Reply to Referee #2

Dear Reviewer 2,

We would like to thank you for your in-depth review and interesting thoughts. We believe that your comments and suggestions will help us to improve our manuscript. Please find below a step-by-step reply to your comments and suggestions.

Yours faithfully,
The authors

L 21 – What do the the authors call "anomalous radionuclide detections"? That is something I am perfectly aware of, but it is perhaps not the case of all readers.

Reply:

Thank you for this suggestion. We will add the following definition to the manuscript:

“Anomalous radionuclide detections are detections of anthropogenic radionuclides originating from upwind nuclear facilities, where the detected concentration of (a) specific radionuclide(s) and/or the combination of several detected radionuclides are anomalous with respect to the station’s detection history and/or with respect to what can be expected from these upwind nuclear facilities operating under normal conditions.”

L 25 – According to the authors, atmospheric transport and dispersion modelling is "one of the methods" to relate detections and the source of emission. I do not see other methods. Which other methods do the authors have in mind?

Reply:

In theory, ratios of specific radionuclides (if these are all detected in a certain sample, and assuming no contamination from other sources) could help to discriminate between different sources, without using an atmospheric transport model.

L 30 – In backward modelling, the source-receptor relationships are calculated from fixed receptors to potential sources (not the opposite as written in the sentence in L 30).

Reply:

We will correct this ambiguity in the revised manuscript.

L 32 – The concept of "non-detection" should be explained (or ignored as it is not used in the paper).

Reply:

We will add to the revised manuscript (in green):

“Statistical methods can then be employed to combine the information from all these detections (and possibly non-detections - observations where the activity concentration is below a minimum detectable concentration) in a meaningful way in order to infer relevant information on the source.”

L 44 – In this paper, the model error is considered as a whole. Thus, it does not originate only from the numerical weather predictions, but also from the atmospheric transport and dispersion model. The word "mainly" ("because of the underlying weather prediction data") is questionable. The authors should consider rephrasing the sentence.

Reply:

In our experience, the NWP data results in the largest uncertainty in atmospheric transport modelling using the Flexpart model. In Flexpart, the NWP data determines the transport (by the wind) and dispersion (through parameterisation using atmospheric stability) of particles. Source uncertainties are not applicable here, since we work backward in time. In our experience, Flexpart is fairly robust against perturbations of the Flexpart model parameters.

There is also literature that supports our claim in L 44. We will add these references in the revised manuscript:

Engström, A., & Magnusson, L. (2009). Estimating trajectory uncertainties due to flow dependent errors in the atmospheric analysis. *Atmospheric Chemistry & Physics*, *9*(22).

Harris, J. M., Draxler, R. R., & Oltmans, S. J. (2005). Trajectory model sensitivity to differences in input data and vertical transport method. *Journal of Geophysical Research: Atmospheres*, *110*(D14).

Hegarty, J., Draxler, R. R., Stein, A. F., Brioude, J., Mountain, M., Eluszkiewicz, J., ... & Andrews, A. (2013). Evaluation of Lagrangian particle dispersion models with measurements from controlled tracer releases. *Journal of Applied Meteorology and Climatology*, *52*(12), 2623-2637.

However, we welcome findings or literature from the Reviewer that would contradict or complement the above and remain open to adapt that part of the manuscript accordingly.

L 54 – As for me, it is difficult to create and use a relevant ensemble. The reason is not only (and perhaps not mainly) the computational cost of the ensemble, but the way to constitute it with enough variety, limited redundancy, etc. This complex task should be mentioned in the paper.

Reply:

We agree with that and propose to add the following to the revised manuscript:

“Creating an ensemble with a meaningful spread between its different members (that is, spread which represents the model uncertainty) is a very complex task which requires expert knowledge of all data, processes and their associated uncertainties at each level of the modeling process.”

L 57 – Ditto. It is complicated and not guaranteed that an ensemble captures "most of the possible outcomes". This should be indicated in the paper.

