Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. *doi: 10.18637/jss.v000.i00*

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.

```
$ data(eurotempforecast)
$ ens <- ens - mean(ens) + mean(obs)
$ yrs <- as.numeric(names(obs))
$ N <- length(obs)
```

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

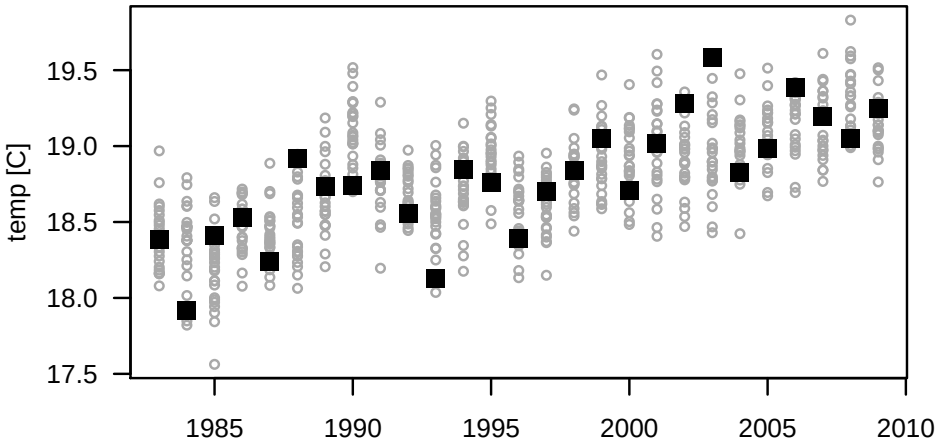

Figure 1: Seasonal European temperature forecasts by NCEP CFSv2, initialised in May, verified in JJA.

2. Ensemble-adjusted verification scores

2.1. Representation of ensemble and observation data

An archive of N time instances of ensemble forecasts, each with R members, can be conveniently represented by a $N \times R$ matrix:

This data can be interpreted that, for example, on the first time instance 1 out of 4 ensemble members forecast rain, and rain actually occurs.

MORE HERE

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:

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

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).

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

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

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

$$E\left[\left(\frac{i_t}{R} - y_t\right)^2\right] = (p_t - y_t)^2 + \frac{p_t(1-p_t)}{R} \tag{3}$$

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 $R \to \infty$. 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) Brier score on average. This is reasonable since more ensemble members allow for more robust estimation of the "true" probability p_t. But there are cases, where it is desirable to estimate and correct the finite-ensemble bias.

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

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

contains a correction for the finite-ensemble bias. The ensemble-adjusted Brier score is 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.,

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

Note that, trivially, $s^*(i_t, R, R, y_t) = s(i_t/R, y_t)$. 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"?

```
$ obs.bin <- 1 * (obs[2:N] > obs[1:(N-1)])
$ ens.bin <- 1 * (ens[2:N, ] > obs[1:(N-1)])
$ print(c(mean(EnsBrier(ens.bin, obs.bin)),
+         mean(EnsBrier(ens.bin, obs.bin, R.new=Inf))))

[1] 0.1405582 0.1337793
```

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 $\mathbf{p}_t = (p_{t,1}, \cdots, p_{t,K})$, 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

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

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 $k = 1, \cdots, K$. Using results obtained for the ensemble-adjusted Brier score, the ensemble-adjusted quadratic score is seen to be

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

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

The quadratic score is insensitive to relabelling the K categories. This is undesired in categorical forecasting problems where the categories are nested. An order sensitive score for categorical forecasts is the ranked probability score (RPS). 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 thus transformed from $i_{t,k}$ ensemble members predict category k to the forecast $j_{t,k}$ ensemble members forecast category k *or less*. The nesting of forecast categories ensures order-sensitivity of the score. Using previous results, we get the ensemble-adjusted RPS

$$s^*(\mathbf{i}_t, R, R^*, \mathbf{y}_t) = \sum_{k=1}^{K} \left\{ \left(\frac{\sum_{l=1}^{k} i_{t,k}}{R} - \sum_{l=1}^{k} y_{t,k} \right)^2 - \left(\frac{1}{R} - \frac{1}{R^*} \right) \frac{\sum_{l=1}^{k} i_{t,k}(R - \sum_{l=1}^{k} i_{t,k})}{R(R-1)} \right\} \tag{8}$$

