manubens, Louis-Philippe Caron, Verónica Torralba, Chloé Prodhomme, Martin Ménégoz, Javier Garcia-Serrano, Fabian Lienert, Ludovic Auger, and Isabel Andreu-Burillo. *s2dverification: Set of Common Tools for Forecast Verification*, 2016. URL `https://CRAN.R-project.org/package=s2dverification`. R package version 2.5.0.

Thomas M. Hamill. Interpretation of rank histograms for verifying ensemble forecasts. *Monthly Weather Review*, 129(3):550–560, Mar 2001. 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`.

Wu-ron Hsu and Allan H. Murphy. The attributes diagram a geometrical framework for assessing the quality of probability forecasts. *International Journal of Forecasting*, 2(3):285–293, Jan 1986. ISSN 0169-2070. doi: 10.1016/0169-2070(86)90048-8. URL `http://dx.doi.org/10.1016/0169-2070(86)90048-8`.

Ian T. Jolliffe. Uncertainty and inference for verification measures. *Weather and Forecasting*, 22(3):637–650, Jun 2007. doi: 10.1175/waf989.1. URL `http://dx.doi.org/10.1175/WAF989.1`.

Ian T. Jolliffe and Cristina Primo. Evaluating rank histograms using decompositions of the chi-square test statistic. *Monthly Weather Review*, 136(6):2133–2139, Jun 2008. ISSN 1520-0493. doi: 10.1175/2007mwr2219.1. URL `http://dx.doi.org/10.1175/2007MWR2219.1`.

Ian T Jolliffe and David B Stephenson. *Forecast verification: a practitioner's guide in atmospheric science*. John Wiley & Sons, 2012.

M. Leutbecher and T.N. Palmer. Ensemble forecasting. *Journal of Computational Physics*, 227(7):3515–3539, Mar 2008. ISSN 0021-9991. doi: 10.1016/j.jcp.2007.02.014. URL `http://dx.doi.org/10.1016/j.jcp.2007.02.014`.

S. J. Mason and N. E. Graham. Areas beneath the relative operating characteristics (roc) and relative operating levels (rol) curves: Statistical significance and interpretation. *Q. J. R. Meteorol. Soc.*, 128(584):2145–2166, Jul 2002. ISSN 0035-9009. doi: 10.1256/003590002320603584. URL `http://dx.doi.org/10.1256/003590002320603584`.

James E. Matheson and Robert L. Winkler. Scoring rules for continuous probability distributions. *Management Science*, 22(10):1087–1096, Jun 1976. doi: 10.1287/mnsc.22.10.1087.

Allan H. Murphy. A new vector partition of the probability score. *J. Appl. Meteor.*, 12(4):595–600, Jun 1973. ISSN 0021-8952. doi: 10.1175/1520-0450(1973)012¡0595:anvpot¿2.0.co;2. URL `http://dx.doi.org/10.1175/1520-0450(1973)012<0595:ANVPOT>2.0.CO;2`.

NCAR - Research Applications Laboratory. *verification: Forecast verification utilities.*, 2015. URL `https://CRAN.R-project.org/package=verification`. R package version 1.42.

Karl Pearson. 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, Jul 1900. ISSN 1941-5990. doi: 10.1080/14786440009463897. URL `http://dx.doi.org/10.1080/14786440009463897`.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2015. URL `https://www.R-project.org/`.

Mark S. Roulston and Leonard A. Smith. Evaluating probabilistic forecasts using information theory. *Monthly Weather Review*, 130(6):1653–1660, Jun 2002. ISSN 1520-0493. doi: 10.1175/1520-0493(2002)130¡1653:epfuit¿2.0.co;2. URL `http://dx.doi.org/10.1175/1520-0493(2002)130<1653:EPFUIT>2.0.CO;2`.

Suranjana Saha, Shrinivas Moorthi, Hua-Lu Pan, Xingren Wu, Jiande Wang, Sudhir Nadiga, Patrick Tripp, Robert Kistler, John Woollen, David Behringer, and Coauthors. The ncep climate forecast system reanalysis. *Bulletin of the American Meteorological Society*, 91(8):1015–1057, Aug 2010. ISSN 1520-0477. doi: 10.1175/2010bams3001.1. URL `http://dx.doi.org/10.1175/2010BAMS3001.1`.

Suranjana Saha, Shrinivas Moorthi, Xingren Wu, Jiande Wang, Sudhir Nadiga, Patrick Tripp, David Behringer, Yu-Tai Hou, Hui-ya Chuang, Mark Iredell, and Coauthors. The ncep climate forecast system version 2. *J. Climate*, 27(6):2185–2208, Mar 2014. ISSN 1520-0442. doi: 10.1175/jcli-d-12-00823.1. URL `http://dx.doi.org/10.1175/JCLI-D-12-00823.1`.

Santander Meteorology Group. *ecomsUDG.Raccess: R interface to the ECOMS User Data Gateway*, 2015. URL `http://meteo.unican.es/trac/wiki/udg/ecoms`. R package version 4.2-0.

Stefan Siegert. Variance estimation for brier score decomposition. *Quarterly Journal of the Royal Meteorological Society*, 140(682):1771–1777, Aug 2013. ISSN 0035-9009. doi: 10.1002/qj.2228. URL `http://dx.doi.org/10.1002/qj.2228`.

Stefan Siegert, David B. Stephenson, Philip G. Sansom, Adam A. Scaife, Rosie Eade, and Alberto Arribas. A bayesian framework for verification and recalibration of ensemble forecasts: How uncertain is nao predictability? *J. Climate*, 29(3):995–1012, Feb 2016. ISSN 1520-0442. doi: 10.1175/jcli-d-15-0196.1. URL `http://dx.doi.org/10.1175/JCLI-D-15-0196.1`.

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

O Talagrand, R Vautard, and B Strauss. Evaluation of probabilistic prediction systems. In *Proc. ECMWF Workshop on Predictability*, volume 1, page 25, 1997.

Hans Von Storch and Francis W Zwiers. *Statistical analysis in climate research*. Cambridge university press, 2001.

Daniel S Wilks. *Statistical methods in the atmospheric sciences*, volume 100. Academic press, 2011.

Guang Yong Zou. Toward using confidence intervals to compare correlations. *Psychological Methods*, 12(4):399–413, 2007. ISSN 1082-989X. doi: 10.1037/1082-989x.12.4.399. URL `http://dx.doi.org/10.1037/1082-989X.12.4.399`.