VERIFICATION OF NATIONAL WEATHER SERVICE PROBABILISTIC HYDROLOGIC FORECASTS

by

Kristie J. Franz
and
Soroosh Sorooshian

Department of Hydrology and Water Resources
The University of Arizona

Tucson, Arizona

November 2002

Submitted Under Grant Number 40-AA-NW-217447

FINAL REPORT

VERIFICATION OF NATIONAL WEATHER SERVICE PROBABILISTIC HYDROLOGIC FORECASTS

Kristie J. Franz and Soroosh Sorooshian
Department of Hydrology and Water Resources
The University of Arizona

I. Introduction and Scope

Hydrologic outlooks issued by the National Weather Service (NWS) have traditionally been based on linear regression procedures that produce a single-valued forecast, with no indication of uncertainty in the predicted streamflow event. The Ensemble Streamflow Prediction (ESP) method was developed to improve the quantity and quality of information in the NWS outlooks. The ESP produces multiple estimates of a streamflow variable based on current basin conditions and past meteorological observations. Probability and uncertainty are derived from the distribution of the predicted values. ESP is considered to be an advanced hydrological forecasting product because it provides probabilistic forecast information rather than deterministic (Day, 1985).

The ESP forecast system has been distributed to NWS River Forecast Centers (RFCs) across the country as part of the NWS River Forecast System (NWSRFS). The procedure has been operationally implemented at several RFCs within the past few years and will become more widely used in the coming years. As ESP forecasts accumulate, it will be important that they are evaluated because verification provides hydrologists information about forecast qualities and needed improvements. In addition, verification gives the users (e.g. water supply managers, municipalities, flood managers) information about how to most effectively use the forecast (Murphy and Winkler, 1987). Without verification, there is a lack of quantitative information to support forecast credibility (Hartmann, 1999). It is necessary that the NWS develop a thorough and informative evaluation method(s), which will allow consistent tracking of ESP performance in both space and time and be useful to both forecasters and users.

Franz (2001) described and applied probabilistic forecast evaluation methods to simulated NWS ESP water supply outlooks for the Colorado River basin. Three types of probabilistic verification methods were successfully applied to the forecasts: RPS (Epstein, 1969; Wilks, 1995), ranked probability skill score (RPSS) (Wilks, 1995), and discrimination and reliability (Murphy et al., 1989, 1987; Wilks, 1995). Franz (2001) illustrated that these methods are capable of describing the probabilistic skill of the ESP forecasts and a variety of information contained within them. The purpose of the current study was to investigate the feasibility of using the RPS, RPSS, discrimination, and reliability as applied by Franz (2001) for verification of operational (or RFC generated) NWS ESP forecasts.

This project has been conducted in cooperation with the NWS and is part of a comprehensive effort to examine forecast evaluation as part of the Advanced Hydrologic Prediction System (AHPS). The Colorado River Basin RFC has studied water supply forecasts produced using ESP and regression equations to compare the forecast

information obtained from both methods. The NWS OH is currently developing evaluation tools as part of a short-term ensemble demonstration project at the Mid-Atlantic RFC. Ensemble forecasts of temperature, precipitation, and streamflow from this region are being evaluated by comparison of the ensemble mean and observation, the calculation of the ranked probability score (RPS), and comparison of the observation and forecast distributions to test forecast probability. In addition, the ESP Verification system (ESPVS) was developed to aid in the evaluation of the ESP forecasting method through the use of synthetic historical ESP forecasts. The North Central RFC has used this application to compare the ESP method to previous forecast methods for the generation of spring flood outlooks. The ESPVS was also used by Franz (2001) to generate the simulated forecasts required to conduct the ESP water supply forecast evaluation.

This final report is a summary of a forecast evaluation study completed under Grant Number 40-AA-NW-217447 and is compiled as follows. Probabilistic verification procedures are discussed in Section II. Forecasts and observation data used for completion of the study are discussed in Section III. Evaluation procedures specific to this study are discussed in Section IV. Results are presented and discussed in Section V. A discussion of the results, feasibility of implementation, and issues that arose are discussed in Section VI. Conclusions are given in Section VII. Recommendations based on this study are summarizes in Section VIII, and acknowledgements and references are provided in Sections IX and X, respectively.

II. Probabilistic Forecast Verification Procedures

Conventional summary statistical methods, such as root mean square error and bias, are unable to evaluate the probabilistic nature of ESP forecasts because they only evaluate whether a forecast is right or wrong, 1 or 0. A probabilistic forecast, such as ESP, has a set of values between 1 and 0, and they are therefore neither right nor wrong (Wilks, 1995). The observation will have a value of either 0 (did not occur) or 1 (did occur) (Murphy and Winkler, 1992). In addition, a meaningful assessment of probabilistic forecasts cannot be made on a single prediction; behavior trends can only be assessed through the evaluation of a collection of forecasts and observation pairs (Wilks, 1995).

The diagnostic verification methods ranked probability score (RPS), ranked probability skill score (RPSS), discrimination, and reliability are useful in detailed investigations of probabilistic forecasting skill and are the focus of this study. The application of diagnostic verification has occurred predominantly in the meteorological field. Diagnostic verification has been developed and illustrated previously by Murphy and Winkler (1992, 1987) and by Murphy et al. (1989) using probability of precipitation and maximum temperature forecasts. Wilks (1995) described the methods in detail and discussed their application to meteorological forecasts. Wilks (2000) used diagnostic verification in a study of Climate Prediction Center average temperature and total precipitation long-range forecasts over the United States. The methods have not been applied extensively to hydrologic forecasts.

Forecast verification involves analysis of the correspondence between the forecast and observation of a predicted event. The strength of the correlation between the forecast

and observations can be investigated through analysis of their joint distribution denoted by:

$$(p(f_i,o_j)) \tag{1}$$

where f_i = forecasts that have any of I values $f_1, f_2, \ldots f_I$; and o_j = observations that have any of J values $o_1, o_2, \ldots o_J$. A probability ([0 1]) is associated with each of the possible combinations of forecasts and observation (Wilks, 1995).

The information contained in the joint distribution is more accessible through analysis of the marginal distributions ($p(f_i)$ and $p(o_j)$) and conditional distributions ($p(o_j|f_i)$ and $p(f_i|o_j)$) (Murphy and Winkler, 1987). The term $(p(o_j|f_i)p(f_i)$ is referred to as the calibration refinement factorization (Wilks, 1995) and is used to evaluate forecast reliability. The term $p(o_j|f_i)$ indicates how often a various observation occurred given a specific forecast, while $p(f_i)$ indicates how often a particular forecast was made. The term $p(f_i|o_j)\ p(o_j)$ is called the likelihood-base rate by Wilks (1995) and is used to examine forecast discrimination. The first term describes the frequency with which specific forecast probability values were given prior to a specific observation. The second term indicates the relative frequency of the various observations and is equal to the climatology of the observations.

Ranked Probability Score (RPS)

The ranked probability score (RPS) is used to assess the overall forecast performance of the probabilistic forecasts (Epstein, 1969; Wilks, 1995). To calculate the RPS, specific forecast probability categories are first created (Table 1, column 1). The forecast non-exceedance probability categories chosen were based on the values archived in the OHRFC forecast files (5, 10, 25, 50, 75, 90, and 95% exceedance). Several variations of this set were investigated and are identified with the corresponding results. Each forecast probability category is bounded by its associated streamflow non-exceedance values determined from the cumulative distribution function (CDF) of the traces or climatology. Once the ensemble members are distributed into these categories, the relative frequency (Table 1, column 2) and forecast cumulative distribution (F_m) (Table 1, column 4) are obtained:

$$F_m = \sum_{j=1}^{m} f_j, m = 1,\ldots J, \tag{2}$$

where J equals the number of forecast categories (Wilks, 1995).

The observation occurs in only one of the flow categories, which is given a value of 1; the remaining categories are given a value of 0 (Table 1, column 3). The cumulative distribution of the observations (O_m) is then calculated (Table 1, column 5) (Wilks, 1995).

$$O_m = \sum_{j=1}^{m} o_i, m = 1,\ldots J \tag{3}$$

The RPS for one forecast is the sum of the squared errors of the cumulative distributions:

$$RPS = \sum_{m=1}^{J} (F_m - O_m)^2 \tag{4}$$

(Table 1: column 6). For a group of *n* forecasts, the RPS is the average ($\overline{RPS}$) of the *n* RPSs:

$$\overline{RPS} = \frac{1}{n} \sum_{k=1}^{n} RPS_k . \tag{5}$$

A perfect forecast would assign all of the probability to the same streamflow category in which the event occurs, resulting in an RPS value of 0. The RPS is said to be "sensitive to distance" because it increasingly penalizes forecasts that assign probability to streamflow categories further from the observation (Wilks, 1995).

Table 1: Example ranked probability score calculation

Column 1	*Column 2*	*Column 3*	*Column 4*	*Column 5*	*Column 6*
Non-Exceedance Probability Category	Forecast Probability *(f)*	Observation *(o)*	Forecast Cumulative Sum (F)	Observation Cumulative Sum (O)	$(F-O)^2$
0-10%	0.1	0	0.1	0	0.01
>10-30%	0.2	0	0.3	0	0.09
>30-70%	0.4	1	0.7	1	0.09
>70-90%	0.2	0	0.9	1	0.01
>90-100%	0.1	0	1	1	0
				RPS = Sum =	0.2

Ranked Probability Skill Score

A single value, such as the RPS, often cannot put into context the actual quality of a forecast or set of forecasts. It is useful then to compare the forecast of interest to a reference forecast, such as persistence forecasts, historical operational forecasts, or forecasts based on the historical distribution of observed values (climatology) (Wilks, 1995). Due to a lack of other available probabilistic streamflow forecasts, climatology forecasts were generated from the historical observations to serve as reference forecasts. The relative skill of the ESP forecasts was evaluated against the climatology forecasts through the use of the Ranked Probability Skill Score (RPSS):

$$RPSS = \frac{\overline{RPS}_f - \overline{RPS}_{cl}}{\overline{RPS}_{perfect} - \overline{RPS}_{cl}} x100\%, \tag{6}$$

where $\overline{RPS_f}$ is the average RPS of the forecasts for a particular forecast period, $\overline{RPS}_{cl}$ is the RPS of the climatology (reference) forecasts, and $\overline{RPS}_{perfect}$ is equal to a perfect RPS score (Wilks, 1995).

A positive RPSS indicates improvement over climatology and that the forecasts provided additional accurate predictive information. A perfect score is 100%. A negative RPSS indicates that the ESP forecasts performed worse than the climatology forecasts.

Advantage of RPS and RPSS

The RPS evaluates the entire distribution of the forecast. It increasingly penalizes the forecast for assigning high probability to categories farther from where the observation occurred. In this way, the RPS is able to account for the distance and the magnitude of the forecast probability with respect to the observation. The RPSS can be used to compare the RPS of ESP to the RPS of reference forecasts (e.g., climatology, other forecast methods) in order to evaluate the percent improvement that the ESP forecasts display over alternative forecasts. The RPSS can also be used to evaluate improvements made during the forecasting process.

Calculation Steps for RPS and RPSS

For one forecast:

1. Choose forecast percentiles to represent forecast probability and flow categories.
2. Determine streamflow threshold values for each portion of the distribution equal to the forecast probability (or streamflow) desired, based on the historical distribution or distribution of the traces.
3. Determine forecast trace relative frequency per forecast probability category. When using traces rather than flow climatology, the relative frequency will be equal to the probability for each non-exceedance category. (Note: RPS will be the same whether exceedance or non-exceedance probabilities are used.)
4. Place a one in the category in which the observation occurred and a zero in all others.
5. Determine the cumulative distribution of the forecast and observation probability.
6. Take the squared difference of each forecast cumulative probability category and corresponding observation cumulative probability category and sum the results. Because the last interval will always contain ones for both the forecasts and observation, this will be equal to zero.

For several forecasts:

7. The average RPS at one forecast point or group of points should be calculated for forecasts grouped according to lead time, forecast issue date, or some similar characteristic.
8. The skill score can be calculated for average RPS or individual RPS values. The reference forecasts must follow the same format as the ESP forecasts.

Example MATLAB scripts used to calculate RPS and RPSS are provided in Appendix I.

Discrimination

The likelihood that a particular forecast would have been issued prior to a specific observation is expressed in the conditional distribution of the forecasts given the observed (*p(f|o)*) (Wilks, 1995). If the value of *p(f|o)* for a particular observation category is similar to that for a different observation, the forecasts are not discriminatory for that observation. On the other hand, when *p(f|o)* equals zero for all possible observations except one, the forecast procedure is perfectly discriminatory for forecasts for that observation (Murphy and Winkler, 1987).

The discrimination diagram displays the conditional distribution (*p(f|o*) as a function of forecast probability (Figure 1). If the forecasts are discriminatory, then the *p(f|o)* for various *o* will not overlap to a great degree on the discrimination diagram (Figure 1a). If there is little discrimination, there will be considerable overlapping (Figure 1b) (Murphy et al., 1989). A discrimination diagram is produced for occurrences of observations in each flow category; therefore, forecasts that were issued prior to observations of low flows are plotted on a separate discrimination diagram than forecasts that were issued prior to other flow categories. The number of observations represented on each plot is dependent upon the number of historical observations in the respective flow category.

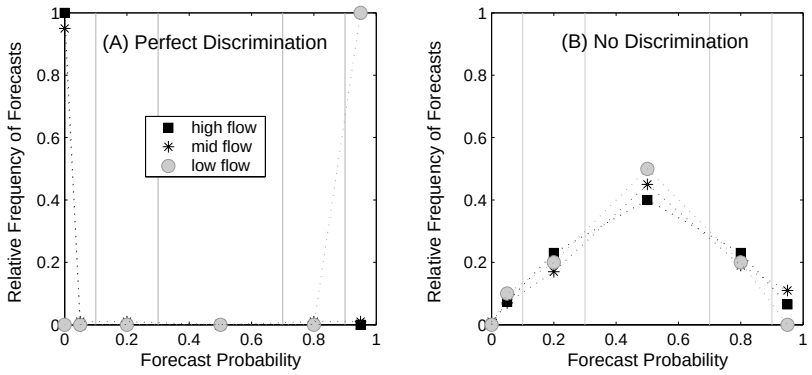

Figure 1: Example discrimination diagrams for forecasts with (A) perfect discrimination for low flows, and (B) no discrimination for forecasts issued prior to low flow observations.