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 within one half of a degree of last year's temperature, colder, or warmer?"

```
$ categ <- function(x, cat.ctr) {
+   as.numeric(cut(x, breaks=c(-Inf, cat.ctr-.25, cat.ctr+.25, Inf)))
+ }
$ obs.cat <- sapply(2:N, function(i) categ(obs[i], obs[i-1]))
$ ens.cat <- sapply(1:ncol(ens), function(j) {
+                   sapply(2:N, function(i) categ(ens[i, j], obs[i-1]))
+             })
$ print(c(mean(EnsQs(ens.cat, obs.cat)), mean(EnsQs(ens.cat, obs.cat, R.new=Inf))))

[1] 0.5956197 0.5777592

$ print(c(mean(EnsRps(ens.cat, obs.cat)), mean(EnsRps(ens.cat, obs.cat, R.new=Inf))))

[1] 0.3448851 0.3417778
```

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

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

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

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

The CRPS of this empirical distribution function is given by

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

Fricker, Ferro, and Stephenson (2013) show that the ensemble-adjusted CRPS is given by

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

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. The ensemble-adjusted CRPS is implemented in the **SpecsVerification** function `EnsCrps`.

```
$ print(c(mean(EnsCrps(ens, obs)), mean(EnsCrps(ens, obs, R.new=Inf))))

[1] 0.1380708 0.1328890
```

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

2.5. Deterministic forecasts

For completeness, functions for verification of deterministic (point) forecasts have been included in **SpecsVerification**, however, without any adjustement for ensemble size:

- `Sqerr(fcst, obs)`
- `Mae(fcst, obs)`

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:

```
$ ens.ref <- ClimEns(obs, leave.one.out=TRUE)
$ ens.cat.ref <- ClimEns(obs.cat, leave.one.out=TRUE)
$ 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 $\{S_{1,1}, S_{1,2}, \ldots, S_{1,N}\}$ and $\{S_{2,1}, S_{2,2}, \ldots, S_{2,N}\}$ 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 $d_1, \ldots, d_N$ 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

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

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`.

```
$ 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 0.12344177    0.04243153 1.811778e-03  0.04027751 0.2066060
qs    0.11158034    0.09380369 1.171197e-01 -0.07227152 0.2954322
rps   0.09831485    0.06649896 6.964476e-02 -0.03202071 0.2286504
crps  0.09391427    0.02399588 4.543379e-05  0.04688321 0.1409453
```

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:

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

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

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

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.

```
$ 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      stdev
brier  0.4675825 0.15188106
qs     0.1577776 0.13439394
rps    0.2218295 0.14786715
crps   0.4048290 0.07343353
```

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:

```
$ ens.mean <- rowMeans(ens)
$ Corr(ens.mean, obs)

        corr      p.value            L            U
7.570956e-01 2.426814e-06 5.293911e-01 8.830500e-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 (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:

```
$ persist <- c(NA, obs[1:(N-1)])
$ CorrDiff(ens.mean, persist, obs, handle.na="only.complete.triplets")

  corr.diff     p.value           L           U
 0.18861180  0.03235523 -0.01072800  0.46511450
```

4. Rank histogram analysis for ensemble forecasts

Talagrand; Anderson; Broecker; Jolliffe and Primo

5. Reliability diagrams for probability forecasts

6. 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.

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.

Matheson JE, Winkler RL (1976). "Scoring Rules for Continuous Probability Distributions." *Management Science*, **22**(10), 1087–1096. doi:10.1287/mnsc.22.10.1087.

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

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
MMMMMM YYYY, Volume VV, Issue II *Submitted:* yyyy-mm-dd
doi:10.18637/jss.v000.i00 *Accepted:* yyyy-mm-dd