Reply:

We propose the following rewording in the revised manuscript:

“Therefore, ensembles used operationally at major weather institutes around the world are designed in a way that, even with a limited number of members (between 14 and 50, Leutbecher, 2019), the ensemble ~~can capture most~~ tries to capture all (and not more) of the possible outcomes.”

L 59 – What is a "measurement model"?

Reply:

We will refer to Eq. 19 and Eq. 20 in the revised manuscript, and will write that “a measurement model relates the model variable with the observation.”

L 88 – The description of the detections should be gathered in a table with the collection start and stop times (even if I guess that the authors do not wish to develop this aspect of the data).

Reply:

You are correct and this is a helpful suggestion. The text from the paragraph has been reworked to present the data in tabular format.

L 96 – The beginning of the sentence is "the above observation times". I do not see any observation times above?

Reply:

In that paragraph, we mentioned when the observations were made (L 89 – 92 in the original manuscript). However, we will clarify this in the revised manuscript.

L 101 – It is written that FLEXPART is run in backward mode. I wonder how long the simulations go back in time. Could the authors give information about this?

Reply:

We will add to the revised manuscript:
“All simulations ended on 20 September 2017.”

L 110 – It is not obvious that adding and substracting perturbations from an ensemble mean are a legitimate process. Could the authors comment on this?

Reply:

This was motivated by the idea that the unperturbed member could perform slightly better than the perturbed members, so that better results could be obtained by centering the perturbations around the unperturbed member rather than around the ensemble mean.

L 113 – The authors assert that "the spread between the different members represent *the uncertainty". This is undoubtedly a way to account for uncertainty in weather predictions, but are the authors sure that the ensemble perfectly encompasses the uncertainty on the meteorological data? The authors should consider being more cautious and rephrasing this sentence.*

Reply:

With that, we rather meant the general principle of an ensemble: viz. the spread between the members represents the uncertainty. Of course, a bad ensemble will result in a bad uncertainty estimate. We propose to rewrite it as follows:

“The perturbations are created in such a way that each ensemble member represents a possible scenario for the true (unknown) atmospheric state, and the spread between the different members ~~represents the uncertainty~~ is simply the model uncertainty as estimated by the ensemble.”

L 130 – What are the values of t1 and tm, the first and last time for which source-receptor-sensitivities are available for the source reconstruction?

Reply:

This is discussed in Subsection 3.2 “prior distribution”: t_1 is 25 September 2017 0000 UTC and t_m is 28 September 2017 0000 UTC. (Flexpart output files were available for other times too.)

L 131 – The authors assume that the release rate is constant during the release period. I would like to point out that this is a strong assumption as in principle, the release is not known at all. Could the authors comment on this?

Reply:

We repeat our answer to Reviewer 1, who made a similar comment:
“It is important to distinguish between different geotemporal scales. While time-varying emissions can have a huge impact nearby the source, these effects are less significant further away from the source due to the atmospheric transport and dispersion processes (and the atmospheric transport model, which filters such information out). Hence, we expect a constant release within release parameters t_start and t_stop to be appropriate to describe the Ru-106 source.”

See also:

De Meutter, P., Camps, J., Delcloo, A., Deconninck, B., & Termonia, P. (2018). Time resolution requirements for civilian radioxenon emission data for the CTBT verification regime. *Journal of environmental radioactivity*, *182*, 117-127.