Calculation Steps for Discrimination

1. Choose flow intervals and probability categories of interest (may be different).
2. Determine forecast probability from the relative frequency of traces within each forecast category. The categories are based on the historical streamflow distribution.
3. Divide the forecasts into groups based on the observation type that followed the forecast (i.e., low flow/high flow, flood/no flood).
4. For each observation group, determine the relative frequency of the different forecast probability values that were given to each of the observation types. Each

observation group should have information about each forecast probability level and the amount of probability given to each flow type.

5. Plot the forecast probabilities given to each flow category according to the observations (one graph for each observation category).
6. Count the number of observations in each category; this equals the marginal distribution of observations.

Reliability

Reliability summarizes the information contained in the conditional distribution ($p(o|f)$) and describes how often an observation occurred given a particular forecast. Ideally:

$$p(o = 1 \mid f) = f \tag{7}$$

(Murphy and Winkler, 1987). That is, for a set of forecasts where a forecast probability value f was given to a particular observation, o, the forecasts are considered perfectly reliable if the relative frequency of the observation equals the forecast probability. (Murphy and Winkler, 1992).

The reliability diagram is used to display forecast reliability (Figure 2). The conditional distribution *($(p(o|f))$* of a set of perfectly reliable forecasts will fall along the diagonal line on the diagram. Forecasts that fall within the shaded region of this figure are underforecasting or are not assigning enough probability to the subsequent observation. Those that fall opposite the shaded regions are overforecasting (Murphy et al., 1989). Forecasts that fall on the no-resolution line are unable to resolve occasions when the event is more or less likely than the overall climatology (Wilks, 1995). Such forecasts plot along the horizontal line associated with their climatology (e.g., forecasts with no-resolution for flow in the highest 25% (high flows) would plot along the no-resolution line at 25% relative frequency of observations).

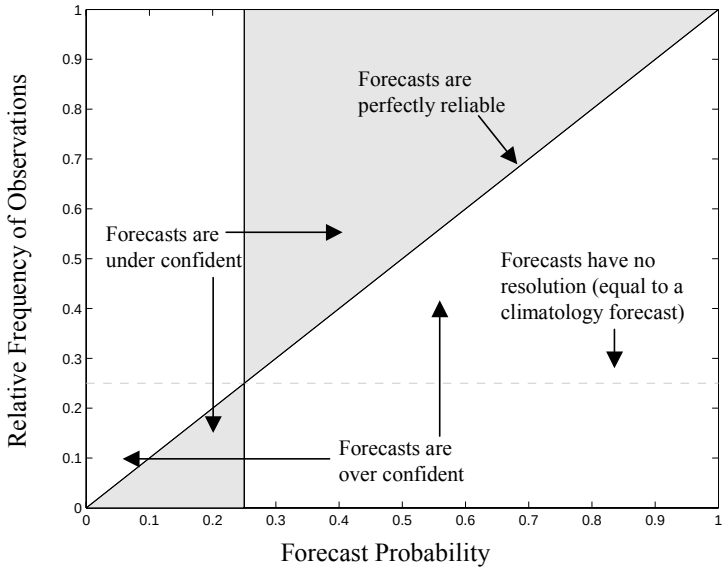

Figure 2: Diagram describing interpretation of the reliability diagrams. Example is for forecasts that are predicting flows in lowest or highest 25% flow category.

As forecasts become sharper or more refined, the forecast probability becomes less distributed, and the forecast probability is more frequently assigned to the extreme non-exceedance categories (e.g., 0% and 90-100%) (Murphy and Winkler, 1987). Thus, the sample sizes within the middle categories become smaller with sharper forecasts. Variations in small sample sizes decrease the ability to assess the quality of the forecasts; hence forecast frequency is often displayed with reliability diagrams for better contextual analysis (Wilks, 1995).

Calculation Steps for Reliability

Repeat steps 1 and 2 from discrimination.

3. For each observation type, determine the number of times that observation occurred after a given forecast probability value was issued for that flow. Do this for each probability category. Calculate the relative frequency of the observation type at each probability category.
4. Determine the number of forecasts that were issued with probability values equal to the probability categories (marginal distribution of the forecasts).
5. Plot the relative frequency of the observations for each observation type. The data point size may reflect the marginal distribution of the forecasts in each category, or a separate histogram can be created.

Advantages of Discrimination and Reliability

Discrimination and reliability are useful in assessing forecast performance at various levels of the historical distribution. Different parts of the forecast probability distribution and the streamflow distribution can be analyzed in a way that reflects the concerns of the forecasters and the forecast users (e.g., drought conditions, flood levels). Discrimination and reliability can give insight into forecast performance and where improvements are needed. Discrimination gives information about the ability of the forecasts to distinguish between the probabilities of various observations. It reveals whether or not the forecasts are able to decipher a higher likelihood for one type of observation over another. It can show whether forecasts are better at predicting some flow types over others and which predictions are more likely to be consistent over a number of forecasts. Reliability gives information about the calibration between forecast probability and the frequency of the predicted observations. It reveals bias in the forecasts and thereby gives insight into needed adjustments.

An example MATLAB script used to calculate discrimination and reliability is provided in Appendix II.

III. Study Data and Data Processing

Forecast data, forecast observations, and historical observations were requested from NWS personnel at the Office of Hydrology (OH) in Silver Spring, Maryland, for as many forecast points as were available. OH collected streamflow forecasts and observation data from the issuing RFCs. The data were then screened and reformatted (if necessary) by OH personnel before being sent to the University of Arizona (UA) Department of Hydrology and Water Resources.

Data for 42 Ohio River Forecast Center (OHRFC) forecast points was received by UA. The forecast files had the following characteristics:

- ASCII format
- Contained ESP trace values and forecast exceedance probabilities based on the empirical distribution of the traces.
- Forecast Type 1 was predictions of mean weekly stage (7-day interval) with a 6-day lead time.
- Forecast Type 2 was predictions of maximum monthly stage (30 day interval) with a 6-day lead time.
- Average number of Type 1 and Type 2 forecasts per forecast point was 11.
- Forecast periods covered 12/12/2001 to 3/24/2002.
- Forecast values included any effects of post-processing and represent the actual forecast information that was issued by OHRFC.
- The forecasts started and ended at 11 Z time (6:00 a.m. EST) on the first and last day of the forecast interval.

Forecast observations are defined as the value of the forecasted variable observed at the forecast point for the exact period of the forecast. The forecast observations files had the following characteristics:

- Observations of river stage in feet with an average time interval of one hour.
- Spanned the forecast period.
- Available for each point.

In some forecast observation data files, the hour interval was shortened during certain periods to ½ hour and lengthened during other periods to intervals greater than 1 hour. In the latter case, this resulted in a situation where data were missing within the forecast interval. Missing data at the beginning and the end of the forecast interval made it necessary to generate a set of rules to deal with the missing forecast observations.

- If an 11 Z data point was missing at the beginning of the forecast interval, the observed data were taken to start at 0 Z of the first day and end at 11 Z of the last day.
- If an 11 Z data point was missing at the end of the forecast interval, the interval ended at the last observation of the last day. (Data missing at the beginning and the end of the interval resulted in an 8-day observation period that includes the entire first and last day.)
- If one day was missing at either the beginning or the end of the forecast interval, the observed period started or ended at one day past the missing day.
- Where data for more than one day were missing, the forecast was not evaluated.

In addition, the forecast observed data provided by the NWS ended 3 days earlier than the last 7-day forecast provided. Therefore, the last weekly mean stage forecast was not evaluated. The forecast observations ended and average 24 days before the monthly maximum stage forecast intervals ended; therefore, the last 4-5 monthly forecasts could not be evaluated.

OH provided historical observations for some of the forecast points as available. Historical observations are defined as forecast variable observations realized during the

same forecast interval for years other than the forecast year. The historical data are used to develop the climatology of the forecasted hydrologic variable. The historical observations files had the following characteristics:

- Values of mean daily stage.
- The historical observation data were severely limited.
- Observed data were available for 19 of the 43 forecast points.
- Average number of historical years was 3.
- The most recent data were from 1984 (with the exception of WROT1, which ended at 1992).

Because the historical observations were daily rather than hourly measures, it was necessary to decide whether to include the first and last forecast day, resulting in an 8-day period, or to omit one or the other, resulting in a 7-day period. Because the data was already limited and streamflow data display persistence, an 8-day (30-day) period was used and a mean (maximum) stage was computed using the data from all days within the forecast interval. The missing historical observations were dealt with in a similar manner to forecast observations:

- If the forecast start or end day data were missing, either the day after or the day prior was used, depending upon availability.
- If data for more than one day were missing at either the beginning or the end of the forecast interval, data for that historical year were considered missing.

In some instances, data from as many as 10 days would be missing from the historical record during a 30-day forecast interval. This severely limited the quantity of past observations available to generate a distribution for monthly forecasts.

IV. Procedures

Two ways to derive forecast probability from the traces was examined in this project. In the first case, the exceedance or non-exceedance probability associated with each trace is calculated from the empirical distribution of the traces. This is the simplest method because it does not require data other than the forecast traces. The forecast is usually developed for a set of the probability values (e.g., 5, 10, 25, 50, 75, 90, and 95% exceedance probability) rather than as a continuous function. This method results in forecasts that always provide the same exceedance probability ranges, while the forecasted streamflow values change.

In the second case, the probability is based on the relative frequency of traces that fall within certain streamflow categories. The streamflow categories are chosen based on streamflow levels of interest, such as particular percentiles. Boundary values for the categories are obtained from the empirical distribution of historical observations (or known streamflow climatology) at the forecast point for the same forecast interval. The number of forecast traces that fall within each category according to their value is recorded. The relative frequency of the forecast traces per bin provides the forecast probability for the given flow category. This method results in forecasts that always predict for the same streamflow values, while the probability applied to the streamflow values will change.

In case two, historical observations provide a point of reference by which to qualify the forecasts. This has an advantage over using the forecast traces alone in that it provides the forecaster and forecast user a context with which to evaluate and understand the forecast. This point of reference information gives an indication whether the forecasted flow is characteristic of or unusual for the forecast point. Without past records, such as in case one, the ESP forecast can indicate only what streamflow values could be expected. It is difficult to know whether the forecast is predicting flows to be below, at, or above normal. A forecast user without personal or ancillary knowledge of expected values will have a difficult time relating this information to their specific applications. A final benefit to applying the second case to the ESP forecasts is that the verification statistics described in Section II can be utilized.

In this study, forecast categories were based on chosen percentiles of the historical distribution. To estimate the outer 1%, it is necessary to have 100 sample points. To estimate the outer 5%, at least 20 samples would be needed. Because the historical observations sets were so limited, it was impossible to use the same exceedance probability categories used by OHRFC. Only 10 data sets had records long enough to allow application of the verification statistics (Table 2). The minimum flow category that could be achieved was 15% (a minimum of 6 observations required) for 6 points and 25% (a minimum of 3 observations required) for 10 points.

Table 2: Study forecast points.

Minimum Flow Distribution Category	Forecast Point	Length of Historical Record
15% & 25%	BBVK2	8
	CMBK2	8
	ELKK2	7
	PKYK2	6
	PTVK2	6
	WLBK2	8
25%	DLYW2	3
	FLRK2	3
	PSTK2	5
	WRTO1	4

Two groups of forecasts were created: weekly mean stage forecasts with a 6-day lead time and maximum monthly stage forecasts with a 6-day lead time. The statistics from all forecast points were averaged as an assessment of forecast skill in the general study area of the OHRFC region. Four main analyses were applied to the forecasts: calculation of RPS without historical information; calculation of RPS with historical information and comparison to climatology forecasts (RPSS); reliability; and discrimination. Variations of the forecast probability and flow category sets were examined for RPS, RPSS, discrimination, and reliability based on the limitation in the previous paragraph. The variations are explicitly stated in the results section.

V. Results and Interpretation

RPS Only

The RPS was calculated for ESP forecasts using probability based on the empirical distribution of the traces. The probability categories were: .05, 1.0, .25, .5, .75, .90 and .95 non-exceedance. The RPS was calculated for each forecast and averaged by forecast point.

For mean weekly stage forecasts, the forecasts from FLRK2 and WRTO1 performed the best (smallest RPS values) (Figure 3). BBVK2 and WLBK2 forecasts performed the worst (highest RPS values). The best maximum monthly stage forecasts were issued for WRTO1 and CMBK2, while the worst forecasts were issued for PTVK2 and WLBK2 (Figure 4).

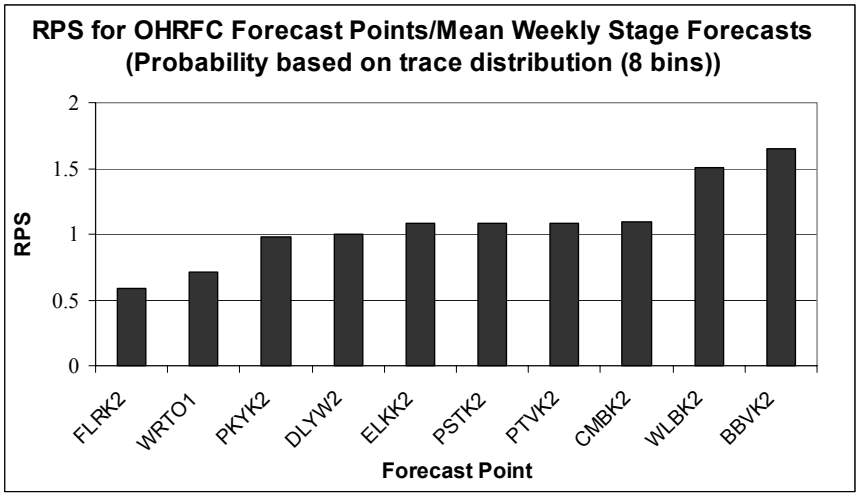

Figure 3: RPS analysis results for mean weekly stage forecasts.

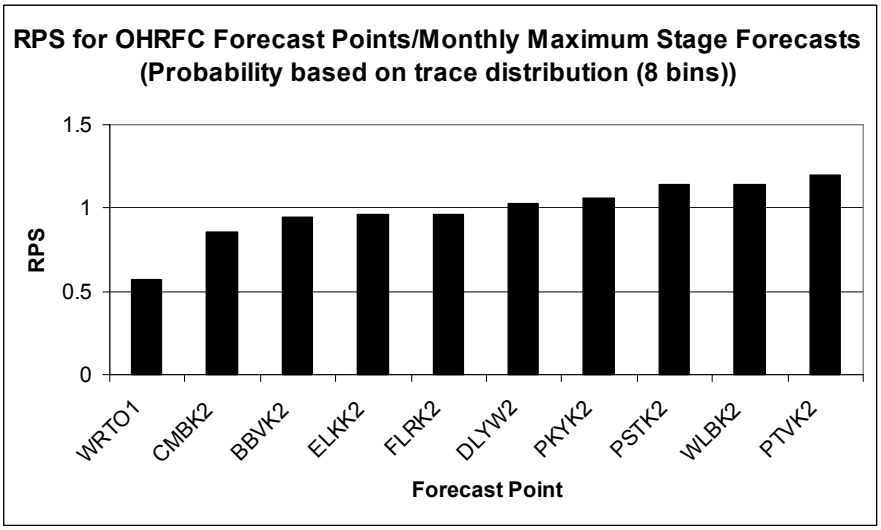

Figure 4: RPS analysis results for maximum monthly stage forecasts.

RPS scores for the 33 forecast points not included in this study are provided in Appendix III.

Because the actual RPS value is difficult to evaluate independently, the use of the RPS in the absence of reference forecasts is limited to forecast comparison among different forecast locations, as illustrated in this section. However, the RPS is useful in identifying regions where the predictions are not performing as well as others. This information can give forecasters an indication of where to conduct investigations into calibration, data quality control, event hydrology, or other factors contributing to the poorer forecast performance. Adjustments to the forecast system or subsequent forecasts can then be completed accordingly.

RPS and RPSS

In this section, the forecast probability was distributed according to categories defined by the historical climatology. The RPS for the forecasts and the streamflow climatology forecast were calculated. The performance of the ESP forecasts was then compared to climatology forecasts for the same location by calculating the skill score.

Historical observations were required for both development of ESP probability within streamflow categories and for generating climatology forecasts. Therefore, the forecast probability intervals were limited to available historical data. The following combinations of forecast categories was examined: (.25, .5, .75, and 1.0) and (.25, .75, and 1.0) for all 10, and (.15, .25, .5, .75, .85, and 1.0) and (.15, .85, and 1.0) for 6 of the ten. Any number of combinations of probability values can be created; therefore, some criteria needed to be defined to make the results more concise. The sets chosen here were based on the limitations of the observed data, the values used by the NWS in their forecasts files, and the number of intervals desired.

The average skill score for mean weekly stage forecasts for the four forecast categories is shown in Figure 5. The skill score is interpreted as percent improvement over climatology. The results indicate that the mean weekly stage ESP forecasts performed better on average than the climatology forecasts. In addition, forecast improvement was highest when the (.25, .75, and 1.0) probability intervals were used and lowest when the (.15, .25, .5, .75, .85, and 1.0) intervals were used. Forecasts that included the 15% outer categories displayed poorer performance than those where the smallest category was 25%.

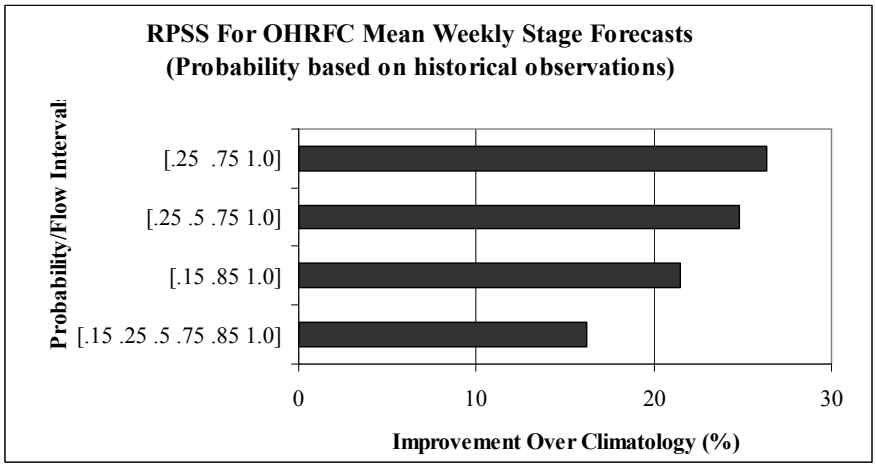

Figure 5: Ranked probability skill score for mean weekly stage forecasts.

Monthly maximum stage forecasts performed worse than the climatology forecasts (Figure 6). The forecasts that included a 15% outer category were much worse (more negative) than the forecast with only 25% categories. This indicates that either the forecasts have low skill when predicting small categories for monthly forecasts, or that the four forecast points not included in this average were significantly better than the remaining six.

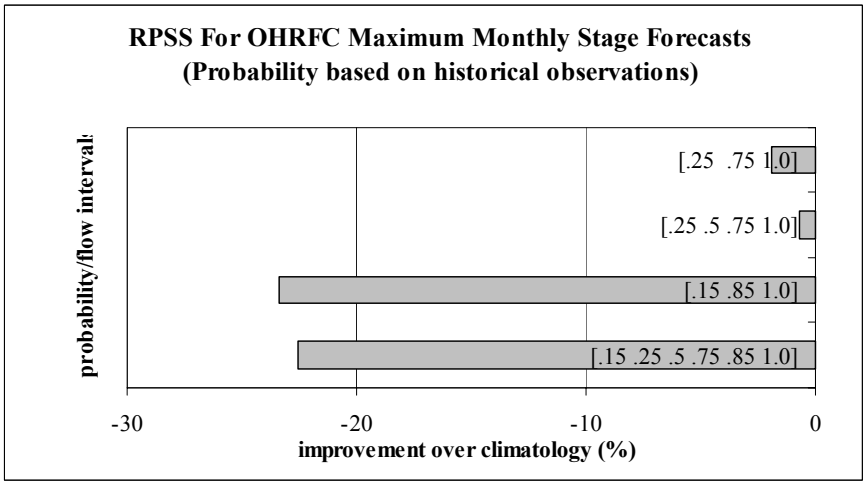

Figure 6: Ranked probability skill score for maximum monthly stage forecasts.

The RPS is sensitive to the number and size of forecast categories. Because the RPS is a measure of distance as well as probability value, fewer categories will result in less distance from the observations and a better score. This further illustrates that an RPS score alone is difficult to interpret.

With the absence of a reference forecast, such as climatology, it is difficult to understand the RPS value. The RPSS, therefore, is a more meaningful score because it can convey both the direction and degree of improvement over another forecast of

interest. The RPSS is also useful in assessing forecast improvement due to operational modifications of the forecast or forecast system such as pre-processing or post-processing, re-calibration, initial state modifications, and/or trace weighting.

Discrimination & Reliability

Discrimination and reliability analyses require historical observations. In this applications, the forecast flow categories considered were the lowest 25%, middle 50%, and highest 25% of the historical distribution. For 6 forecast points, the forecasts were re-examined with the outer 15% category and middle 70% categories. Because the discrimination and reliability diagrams become difficult to read when more than three flow categories are displayed at once, no more than three categories were used.

In the case of discrimination and reliability, the probability categories do not need to match the flow categories as in the RPS calculation. Experimentations with different probabilities categories indicated that the statistical information was best displayed with five or more intervals. Therefore, 7 forecast probability categories were applied to the discrimination and reliability diagrams (0%, >0-10%, >10-25%, >25-50%, >50-75%, >75- 90%, >90-100%).

Forecasts of mean weekly stage displayed good discrimination for predicting flows in the middle 50% of the historical distribution (Figure 7). Forecasts indicated a decreased likelihood of high flows occurring prior to observations in the lowest 25%. Discrimination was very good when predicting flows in the middle 70% of the distribution (Figure 8); however, this category is rather large and may not be useful to water managers and decision makers. The forecasts displayed no discrimination for predicting flows in the highest 25%, highest 15%, and lowest 15% of the streamflow distribution.

Monthly maximum forecasts showed no discrimination for high or low flows of the various flow categories examined and only a small degree of discrimination for predicting the middle 50% flow category (Figure 9). However, the forecasts did discriminate well for the middle 70% (Figure 10).

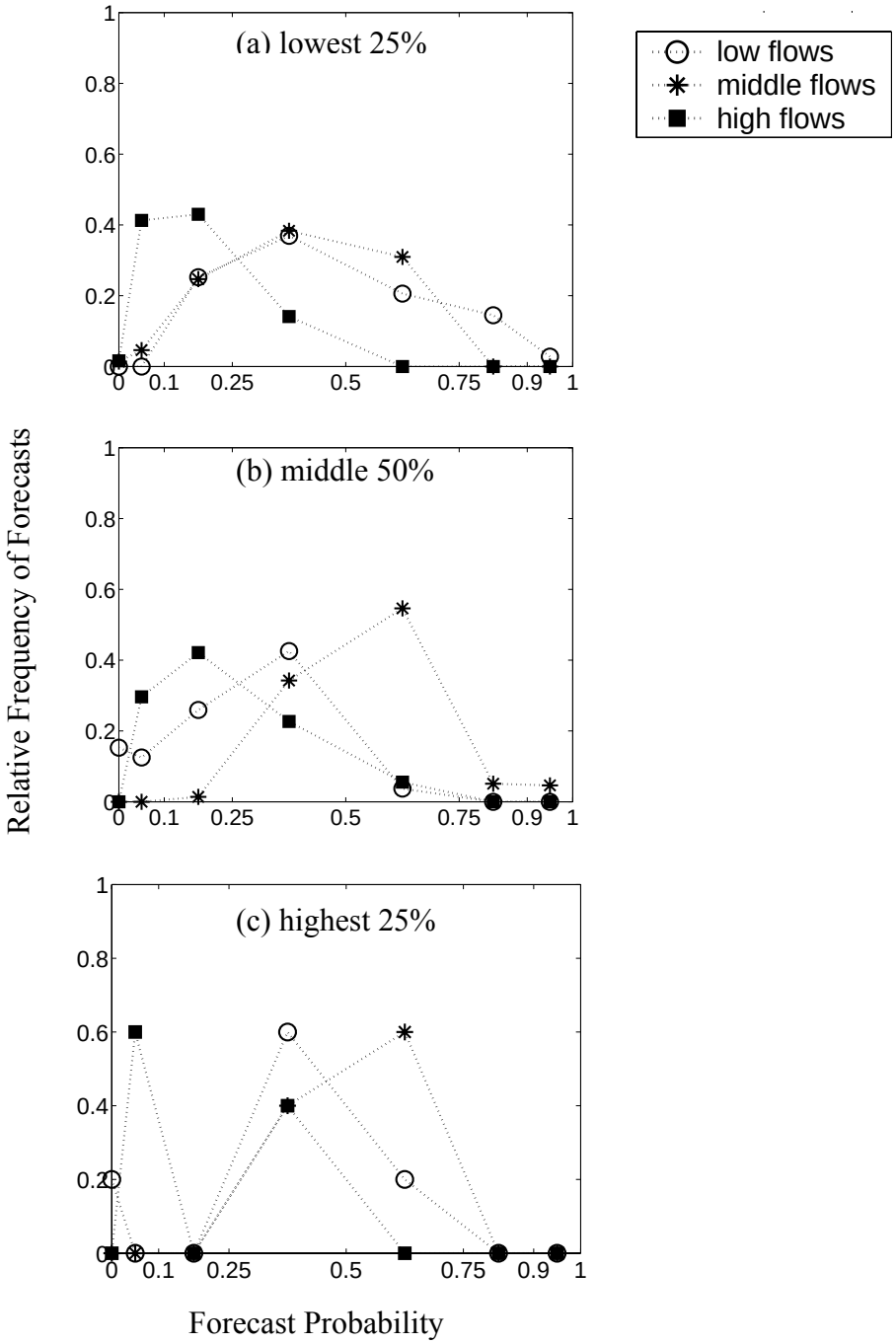

Figure 7: Discrimination diagrams for weekly mean stage forecasts for forecasts issued prior to observations in the (a) lowest 25%, (b) middle 50%, and (c) highest 25% of the historical distribution.

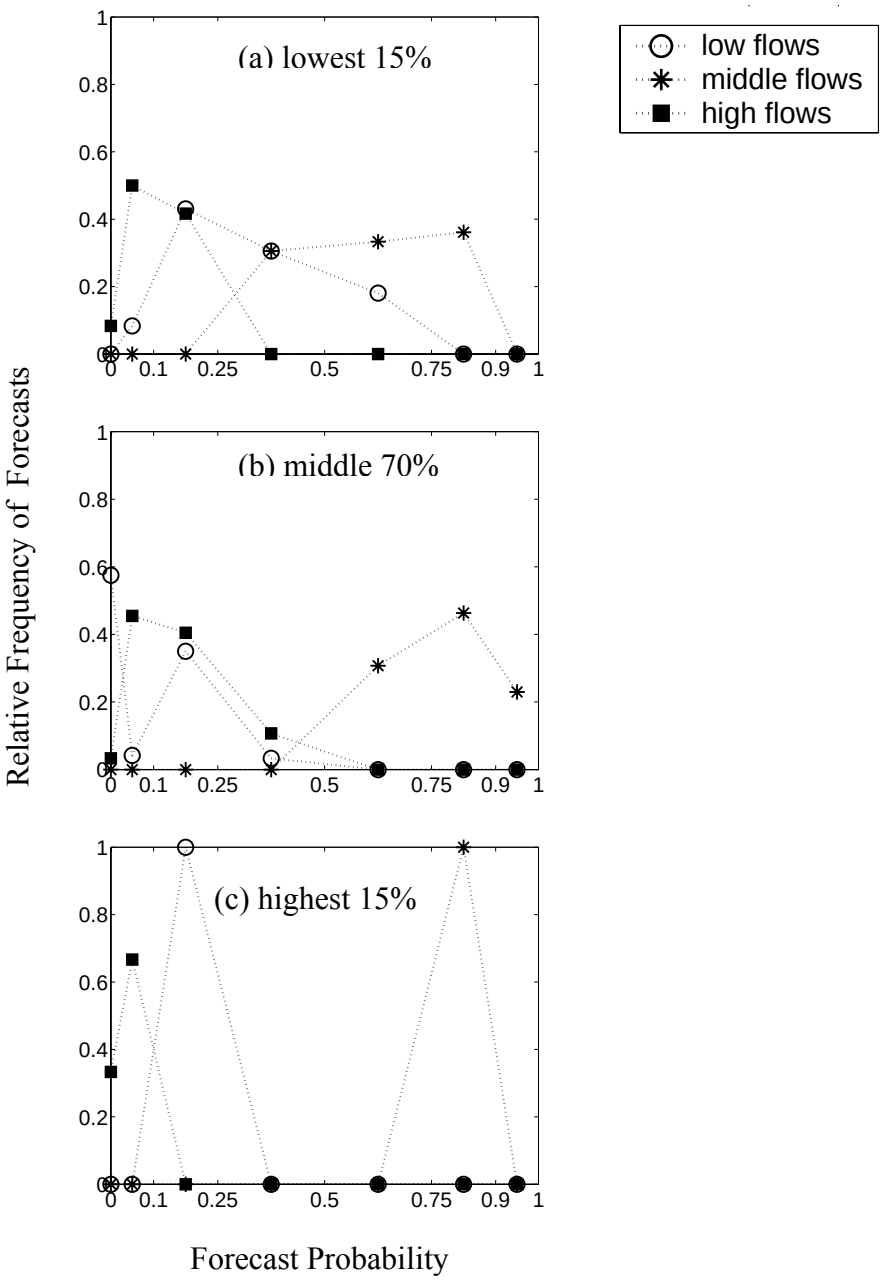

Figure 8: Discrimination diagrams for weekly mean stage forecasts for forecasts issued prior to observations in the (a) lowest 15 %, (b) middle 70%, and (c) highest 15% of the historical distribution.

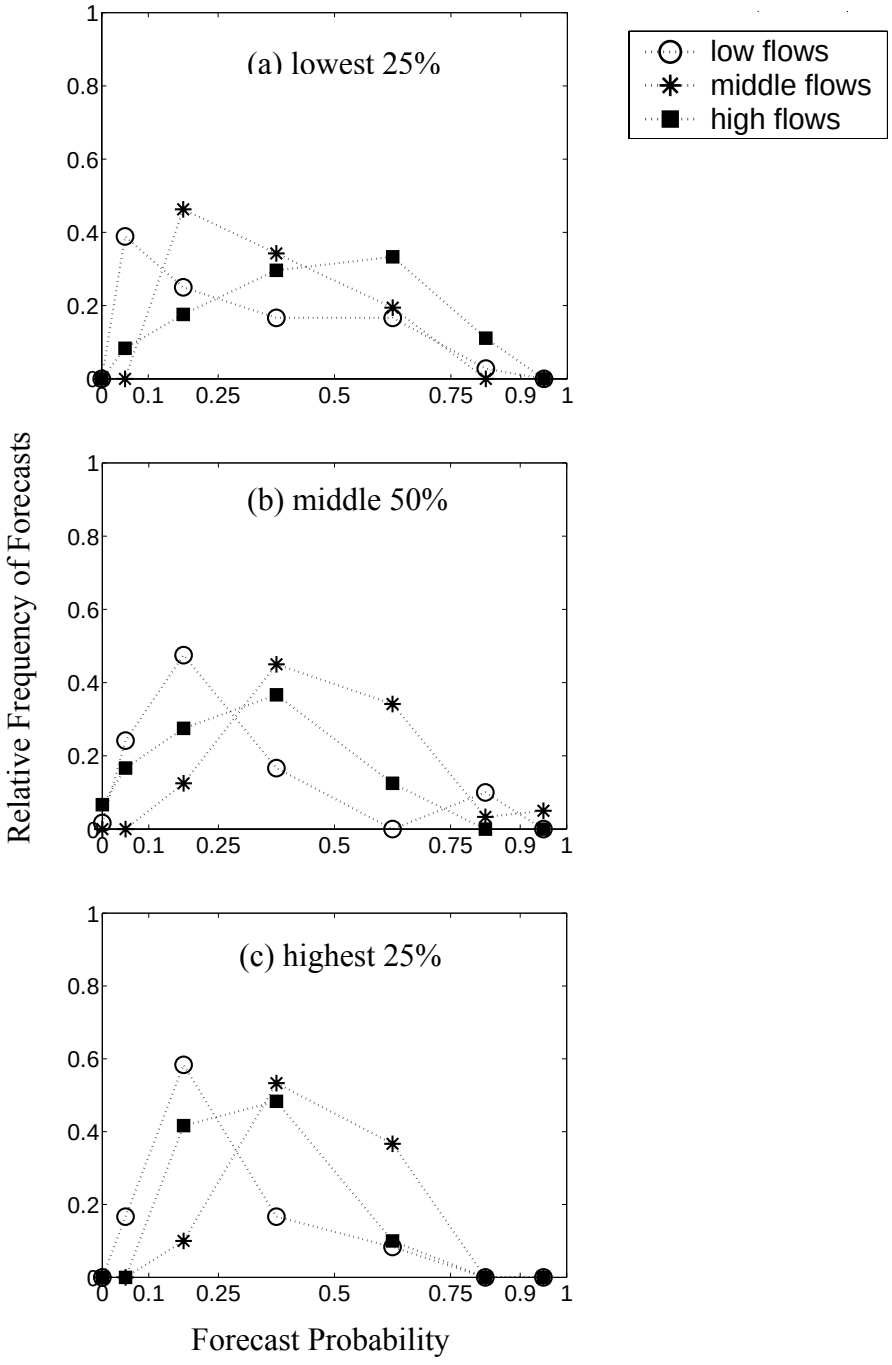

Figure 9: Discrimination diagrams for monthly maximum stage forecasts for forecasts issued prior to observations in the (a) lowest 25 %, (b) middle 50%, and (c) highest 25% of the historical distribution.

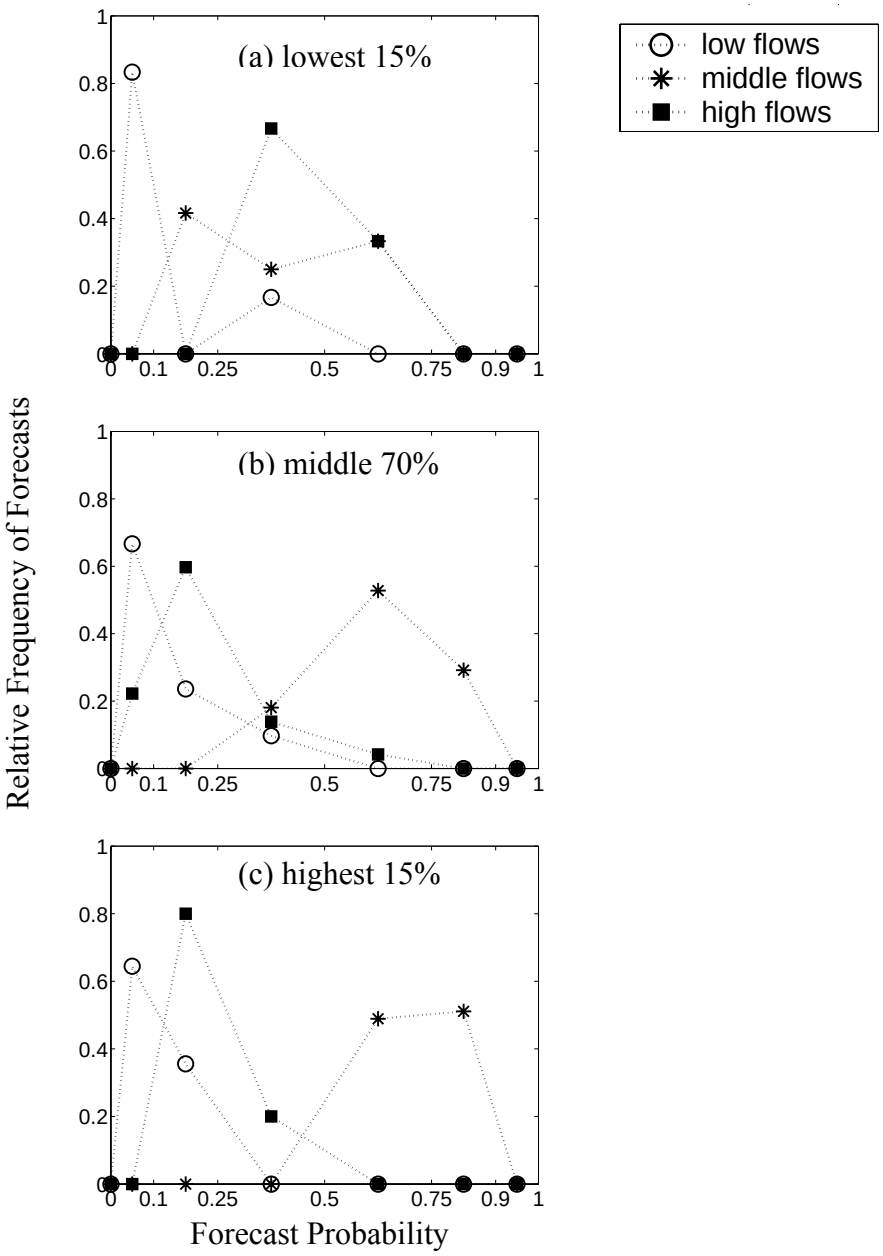

Figure 10: Discrimination diagrams for monthly maximum stage forecasts for forecasts issued prior to observations in the (a) lowest 15%, (b) middle 70%, and (c) highest 15% of the historical distribution.

Reliability diagrams shows that weekly mean forecasts underestimate both the lowest 25% (Figure 11) and lowest 15% flows categories (Figure 12). The ESP forecasts tended to overestimate the middle flows when high forecast probability values were assigned. The forecasts displayed no reliability for the highest 25% or highest 15% flow categories. Changes in the forecast performance between the middle 50% and middle 70% are most likely due to the most accurate forecasts switching categories at the different cutoff points, thereby changing the skill of the middle category.

Reliability performance for maximum stage forecasts was quite variable across the forecast probability values. Reliability was best for the lowest 25% and middle 50% categories when probability values between 10% and 75% were issued (Figure 13). They also showed some skill when predicting the highest 25% with probability below 50%. The forecasts show poor reliability for the highest 15% flow category (Figure 14). The forecast skill is variable over the range of forecast probabilities for predicting the lowest 15% and middle 70% flows.

The size of the data points on the reliability diagrams illustrates the relative frequency of the forecasts for each forecast probability category (bin). For better illustration, the bin value was transformed where point size = $2([(\text{bin size} +1)^{\lambda}-1]/\lambda)$ and $\lambda = 0.6$. The relative frequency of the forecasts indicates the degree of refinement. A forecasting system that more often predicts with extreme forecast probability values such as 0 or 100% is considered to produce sharp forecasts. The ESP stage forecasts do not illustrate a great degree of refinement.

The number of observations that fall within the various categories can modify the results. If the forecast system has skill, the chances of revealing this skill through the use of discrimination and reliability will be greater when the sample size is larger. Statistical results are considered more reliable from a larger sample size. The relative frequency of the forecasts can give insight into what statistical information may be most valid based on the number of forecasts in each probability category.

The reliability diagrams indicate where biases occur in the forecasts and provide a direction of improvement that is required. Operationally, forecasters can use this tool to interpret their forecasts in light of the knowledge that they tend to over- or under-forecast at different parts of the flow distribution. Confidence in the statistical results will come with increased sample sizes and better historical observations. This will provide the justification for altering forecasts based on statistical results.

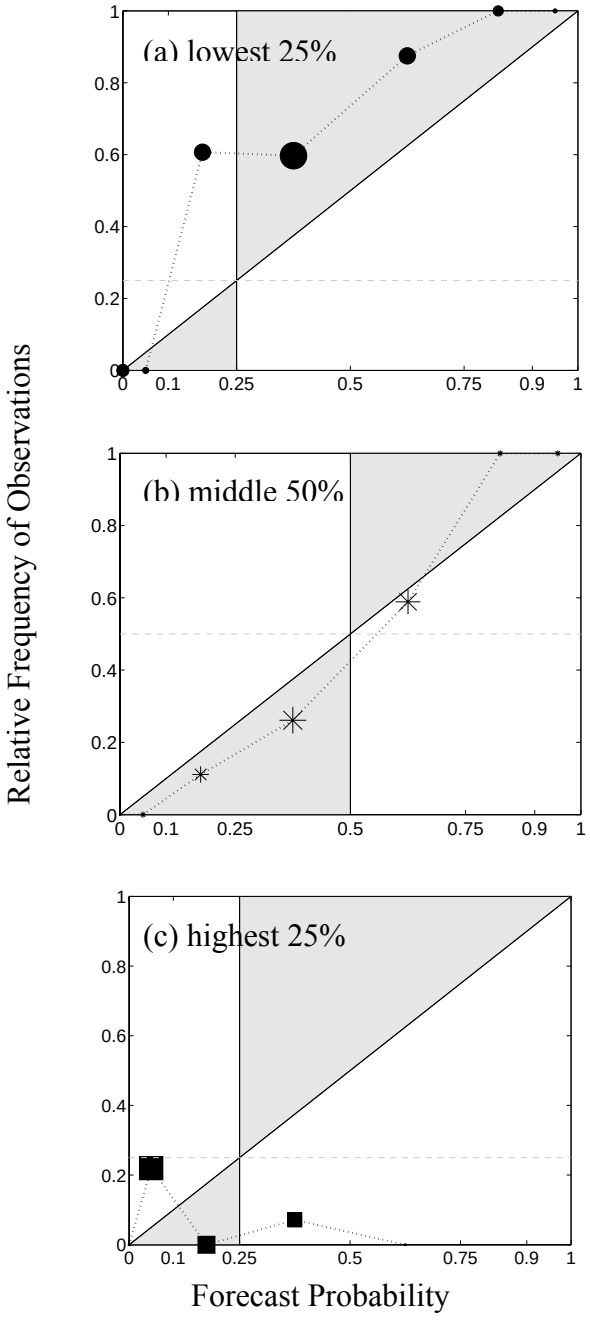

Figure 11: Reliability diagrams for weekly mean stage forecasts for forecasts issuing probability to the (a) lowest 25 %, (b) middle 50%, and (c) highest 25% of the historical distribution.

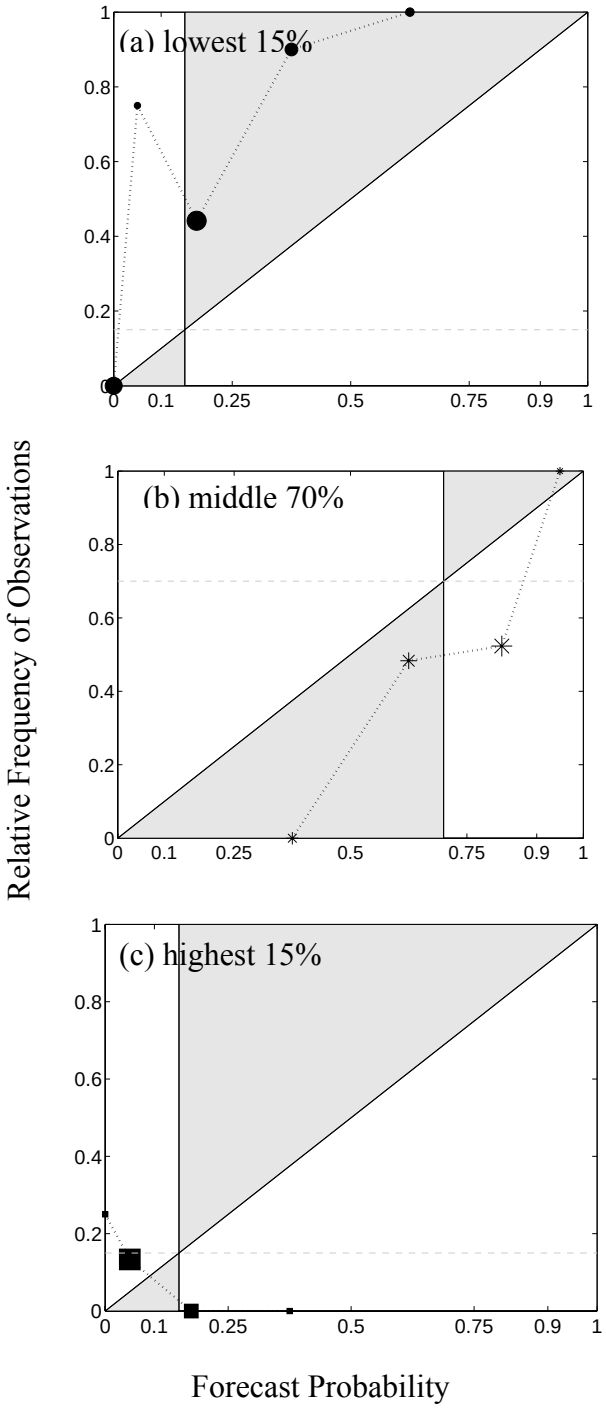

Figure 12: Reliability diagrams for weekly mean stage forecasts for forecasts issuing probability to the (a) lowest 15 %, (b) middle 70%, and (c) highest 15% of the historical distribution.

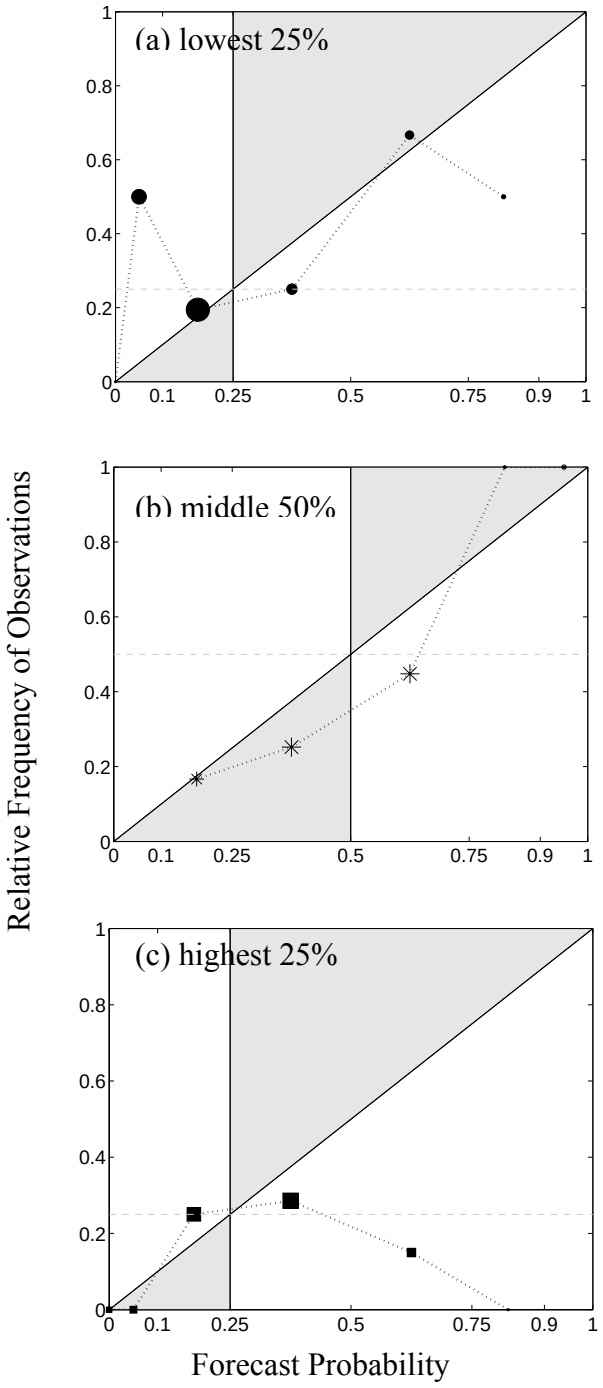

Figure 13: Reliability diagrams for monthly maximum stage forecasts for forecasts issuing probability to the (a) lowest 25 %, (b) middle 50%, and (c) highest 25% of the historical distribution.

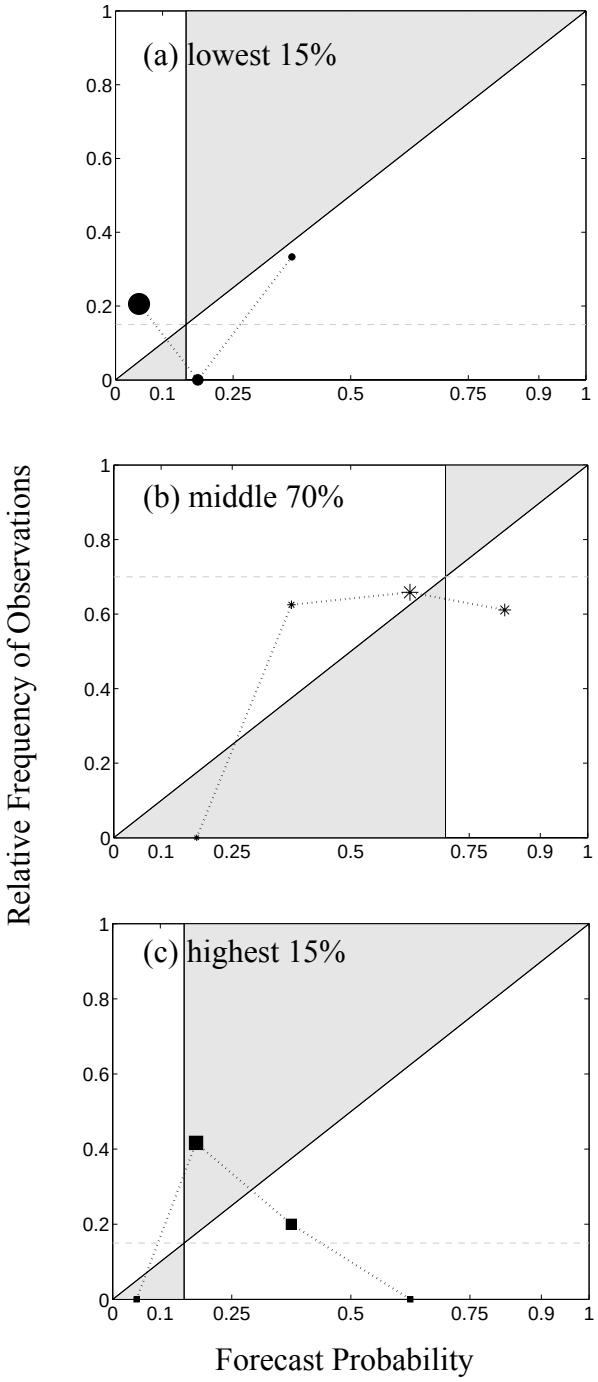

Figure 14: Reliability diagrams for monthly maximum stage forecasts for forecasts issuing probability to the (a) lowest 15 %, (b) middle 70%, and (c) highest 15% of the historical distribution.

VI. Discussion

To evaluate the ESP forecasts using the statistics applied in Franz (2001), the following data were needed:

- Forecast data
- Observation(s) over the forecast interval (termed the forecast observation)
- Historical observations or streamflow (stage) climatology

The forecast data should include the forecasted variable values and the associated probability. Current NWS products and data that would be useful are the ESP-generated forecast traces and the cumulative distribution graphs. Plots such as bar graphs would not contain the proper information to apply the verification statistics. Further investigation into current NWS ESP products that would be conducive to probabilistic forecast evaluation is needed.

The single most limiting factor in completing a thorough and confident forecast evaluation using the procedures proposed for this study was the missing observed data. The observations (both forecast and historical) should be derived from data spanning the same calendar days as the forecast. In addition, the time step of the observation records should be commensurate to the time step at which the forecast traces were generated. For example, the historical data available were mean daily stage. These data types are suitable for use in evaluating the mean weekly stage forecasts, but not ideal for evaluation of the monthly maximum forecasts. Because maximum daily stage observations were not available, the maximum monthly stage climatology needed to be calculated from the mean daily stage, resulting in climatology for monthly maximum mean stage.

The historical data used in the development of the streamflow climatology were not current; they generally included only information from the 1970s and 1980s. Data from more recent years may better represent the current hydrologic systems. Possible changes in equipment, data collection techniques, data quality control, basin hydrology, and climate all challenge the validity of using only 20- to 30-year old data to describe the current flow regime.

The data issues mentioned above should be considered when drawing conclusions from the evaluations presented in this report. The lack of sufficient historical observations may be a major limiting factor in the NWS effort to fully verify probabilistic forecasts at this time using methods outlined and applied in Franz (2001), unless a feasible alternative method of obtaining climatology distributions is determined. Discrimination and reliability could be used without a distribution if critical flow or stage levels were defined such as above or below flood stage. In this case, only the flood stage value would needed.

Probabilistic forecast evaluations conducted without past flow records provide limited information. From a user's perspective, the past distribution is a means by which a forecast can be put into context. In other words, the forecast user may gain more useful information about anticipated flows from a forecast that gives an indication of the likelihood of having extreme low, extreme high, or normal flows rather than the likelihood of a specific flow value. Without characteristic knowledge of the basin, a forecast user would not know if a prediction for a particular flow value indicated flood

conditions or drought conditions. By using past observations, a categorical forecast can be generated that may be more informative to a user unfamiliar with the basin. Application of RPSS, discrimination, and reliability as illustrated in this report will provide the forecaster and forecast user insight into forecast skill for predicting different flow conditions. This information is valuable to a decision maker when deciding whether or not to rely on the forecast. The forecaster can use this information to indicate confidence in the predictions and to adjust them accordingly.

The methods presented here will allow the NWS to track ESP forecast performance in both time and space. It will also allow for identification of needed improvements and regions where ESP may not work well. Additionally, forecast quality may change with season, climate state, or length of historical record. Once implemented, the verification system will allow comparison of forecast performance for different regions, for different events, and for changes in the forecasting system. Continual evaluations will allow forecasters to evaluate their current forecasts based on past performances and make adjustments accordingly. Comparison of forecast points will allow insight into why ESP works better in some regions than in others. In addition, the forecast verification will provide a basis for making adjustments and improvements to the forecast system. Scores such as the RPSS lend themselves to assessing forecast skill changes resulting from pre-processing, post-processing, and system modifications activities.

The statistical measures are also flexible and would allow forecasters (and users) to analyze streamflow levels of interest. Forecast probability intervals can also be changed. However, some analytical consistency is required as illustrated in the changes in RPSS due to category size. Finally, the statistics can be updated as new forecasts are generated providing long-term average forecast performance information.

VII. Conclusions

Traditionally, forecast evaluations have focused on dichotomous outcomes, whether the event occurred or not. However, this type of analysis is not sufficient or proper in the evaluation of probabilistic forecasts because the traditional verification methods cannot directly evaluated forecast probability. Probabilistic forecasts require verification methods that can assess the degree to which the forecasts apply probability to the subsequently observed event.

This “proof of concept study” was a necessary step to support incorporation of probabilistic evaluation methods into the NWSRFS. The results of this study support the recommendation that the NWS begin implementing the RPS, RPSS, discrimination, and reliability verification procedures in their ESP evaluation procedures. These methods were successfully applied to mean weekly stage and maximum monthly stage forecasts obtained from the Ohio River Forecasting Center (OHRFC). While the evaluation presented here only included stage forecast, the probabilistic forecast evaluation methods described in this document can be applied to any predicted streamflow variable given the appropriate forecast and observation data. The methods allow detailed evaluation of a variety of information contained within the ESP forecasts. The method can be designed to emphasize events of interest and concern.

The ESP system has been available to RFCs for close to 15 years and has been implemented on a limited basis at some RFCs. Within the next several years, it is anticipated that ESP will become more widely used. Implementation of ESP verification methods will allow forecasters and users to begin developing an understanding of the usefulness and limitations of the system.

VIII. Recommendations

Forecast verification methods have been successfully applied to operational NWS ESP forecasts; therefore, based on this study, the following recommendations are being made:

(1) Three verification measures are recommended for evaluation of NWS ESP forecasts:

- ranked probability score (RPS) (from which is derived the ranked probability skill score (RPSS)) (Epstein, 1969, and Wilks, 1995),
- discrimination (Murphy and Winkler, 1987, Murphy et al., 1989, and Wilks, 1995), and
- reliability ((Murphy and Winkler, 1987, Murphy et al., 1989, and Wilks, 1995).

It has been illustrated that these methods are well-suited for probabilistic streamflow forecasts verification.

(2) In order to implement the recommended evaluation techniques, it is necessary to have historical streamflow data to characterize the hydrologic system of the forecast basin and to develop comparison forecasts. Because sufficient climate data may be lacking in some regions, it is recommended that research be conducted to investigate other sources of the information while observations are accumulating. The following suggestions supply a basis for such an investigation.

Calculation of the skill score requires a comparison forecast against which the ESP forecast is evaluated. In this study, the ESP forecasts were evaluated against climatology forecasts developed from climate data. The flow categories were also defined based on this data. Insufficient climate data requires the use of alternative information for use in calculation of the skill score such as model climatology generated from historical temperature and precipitation data, persistence, or regression forecasts. Climatology for the development of flow categories could possibly be derived from a forecaster's personal knowledge, USGS rating curves (back-calculating from streamflow records), or USGS return flow equations. Data are also published yearly by the USGS for every gauge station; however, digitization of these data would be extremely time-consuming.

Finally, the verification methods recommended have been applied to hydrologic forecasts with medium to long (week to months) forecast windows. They have not been tested on forecasts with shorter intervals, such as daily flows. Further investigation into the feasibility of using these methods for such forecasts is recommended.

(3) The two forecast development methods showed two ways to think about hydrologic forecast probability. One included only trace distribution information from which the RPS was calculated; the other used historical observations to shift forecast probability between defined flow categories. **The forecast data required are primarily the issued forecast traces or probability distribution functions.** Archiving of the traces provides the evaluator with the ability to apply a much broader variety of analyses; therefore, it is recommended that the traces always be archived when an ESP forecast is produced. However, the NWS may also want to archive the following data for use in re-analysis studies:

- Pre- and post-processing information
- Initial conditions
- State updates
- Run-time modifications
- Trace distribution type
- Original traces (pre-modifications)

(4) These evaluation techniques rely on past records to generate and evaluate the forecasts; a poor record imparts obvious limitations to these techniques. Thus, the NWS RFCs should begin archiving streamflow data at the appropriate time interval and locations so that a proper data archive can begin to be developed. The following data are required:

- Observations from the forecast year for forecast period
- Historical observations for forecast period

Observations should be commensurate with the forecasted variable type, i.e., stage, discharge, etc.

The RFCs may also wish to archive the following data:

- Rating curves
- Quantitative precipitation forecasts (QPFs)

IX. Acknowledgements

This study was supported by the National Weather Service under Grant # 40-AA-NW-217447.

X. References

Day, G.N., 1985: Extended streamflow forecasting using NWSRFS. *Journal of Water Resources Planning and Management*, 111(2),157-170.

Franz, K.J., 2001: Evaluation of National Weather Service Ensemble Streamflow Prediction (ESP) Water Supply Forecasts. Master's Thesis, Department of Hydrology and Water Resources, University of Arizona, Tucson, Arizona.

Epstein, E.S., 1969: A Scoring System for Probability Forecasts of Ranked Categories. *Journal of Applied Meteorology*, 8, 985-987.

Hartmann, H.C., R. Bales, and S. Sorooshian, 1999: Weather, Climate, and Forecasting for the Southwest U.S., Report Series: CL2-99. Institute for the Study of Planet Earth, University of Arizona, 172 p.

Murphy, A.H. and R.L. Winkler, 1992: Diagnostic verification of probability forecasts. *International Journal of Forecasting*, 7, 435-455.

Murphy, A.H., B.G. Brown, and Y. Chen, 1989: Diagnostic verification of temperature forecasts. *Weather and Forecasting***,** 4, 485-501.

Murphy, A. H. and R.L. Winkler, 1987: A general framework for forecast verification. *Monthly Weather Review*, 115**,** 1330-1338.

Wilks, D.S., 2000: Diagnostic Verification of the Climate Prediction Center Long-Lead Outlooks, 1995-98. *Journal of Climate*, 13, 2389-2403.

Wilks, D.S., 1995: Forecast verification. Statistical Methods in the Atmospheric Sciences, Academic Press, 467 p.

Appendix I

Example MATLAB scripts for calculation of RPSS and RPS

```
%Script to calculate ranked probability score and ranked probability skill scores for ESP
%forecasts
%Kristie Franz
%University of Arizona
%June, 2002
%For more information contact Kristie at kristie@hwr.arizona.edu

clear all;
close all;

var = input('Enter forecast point name: \n', 's');
fname1 = sprintf('%s%s%s', 'results/',var,'SS_all_wk_mo_emp1');
fname2 = sprintf('%s%s%s', 'results/',var,'SS_rps_wk_mo_emp1');

fid1 = fopen(fname1,'w');
fid2 = fopen(fname2,'w');

load obs_mean_max;
load traces_mean;
load traces_max;
load obs_mean_max;
load wkly_mean_stage;
load mnthly_max_stage;
load fcst_mean;

[z C] = size(obs_mean_max);

quants = 6;
SS_all = zeros(2,z);

for cycle = 1:2
    clear rps_y;
    clear rps_o;
    clear rps_y3;
    clear rps_o3;
    clear rps_y2;
    clear rps_o2;
    clear rps_y4;
    clear rps_o4;
    clear rps_temp_f;
    clear rps_temp_o;
    clear rps_temp_f3;
    clear rps_temp_o3;
    clear rps_temp_f2;
    clear rps_temp_o2;
    clear rps_temp_f4;
    clear rps_temp_o4;
    clear cum_fcasts;
    clear cum_obs;
    clear cum_fcasts3;
    clear cum_obs3;
    clear cum_fcasts2;
    clear cum_obs2;
    clear cum_fcasts4;
    clear cum_obs4;
    clear prob_fcasts;
    clear prob_obs;

    ignore = 0;
        ct = 0;
        ct2=0;
    if cycle == 1
                [y B] = size(wkly_mean_stage);
                [x A] = size(traces_mean);
                forecasts = traces_mean;
                prob_obs = zeros(y,quants);
                quantiles = zeros(y,quants);
                obs_col = 1;
                temp_obs=wkly_mean_stage;
    else
                [y B] = size(mnthly_max_stage);
                [x A] = size(traces_max);
                forecasts = traces_max;
                prob_obs = zeros(y,quants);
                quantiles = zeros(y,quants);
                obs_col = 2;
                temp_obs=mnthly_max_stage;
      end

    for yr = 1:y
        clear observed;
```

```
        if obs_mean_max(yr,obs_col)~=-9999
                observed = -9999;
                temp_k = 0;
                for u = 1:B
                    if temp_obs(yr,u)~=-9999
                        temp_k = temp_k+1;
                        observed(temp_k) = temp_obs(yr,u);
                    end
                end

            obs = sort(observed');
            ct3 = 0;
            for i = 1:length(obs)
                ct3 = ct3+1;
                obs(i,2) = (ct3/(length(obs)+1));
            end

            emp15 = interp1(obs(:,2),obs(:,1),.15);
            emp25 = interp1(obs(:,2),obs(:,1),.25);
            emp50 = interp1(obs(:,2),obs(:,1),.5);
            emp75 = interp1(obs(:,2),obs(:,1),.75);
            emp85 = interp1(obs(:,2),obs(:,1),.85);

                   for i = 1:x
                      if forecasts(i,yr) <= emp15
                           quantiles(yr,1) = quantiles(yr,1) + 1;
                      elseif forecasts(i,yr) <= emp25
                           quantiles(yr,2) = quantiles(yr,2) + 1;
                      elseif forecasts(i,yr) <= emp50
                           quantiles(yr,3) = quantiles(yr,3) + 1;
                      elseif forecasts(i,yr) <= emp75
                           quantiles(yr,4) = quantiles(yr,4) + 1;
                      elseif forecasts(i,yr) <= emp85
                           quantiles(yr,5) = quantiles(yr,5) + 1;
                      else
                           quantiles(yr,6) = quantiles(yr,6) + 1;
                       end
                   end

            if yr<=z
                if obs_mean_max(yr,obs_col) <= emp15
                    prob_obs(yr,1) = 1;
                elseif obs_mean_max(yr,obs_col) <= emp25
                    prob_obs(yr,2) = 1;
                elseif obs_mean_max(yr,obs_col) <= emp50
                    prob_obs(yr,3) = 1;
                elseif obs_mean_max(yr,obs_col) <= emp75
                    prob_obs(yr,4) = 1;
                elseif obs_mean_max(yr,obs_col) <= emp85
                    prob_obs(yr,5) = 1;
                else
                    prob_obs(yr,6) = 1;
                end
            end

                   for t = 1:quants
                       prob_fcasts(yr,t) = quantiles(yr,t)/x;
                   end
        else
            prob_obs(yr,1:quants) = 0;
            prob_fcasts(yr,1:quants) = 0;
           ct = ct+1;
            ignore(ct) = yr;
        end
end

        climo = [.15 .25 .50 .75 .85 1.0];
        cum_fcasts = cumsum(prob_fcasts,2);
        cum_obs = cumsum(prob_obs,2);

        climo3 = [.25 .75 1.0];
        cum_fcasts3(:,1) = cum_fcasts(:,2);
        cum_fcasts3(:,2) = cum_fcasts(:,4);
        cum_fcasts3(:,3) = cum_fcasts(:,6);
        cum_obs3(:,1) = cum_obs(:,2);
        cum_obs3(:,2) = cum_obs(:,4);
        cum_obs3(:,3) = cum_obs(:,6);

       climo2 = [.15 .85 1.0];
       cum_fcasts2(:,1) = cum_fcasts(:,1);
```

```
       cum_fcasts2(:,2) = cum_fcasts(:,5);
       cum_fcasts2(:,3) = cum_fcasts(:,6);
       cum_obs2(:,1) = cum_obs(:,1);
       cum_obs2(:,2) = cum_obs(:,5);
       cum_obs2(:,3) = cum_obs(:,6);

       climo4 = [.25 .5 .75 1.0];
       cum_fcasts4(:,1) = cum_fcasts(:,2);
       cum_fcasts4(:,2) = cum_fcasts(:,3);
       cum_fcasts4(:,3) = cum_fcasts(:,4);
       cum_fcasts4(:,4) = cum_fcasts(:,6);
       cum_obs4(:,1) = cum_obs(:,2);
       cum_obs4(:,2) = cum_obs(:,3);
       cum_obs4(:,3) = cum_obs(:,4);
       cum_obs4(:,4) = cum_obs(:,6);

for i = 1:yr
       if i~=ignore
           rps_y(i) = (sum((cum_fcasts(i,:)-cum_obs(i,:)).^2));
           rps_o(i) = (sum((climo(1,:)-cum_obs(i,:)).^2));
           SS_all(obs_col,i) = (rps_y(i)-rps_o(i))/(0-rps_o(i));
           rps_y3(i) = (sum((cum_fcasts3(i,:)-cum_obs3(i,:)).^2));
           rps_o3(i) = (sum((climo3(1,:)-cum_obs3(i,:)).^2));
           SS_all3(obs_col,i) = (rps_y3(i)-rps_o3(i))/(0-rps_o3(i));
           rps_y2(i) = (sum((cum_fcasts2(i,:)-cum_obs2(i,:)).^2));
           rps_o2(i) = (sum((climo2(1,:)-cum_obs2(i,:)).^2));
           SS_all2(obs_col,i) = (rps_y2(i)-rps_o2(i))/(0-rps_o2(i));
           rps_y4(i) = (sum((cum_fcasts4(i,:)-cum_obs4(i,:)).^2));
           rps_o4(i) = (sum((climo4(1,:)-cum_obs4(i,:)).^2));
           SS_all4(obs_col,i) = (rps_y4(i)-rps_o4(i))/(0-rps_o4(i));

fprintf(fid1,' %1.4f %1.4f  %1.4f %1.4f\n', SS_all(obs_col,i), SS_all3(obs_col,i),
SS_all2(obs_col,i), SS_all4(obs_col,i));

       else
           rps_y(i) = -9999;
           rps_o(i) = -9999;
           SS_all(obs_col,i)=-9999;
           rps_y3(i) = -9999;
           rps_o3(i) = -9999;
           SS_all3(obs_col,i)=-9999;
           rps_y2(i) = -9999;
           rps_o2(i) = -9999;
           SS_all2(obs_col,i)=-9999;
           rps_y4(i) = -9999;
           rps_o4(i) = -9999;
           SS_all4(obs_col,i)=-9999;

fprintf(fid1,' %1.4f %1.4f  %1.4f %1.4f\n', SS_all(obs_col,i), SS_all3(obs_col,i),
SS_all2(obs_col,i), SS_all4(obs_col,i));

       end
end

       fprintf(fid1,' \n ');

for i = 1:yr
        if rps_y(i) ~= -9999
          ct2 = ct2+1;
          rps_temp_f(ct2) = rps_y(i);
          rps_temp_o(ct2) = rps_o(i);
          rps_temp_f3(ct2) = rps_y3(i);
          rps_temp_o3(ct2) = rps_o3(i);
          rps_temp_f2(ct2) = rps_y2(i);
          rps_temp_o2(ct2) = rps_o2(i);
          rps_temp_f4(ct2) = rps_y4(i);
          rps_temp_o4(ct2) = rps_o4(i);
       end

       rps_fcpt(obs_col) = mean(rps_temp_f);
       rps_ob_yr(obs_col) = mean(rps_temp_o);
       rps_fcpt3(obs_col) = mean(rps_temp_f3);
       rps_ob_yr3(obs_col) = mean(rps_temp_o3);
       rps_fcpt2(obs_col) = mean(rps_temp_f2);
       rps_ob_yr2(obs_col) = mean(rps_temp_o2);
       rps_fcpt4(obs_col) = mean(rps_temp_f4);
       rps_ob_yr4(obs_col) = mean(rps_temp_o4);
end
```

```
    SS(obs_col) = (rps_fcpt(obs_col)-rps_ob_yr(obs_col))/(0-rps_ob_yr(obs_col));
    SS3(obs_col) = (rps_fcpt3(obs_col)-rps_ob_yr3(obs_col))/(0-rps_ob_yr3(obs_col));
    SS2(obs_col) = (rps_fcpt2(obs_col)-rps_ob_yr2(obs_col))/(0-rps_ob_yr2(obs_col));
    SS4(obs_col) = (rps_fcpt4(obs_col)-rps_ob_yr4(obs_col))/(0-rps_ob_yr4(obs_col));
end


fprintf(fid2,'%1.4f %1.4f %2.4f %2.4f \n',SS, rps_fcpt);
fprintf(fid2,'%1.4f %1.4f %2.4f %2.4f \n',SS3, rps_fcpt3);
fprintf(fid2,'%1.4f %1.4f %2.4f %2.4f \n',SS2, rps_fcpt2);
fprintf(fid2,'%1.4f %1.4f %2.4f %2.4f \n',SS4, rps_fcpt4);
```

```
%Script to calculate ranked probability score for ESP forecasts
%Kristie Franz
%University of Arizona
%June, 2002
%For more information contact Kristie at kristie@hwr.arizona.edu

clear all;
close all;

var = input('Enter forecast point name: \n', 's');
fname2 = sprintf('%s%s%s', 'results/',var,'RPS_avg_wk_mo_emp2');
fid2 = fopen(fname2,'w');

load obs_mean_max;

quants = 8;
[z C] = size(obs_mean_max);

for cycle = 1:2

    clear forecasts;
    clear prob_obs;
    clear cum_obs1; clear cum_obs2; clear cum_obs3; clear cum_obs5;
    clear cum_fcasts1; clear cum_fcasts2; clear cum_fcasts3; clear cum_fcasts5;
    clear rps_y1; clear rps_y2; clear rps_y3; clear rps_y5;
       clear rps_temp_f1; clear rps_temp_f2; clear rps_temp_f3; clear rps_temp_f5;
    ignore = 0;
       ct = 0;
       ct2=0;
    if cycle == 1
              load fcst_mean;
        forecasts=fcst_mean;
        prob_obs = zeros(z,quants);
        obs_col = 1;
        fname1 = sprintf('%s%s%s', 'results/',var,'RPS_all_week_emp2');
        fid1 = fopen(fname1,'w');
    else
        fclose(fid1);
              load fcst_max;
        forecasts=fcst_max;
              prob_obs = zeros(z,quants);
              obs_col = 2;
        fname1 = sprintf('%s%s%s', 'results/',var,'RPS_all_mon_emp2');
        fid1 = fopen(fname1,'w');
    end

    for yr = 1:z
        clear observed;
        if obs_mean_max(yr,obs_col)~=-9999
            observed = -9999;
            temp_k = 0;

            if yr<=z
                if obs_mean_max(yr,obs_col) <= forecasts(1,yr)
                    prob_obs(yr,1) = 1;
                elseif obs_mean_max(yr,obs_col) <= forecasts(2,yr)
                    prob_obs(yr,2) = 1;
                elseif obs_mean_max(yr,obs_col) <= forecasts(3,yr)
                    prob_obs(yr,3) = 1;
                elseif obs_mean_max(yr,obs_col) <= forecasts(4,yr)
                    prob_obs(yr,4) = 1;
                elseif obs_mean_max(yr,obs_col) <= forecasts(5,yr)
                    prob_obs(yr,5) = 1;
                elseif obs_mean_max(yr,obs_col) <= forecasts(6,yr)
                    prob_obs(yr,6) = 1;
                elseif obs_mean_max(yr,obs_col) <= forecasts(7,yr)
                    prob_obs(yr,7) = 1;
                else
                    prob_obs(yr,8) = 1;
                end
            end
```

```
        else
            prob_obs(yr,1:quants) = 0;
            ct = ct+1;
            ignore(ct) = yr;
        end
       end

       cum_obs1 = cumsum(prob_obs,2);
    cum_fcasts1 = [.05 .10 .25 .50 .75 .90 .95 1.0];

    cum_fcasts2 = [.1 .9 1.0];
    cum_obs2(:,1) = cum_obs1(:,2);
    cum_obs2(:,2) = cum_obs1(:,6);
    cum_obs2(:,3) = cum_obs1(:,8);

    cum_fcasts3 = [.25 .75 1.0];
    cum_obs3(:,1) = cum_obs1(:,3);
    cum_obs3(:,2) = cum_obs1(:,5);
    cum_obs3(:,3) = cum_obs1(:,8);

    cum_fcasts5 = [.1 .25 .5 .75 .9 1.0];
    cum_obs5(:,1) = cum_obs1(:,2);
    cum_obs5(:,2) = cum_obs1(:,3);
    cum_obs5(:,3) = cum_obs1(:,4);
    cum_obs5(:,4) = cum_obs1(:,5);
    cum_obs5(:,5) = cum_obs1(:,6);
    cum_obs5(:,6) = cum_obs1(:,8);

       for i = 1:yr
        if i~=ignore
            rps_y1(i) = (sum((cum_fcasts1(1,:)-cum_obs1(i,:)).^2));
            rps_y2(i) = (sum((cum_fcasts2(1,:)-cum_obs2(i,:)).^2));
            rps_y3(i) = (sum((cum_fcasts3(1,:)-cum_obs3(i,:)).^2));
            rps_y5(i) = (sum((cum_fcasts5(1,:)-cum_obs5(i,:)).^2));
            fprintf(fid1,' %1.4f %1.4f %1.4f %1.4f\n', rps_y1(i), rps_y2(i),
rps_y3(i),rps_y5(i));
        else
            rps_y1(i) = -9999;
            rps_y2(i) = -9999;
            rps_y5(i) = -9999;
            rps_y3(i) = -9999;
            fprintf(fid1,' %1.4f %1.4f %1.4f %1.4f\n', rps_y1(i),rps_y2(i), rps_y3(i),
rps_y5(i));
        end
       end
       fprintf(fid1,' \n ');
       for i = 1:yr
        if rps_y1(i) ~= -9999
           ct2 = ct2+1;
           rps_temp_f1(ct2) = rps_y1(i);
           rps_temp_f2(ct2) = rps_y2(i);
           rps_temp_f3(ct2) = rps_y3(i);
           rps_temp_f5(ct2) = rps_y5(i);
        end
        rps_fcpt1(obs_col) = mean(rps_temp_f1);
        rps_fcpt2(obs_col) = mean(rps_temp_f2);
        rps_fcpt3(obs_col) = mean(rps_temp_f3);
        rps_fcpt5(obs_col) = mean(rps_temp_f5);


    end
end

fprintf(fid2,'%1.4f %1.4f  \n',rps_fcpt1);
fprintf(fid2,'%1.4f %1.4f \n', rps_fcpt2);
fprintf(fid2,'%1.4f %1.4f  \n', rps_fcpt3);
fprintf(fid2,'%1.4f %1.4f  \n', rps_fcpt5);
```

Appendix II

Example MATLAB script for calculation of discrimination and reliability

```
%Script to generate forecast discrimination and reliability statistics for ESP forecasts
%Kristie Franz
%University of Arizona
%June, 2002
%For more information contact Kristie at kristie@hwr.arizona.edu


clear;
close;
var1 = input('Enter forecast point name: \n', 's');
var = input('Enter 1 for mean flow forecast and 2 for max flow forecast: \n');
if var == 1
    load traces_mean;
    load wkly_mean_stage;
    hist_obs = wkly_mean_stage;
    forecasts = traces_mean;
    obs_col = 1;
    [y B] = size(wkly_mean_stage);
    [x A] = size(traces_mean);
    fname1 = sprintf('%s%s%s', 'results/',var1,'disc_wkly_emp');
    fname2 = sprintf('%s%s%s', 'results/',var1,'rel_wkly_emp');
    fname3 = sprintf('%s%s%s', 'results/',var1,'obs_freq_wkly_emp');
    fname4 = sprintf('%s%s%s', 'results/',var1,'fc_freq_wkly_emp');
    fid1=fopen(fname1,'w');
    fid2=fopen(fname2,'w');
    fid3=fopen(fname3,'w');
    fid4=fopen(fname4,'w');
else
    load traces_max;
    load mnthly_max_stage;
    hist_obs=mnthly_max_stage;
    forecasts = traces_max;
    [y B] = size(mnthly_max_stage);
    [x A] = size(traces_max);
    obs_col = 2;
    fname1 = sprintf('%s%s%s', 'results/',var1,'disc_mnthly_emp');
    fname2 = sprintf('%s%s%s', 'results/',var1,'rel_mnthly_emp');
    fname3 = sprintf('%s%s%s', 'results/',var1,'obs_freq_mnthly_emp');
    fname4 = sprintf('%s%s%s', 'results/',var1,'fc_freq_mnthly_emp');
    fid1=fopen(fname1,'w');
    fid2=fopen(fname2,'w');
    fid3=fopen(fname3,'w');
    fid4=fopen(fname4,'w');
end

load obs_mean_max;
[z C] = size(obs_mean_max);
quants = 6;
prob_obs = zeros(y,quants);
climo = [.1 .25 .5 .75 .9 1.0];
quantiles = zeros(y,quants);
ignore = 0;
ct = 0;

for yr = 1:y
    clear observed;
    if obs_mean_max(yr,obs_col)~=-9999
        clear temp_array;
        observed = -9999;
        temp_k = 0;
        for u = 1:B
            if hist_obs(yr,u)~=-9999
                temp_k = temp_k+1;
                observed(temp_k) = hist_obs(yr,u);
            end
        end

        obs = sort(observed');
        ct3 = 0;
            for i = 1:length(obs)
                ct3 = ct3+1;
                obs(i,2) = (ct3/(length(obs)+1));
            end
            emp10 = interp1(obs(:,2),obs(:,1),.10);
            emp25 = interp1(obs(:,2),obs(:,1),.25);
            emp50 = interp1(obs(:,2),obs(:,1),.5);
            emp75 = interp1(obs(:,2),obs(:,1),.75);
            emp90 = interp1(obs(:,2),obs(:,1),.90);
```

```
            for i = 1:x
             if forecasts(i,yr) <= emp10
                 quantiles(yr,1) = quantiles(yr,1) + 1;
             elseif forecasts(i,yr) <= emp25
                 quantiles(yr,2) = quantiles(yr,2) + 1;
             elseif forecasts(i,yr) <= emp50
                 quantiles(yr,3) = quantiles(yr,3) + 1;
             elseif forecasts(i,yr) <= emp75
                 quantiles(yr,4) = quantiles(yr,4) + 1;
             elseif forecasts(i,yr) <= emp90
                 quantiles(yr,5) = quantiles(yr,5) + 1;
             else
                 quantiles(yr,6) = quantiles(yr,6) + 1;
             end
            end


        if yr<=z
             if obs_mean_max(yr,obs_col) <= emp10
                 prob_obs(yr,1) = 1;
             elseif obs_mean_max(yr,obs_col) <= emp25
                 prob_obs(yr,2) = 1;
             elseif obs_mean_max(yr,obs_col) <= emp50
                 prob_obs(yr,3) = 1;
             elseif obs_mean_max(yr,obs_col) <= emp75
                 prob_obs(yr,4) = 1;
             elseif obs_mean_max(yr,obs_col) <= emp90
                 prob_obs(yr,5) = 1;
             else
                 prob_obs(yr,6) = 1;
             end
        end


            for t = 1:quants
             prob_fcasts(yr,t) = quantiles(yr,t)/x;
            end
    else
    prob_obs(yr,1:quants) = 0;
    prob_fcasts(yr,1:quants) = 0;
    ct = ct+1;
    ignore(ct) = yr;
    end
end

count_obs_1 = 0;
count_obs_2 = 0;
count_obs_3 = 0;
red_1 = [0 0 0 0 0 0 0];
green_1 = [0 0 0 0 0 0 0];
blue_1 = [0 0 0 0 0 0 0];
red_2 = [0 0 0 0 0 0 0];
green_2 = [0 0 0 0 0 0 0];
blue_2 = [0 0 0 0 0 0 0];
red_3 = [0 0 0 0 0 0 0];
green_3 = [0 0 0 0 0 0 0];
blue_3 = [0 0 0 0 0 0 0];

fc_red = [0 0 0 0 0 0 0];
obs_red = [0 0 0 0 0 0 0];
fc_green = [0 0 0 0 0 0 0];
obs_green = [0 0 0 0 0 0 0];
fc_blue = [0 0 0 0 0 0 0];
obs_blue = [0 0 0 0 0 0 0];


for k = 1:yr
    if k ~= ignore
        if (sum(prob_obs(k,1:2))==1)
            count_obs_1= count_obs_1 + 1;
            if ((sum(prob_fcasts(k,1:2)))== 0)
                red_1(1) = red_1(1) + 1;
            elseif (0 < (sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .1)
                red_1(2) = red_1(2) + 1;
            elseif (.1 < (sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2)))<=.25)
                red_1(3) = red_1(3) + 1;
            elseif (.25 < (sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .5)
                red_1(4) = red_1(4) + 1;
            elseif (.5 < (sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .75)
                red_1(5) = red_1(5)+ 1;
            elseif (.75 < (sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .90)
```

```
        red_1(6) = red_1(6)+ 1;
    elseif (.9 < (sum(prob_fcasts(k,1:2))))
        red_1(7) = red_1(7) + 1;
    end

    if (sum(prob_fcasts(k,3:4)) == 0)
        green_1(1) = green_1(1) + 1;
    elseif (0 < (sum(prob_fcasts(k,3:4))) & ((sum(prob_fcasts(k,3:4))) <= .1)
        green_1(2) = green_1(2) + 1;
    elseif (.1 < (sum(prob_fcasts(k,3:4))) & ((sum(prob_fcasts(k,3:4))) <= .25)
        green_1(3) = green_1(3) + 1;
    elseif (.25 < (sum(prob_fcasts(k,3:4))) & ((sum(prob_fcasts(k,3:4))) <= .5)
        green_1(4) = green_1(4) + 1;
    elseif (.5 < (sum(prob_fcasts(k,3:4))) & ((sum(prob_fcasts(k,3:4))) <= .75)
        green_1(5) = green_1(5)+ 1;
    elseif (.75 < (sum(prob_fcasts(k,3:4))) & ((sum(prob_fcasts(k,3:4))) <= .9)
        green_1(6) = green_1(6)+ 1;
    elseif (.9 < (sum(prob_fcasts(k,3:4))))
        green_1(7) = green_1(7) + 1;
    end

    if ((sum(prob_fcasts(k,5:6))) == 0)
        blue_1(1) = blue_1(1) + 1;
    elseif (0 < (sum(prob_fcasts(k,5:6))) & ((sum(prob_fcasts(k,5:6))) <= .1)
        blue_1(2) = blue_1(2) + 1;
    elseif (.1 < (sum(prob_fcasts(k,5:6))) & ((sum(prob_fcasts(k,5:6))) <= .25)
        blue_1(3) = blue_1(3) + 1;
    elseif (.25 < (sum(prob_fcasts(k,5:6))) & ((sum(prob_fcasts(k,5:6))) <= .5)
        blue_1(4) = blue_1(4) + 1;
    elseif (.5 < (sum(prob_fcasts(k,5:6))) & ((sum(prob_fcasts(k,5:6))) <= .75)
        blue_1(5) = blue_1(5)+ 1;
    elseif (.75 < (sum(prob_fcasts(k,5:6))) & ((sum(prob_fcasts(k,5:6))) <= .9)
        blue_1(6) = blue_1(6)+ 1;
    elseif (.9 < (sum(prob_fcasts(k,5:6))))
        blue_1(7) = blue_1(7) + 1;
    end
end

if (sum((prob_obs(k,3:4))==1))
    count_obs_2 = count_obs_2 + 1;
    if ((sum(prob_fcasts(k,1:2))) == 0)
        red_2(1) = red_2(1) + 1;
    elseif (0 < (sum(prob_fcasts(k,1:2))) & ((sum(prob_fcasts(k,1:2))) <= .1)
        red_2(2) = red_2(2) + 1;
    elseif (.1 < (sum(prob_fcasts(k,1:2))) & ((sum(prob_fcasts(k,1:2))) <= .25)
        red_2(3) = red_2(3) + 1;
    elseif (.25 < (sum(prob_fcasts(k,1:2))) & ((sum(prob_fcasts(k,1:2))) <= .5)
        red_2(4) = red_2(4) + 1;
    elseif (.5 < (sum(prob_fcasts(k,1:2))) & ((sum(prob_fcasts(k,1:2))) <= .75)
        red_2(5) = red_2(5)+ 1;
    elseif (.75 < (sum(prob_fcasts(k,1:2))) & ((sum(prob_fcasts(k,1:2))) <= .90)
        red_2(6) = red_2(6)+ 1;
    elseif (.9 < (sum(prob_fcasts(k,1:2))))
        red_2(7) = red_2(7) + 1;
    end

    if (sum(prob_fcasts(k,3:4)) == 0)
        green_2(1) = green_2(1) + 1;
    elseif ( 0 < (sum(prob_fcasts(k,3:4))) & ((sum(prob_fcasts(k,3:4))) <= .1)
        green_2(2) = green_2(2) + 1;
    elseif (.1 < (sum(prob_fcasts(k,3:4))) & ((sum(prob_fcasts(k,3:4))) <= .25)
        green_2(3) = green_2(3) + 1;
    elseif (.25 < (sum(prob_fcasts(k,3:4))) & ((sum(prob_fcasts(k,3:4))) <= .5)
        green_2(4) = green_2(4) + 1;
    elseif (.5 < (sum(prob_fcasts(k,3:4))) & ((sum(prob_fcasts(k,3:4))) <= .75)
        green_2(5) = green_2(5)+ 1;
    elseif (.75 < (sum(prob_fcasts(k,3:4))) & ((sum(prob_fcasts(k,3:4))) <= .90)
        green_2(6) = green_2(6)+ 1;
    elseif (.9 < (sum(prob_fcasts(k,3:4))))
        green_2(7) = green_2(7) + 1;
    end

    if ((sum(prob_fcasts(k,5:6))) == 0)
    blue_2(1) = blue_2(1) + 1;
    elseif (0 < (sum(prob_fcasts(k,5:6))) & ((sum(prob_fcasts(k,5:6))) <= .1)
        blue_2(2) = blue_2(2) + 1;
    elseif (.1 < (sum(prob_fcasts(k,5:6))) & ((sum(prob_fcasts(k,5:6))) <= .25)
        blue_2(3) = blue_2(3) + 1;
    elseif (.25 < (sum(prob_fcasts(k,5:6))) & ((sum(prob_fcasts(k,5:6))) <= .5)
        blue_2(4) = blue_2(4) + 1;
    elseif (.5 < (sum(prob_fcasts(k,5:6))) & ((sum(prob_fcasts(k,5:6))) <= .75)
```

```
            blue_2(5) = blue_2(5)+ 1;
        elseif (.75 < (sum(prob_fcasts(k,5:6)))) & ((sum(prob_fcasts(k,5:6))) <= .9)
            blue_2(6) = blue_2(6)+ 1;
        elseif (.9 < (sum(prob_fcasts(k,5:6))))
            blue_2(7) = blue_2(7) + 1;
        end
    end


    if ((sum(prob_obs(k,5:6)))==1)
        count_obs_3 = count_obs_3 + 1;
        if ((sum(prob_fcasts(k,1:2))) == 0)
            red_3(1) = red_3(1) + 1;
        elseif (0 < (sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .1)
            red_3(2) = red_3(2) + 1;
        elseif (.1 < (sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .25)
            red_3(3) = red_3(3) + 1;
        elseif (.25 < (sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .5)
            red_3(4) = red_3(4) + 1;
        elseif (.5 < (sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .75)
            red_3(5) = red_3(5)+ 1;
        elseif (.75 < (sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .9)
            red_3(6) = red_3(6)+ 1;
        elseif (.9 < (sum(prob_fcasts(k,1:2))))
            red_3(7) = red_3(7) + 1;
        end

        if ((prob_fcasts(k,3:4)) == 0)
            green_3(1) = green_3(1) + 1;
        elseif (0 < (sum(prob_fcasts(k,3:4)))) & ((sum(prob_fcasts(k,3:4))) <= .1)
            green_3(2) = green_3(2) + 1;
        elseif (.1 < (sum(prob_fcasts(k,3:4)))) & ((sum(prob_fcasts(k,3:4))) <= .25)
            green_3(3) = green_3(3) + 1;
        elseif (.25 < (sum(prob_fcasts(k,3:4)))) & ((sum(prob_fcasts(k,3:4))) <= .5)
            green_3(4) = green_3(4) + 1;
        elseif (.5 < (sum(prob_fcasts(k,3:4)))) & ((sum(prob_fcasts(k,3:4))) <= .75)
            green_3(5) = green_3(5)+ 1;
        elseif (.75 < (sum(prob_fcasts(k,3:4)))) & ((sum(prob_fcasts(k,3:4))) <= .9)
            green_3(6) = green_3(6)+ 1;
        elseif (.9 < (sum(prob_fcasts(k,3:4))))
            green_3(7) = green_3(7) + 1;
        end

        if ((sum(prob_fcasts(k,5:6))) == 0)
            blue_3(1) = blue_3(1) + 1;
        elseif (0 < (sum(prob_fcasts(k,5:6)))) & ((sum(prob_fcasts(k,5:6))) <= .1)
            blue_3(2) = blue_3(2) + 1;
        elseif (.1 < (sum(prob_fcasts(k,5:6)))) & ((sum(prob_fcasts(k,5:6))) <= .25)
            blue_3(3) = blue_3(3) + 1;
        elseif (.25 < (sum(prob_fcasts(k,5:6)))) & ((sum(prob_fcasts(k,5:6))) <= .5)
            blue_3(4) = blue_3(4) + 1;
        elseif (.5 < (sum(prob_fcasts(k,5:6)))) & ((sum(prob_fcasts(k,5:6))) <= .75)
            blue_3(5) = blue_3(5)+ 1;
        elseif (.75 < (sum(prob_fcasts(k,5:6)))) & ((sum(prob_fcasts(k,5:6))) <= .9)
            blue_3(6) = blue_3(6)+ 1;
        elseif (.9 < (sum(prob_fcasts(k,5:6))))
            blue_3(7) = blue_3(7) + 1;
        end
    end
end

for k = 1:yr
    if k~=ignore
        if ((sum(prob_fcasts(k,1:2))) == 0)
            fc_red(1) = fc_red(1) + 1;
            if ((sum(prob_obs(k,1:2))) == 1)
                        obs_red(1) = obs_red(1) +1;
            end
        end

        if  (0<(sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .10)
            fc_red(2) = fc_red(2) + 1;
            if ((sum(prob_obs(k,1:2))) == 1)
                obs_red(2) = obs_red(2) + 1;
            end
        end

        if  (.1<(sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .25)
            fc_red(3) = fc_red(3) + 1;
```

```
    if ((sum(prob_obs(k,1:2))) == 1)
        obs_red(3) = obs_red(3) + 1;
    end
end

if  (.25 <(sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .50)
    fc_red(4) = fc_red(4) + 1;
    if ((sum(prob_obs(k,1:2))) == 1)
        obs_red(4) = obs_red(4) + 1;
    end
end

if  (.50 <(sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .75)
    fc_red(5) = fc_red(5) + 1;
    if ((sum(prob_obs(k,1:2))) == 1)
        obs_red(5) = obs_red(5) + 1;
    end
end

 if  (.75 <(sum(prob_fcasts(k,1:2)))) & ((sum(prob_fcasts(k,1:2))) <= .90)
    fc_red(6) = fc_red(6) + 1;
    if ((sum(prob_obs(k,1:2))) == 1)
        obs_red(6) = obs_red(6) + 1;
    end
end

if  (.90<(sum(prob_fcasts(k,1:2))))
    fc_red(7) = fc_red(7) + 1;
    if ((sum(prob_obs(k,1:2))) == 1)
        obs_red(7) = obs_red(7) + 1;
    end
end


if (sum(prob_fcasts(k,3:4)) == 0)
    fc_green(1) = fc_green(1) + 1;
    if ((sum(prob_obs(k,3:4))) == 1)
        obs_green(1) = obs_green(1) +1;
    end
end

if  (0<(sum(prob_fcasts(k,3:4)))) & ((sum(prob_fcasts(k,3:4))) <= .10)
    fc_green(2) = fc_green(2) + 1;
    if ((sum(prob_obs(k,3:4))) == 1)
        obs_green(2) = obs_green(2) + 1;
    end
end

if  (.1<(sum(prob_fcasts(k,3:4)))) & ((sum(prob_fcasts(k,3:4))) <= .25)
    fc_green(3) = fc_green(3) + 1;
    if ((sum(prob_obs(k,3:4))) == 1)
        obs_green(3) = obs_green(3) + 1;
    end
end

if  (.25<(sum(prob_fcasts(k,3:4)))) & ((sum(prob_fcasts(k,3:4))) <= .50)
    fc_green(4) = fc_green(4) + 1;
    if ((sum(prob_obs(k,3:4))) == 1)
        obs_green(4) = obs_green(4) + 1;
    end
end

if  (.50<(sum(prob_fcasts(k,3:4)))) & ((sum(prob_fcasts(k,3:4))) <= .75)
    fc_green(5) = fc_green(5) + 1;
    if ((sum(prob_obs(k,3:4))) == 1)
        obs_green(5) = obs_green(5) + 1;
    end
end

if  (.75 <(sum(prob_fcasts(k,3:4)))) & ((sum(prob_fcasts(k,3:4))) <= .90)
    fc_green(6) = fc_green(6) + 1;
    if ((sum(prob_obs(k,3:4))) == 1)
        obs_green(6) = obs_green(6) + 1;
    end
end

if  (.90<(sum(prob_fcasts(k,3:4))))
    fc_green(7) = fc_green(7) + 1;
    if ((sum(prob_obs(k,3:4))) == 1)
        obs_green(7) = obs_green(7) + 1;
    end
```

```
        end


        if ((sum(prob_fcasts(k,5:6))) == 0)
            fc_blue(1) = fc_blue(1) + 1;
            if ((sum(prob_obs(k,5:6))) == 1)
                obs_blue(1) = obs_blue(1) +1;
            end
        end

        if  (0<(sum(prob_fcasts(k,5:6)))) & ((sum(prob_fcasts(k,5:6))) <= .10)
            fc_blue(2) = fc_blue(2) + 1;
            if ((sum(prob_obs(k,5:6))) == 1)
                obs_blue(2) = obs_blue(2) + 1;
            end
        end

        if  (.1<(sum(prob_fcasts(k,5:6)))) & ((sum(prob_fcasts(k,5:6))) <= .25)
            fc_blue(3) = fc_blue(3) + 1;
            if ((sum(prob_obs(k,5:6))) == 1)
                obs_blue(3) = obs_blue(3) + 1;
            end
        end

        if  (.25 <(sum(prob_fcasts(k,5:6)))) & ((sum(prob_fcasts(k,5:6))) <= .50)
            fc_blue(4) = fc_blue(4) + 1;
            if ((sum(prob_obs(k,5:6))) == 1)
                obs_blue(4) = obs_blue(4) + 1;
            end
        end

        if  (.50 <(sum(prob_fcasts(k,5:6)))) & ((sum(prob_fcasts(k,5:6))) <= .75)
            fc_blue(5) = fc_blue(5) + 1;
            if ((sum(prob_obs(k,5:6))) == 1)
                obs_blue(5) = obs_blue(5) + 1;
            end
        end

        if  (.75 <(sum(prob_fcasts(k,5:6)))) & ((sum(prob_fcasts(k,5:6))) <= .90)
            fc_blue(6) = fc_blue(6) + 1;
            if ((sum(prob_obs(k,5:6))) == 1)
                obs_blue(6) = obs_blue(6) + 1;
            end
        end

        if  (.90<(sum(prob_fcasts(k,5:6))))
            fc_blue(7) = fc_blue(7) + 1;
            if ((sum(prob_obs(k,5:6))) == 1)
                obs_blue(7) = obs_blue(7) + 1;
            end
        end

    end
end

dis_red_1 = (red_1/count_obs_1);
dis_red_2 = (red_2/count_obs_2);
dis_red_3 = (red_3/count_obs_3);
dis_green_1 = (green_1/count_obs_1);
dis_green_2 = (green_2/count_obs_2);
dis_green_3 = (green_3/count_obs_3);
dis_blue_1 = (blue_1/count_obs_1);
dis_blue_2 = (blue_2/count_obs_2);
dis_blue_3 = (blue_3/count_obs_3);

rel_red  = (obs_red./fc_red);
rel_green = (obs_green./fc_green);
rel_blue = (obs_blue./fc_blue);

fprintf(fid2,' %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f \n' , rel_red' );
fprintf(fid2,' %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f \n' , rel_green' );
fprintf(fid2,' %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f \n' , rel_blue' );
fprintf(fid1,' %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f \n' , dis_red_1' );
fprintf(fid1,' %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f \n' , dis_green_1' );
fprintf(fid1,' %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f \n' , dis_blue_1' );
fprintf(fid1,' %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f \n' , dis_red_2' );
fprintf(fid1,' %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f \n' , dis_green_2' );
fprintf(fid1,' %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f \n' , dis_blue_2' );
fprintf(fid1,' %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f \n' , dis_red_3' );
fprintf(fid1,' %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f \n' , dis_green_3' );
fprintf(fid1,' %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f %1.4f \n' , dis_blue_3' );
```

```
fclose(fid1);
fclose(fid2);
fprintf(fid3,'%3.0f %3.0f %3.0f \n',count_obs_1,count_obs_2,count_obs_3);
fprintf(fid4,'%3.0f %3.0f %3.0f %3.0f %3.0f %3.0f %3.0f \n',fc_red');
fprintf(fid4,'%3.0f %3.0f %3.0f %3.0f %3.0f %3.0f %3.0f \n',fc_green');
fprintf(fid4,'%3.0f %3.0f %3.0f %3.0f %3.0f %3.0f %3.0f \n',fc_blue');
fclose(fid3);
```

Appendix III

Average RPS statistics for forecast points not included in this study.

Forecast Point	RPS: weekly stage forecasts)	RPS: monthly stage forecasts
BELW2	1.21	1.04
BRAW2	1.24	1.14
CARI2	0.46	0.84
CLAT1	0.96	1.87
CLKW2	1.25	0.97
CLLP1	0.90	0.54
COKP1	1.07	0.47
ECMP1	0.72	0.47
ELRP1	0.85	0.78
FDLP1	0.73	0.47
FOMK2	2.16	1.14
FRKP1	0.88	0.76
HAIV2	0.69	1.22
LEAO1	1.24	1.23
MEDP1	1.20	0.87
NCSP1	0.80	0.84
OLNN6	1.33	1.21
PARP1	0.94	0.87
PHIW2	1.22	1.41
PINW2	1.51	0.70
PSNW2	0.99	1.61