*L 138 – The total release is assumed to be between 10**10 and 10**16 Bq. This seems to me somewhat arbitrary as it excludes potential releases respectively further downwind and further upwind. Once more, how to proceed when no preconceived solution is available? Could the authors consider commenting on this point?*

Reply:

From the available number of measurements, and the scale at which detections were made, these bounds are not unrealistic. Smaller sources would not have been seen over such a broad geographic area, while larger sources would have been seen at more monitoring locations. The selected bounds represent a conservative, but realistic bound for the source. Furthermore, we have already applied inverse modelling using a cost function approach for this case, which allowed us to make our prior distributions sharper than what can be done without knowledge on this case; please see:

De Meutter, P., Camps, J., Delcloo, A., and Termonia, P.: Source Localization of Ruthenium-106 Detections in Autumn 2017 Using Inverse Modelling, in: Mensink C., Gong W., Hakami A. (eds) Air Pollution Modeling and its Application XXVI. ITM 2018. Springer Proceedings in Complexity., Springer, Cham, https://doi.org/10.1007/978-3-030-22055-6_15, 2020.

L 141 – Ditto. How did the authors choose the time interval of the release (all the more that this time interval is quite short)?

Reply:

(Please also see our reply to your previous comment.) From earlier studies, we knew that the bulk release of Ru-106 likely took place between that period. Since a detailed analysis of the Ru-106 case was not our intention, we have chosen to focus on this time period. An additional benefit of reducing the allowed time interval of the release (when fixing the spatial domain) is that it reduces the memory requirements, which is beneficial when running the case on a personal computer. (Note, however, that the tool can also be run on a server or cluster where more memory is available.)

L 148 – This is another strong hypothesis that the observations are independent while there is likely a space and time dependency between them. Could the authors comment on this?

Reply:

We acknowledged in the manuscript that this is a simplification. Given the large distance (~ 1000 km) between different IMS stations, we believe this approximation is not too incorrect. Furthermore, the authors are not aware of similar studies that take into account geotemporal dependencies between observations, and we would be grateful if the Reviewer could provide some references.

L 160 – Does the index “i” in formula (5) indicate that there are as many applications of this formula (with possibly different values of the s, alpha bar and beta bar parameters) as the number of observations?

Reply:

Eq. 5 is indeed for a single observation. The values for s, alpha bar and beta bar can be made observation-specific (which is also done further in the paper).

L 189 – I wonder if the general-purpose Markov Chain Monte Carlo algorithm MT-DREAM(ZS) is freely available? Who developed this MCMC method?

Reply:

It was developed by Laloy and Vrugt and described in their paper Laloy and Vrugt (2012). Some implementations of DREAM can be found in open source packages on the internet.

Figure 2 – I suppose that "MDC" stands for "Minimum Detectable Concentration" and that we have LC # MDC / 2. In the formulae, it seems that only LC is used. Could the authors confirm this point?

Reply:

In the formulae, L_C is used. With the observations, typically the MDC is reported and not L_C. For the observations in the IMS network of CTBTO, we can assume that L_C = MDC/2.

L 192 – While popular, MCMC methods have well-known drawbacks like the burn-in period or convergence problems. Could the authors consider commenting on this with respect to the MT-DREAM(ZS) algorithm?

Reply:

This depends on the case, but from the authors' experience over the past year, we typically run the tool using ~ 10,000 iterations and convergence occurs after ~ 2,500 iterations (where we discard these first 2,500 iterations). In our previous study however, (De Meutter and Hoffman, 2020) where we studied the Se-75 release, we used 150,000 iterations.
The required number of iterations is also affected by the choice of the uncertainty "s": lower uncertainties require more iterations before convergence takes place.

L 197 – I have the feeling that all technical details in the last part of this paragraph (and notably the "snooker step") would need some more explanations as this part of the text is too concise (and a bit obscure).

Reply:

Regarding the snooker step, we were informed by one of the developers of MT-DREAM(ZS) that the snooker step is theoretically not compatible with the multiple-try part of the algorithm, so that we no longer use the snooker step. The difference in the posterior after using and not using the snooker step is not noticeable in our simulations.

To prove the latter, please find the results below for two simulations for the Ru-106 case, with and without the snooker step:

1/ simulation with the snooker step for the unperturbed member and s_i = 0.5

```
 Running MT-DREAMzs, iteration 7800 of 50001 . Current logp -37.44259 -41.24711 -39.54531
Converged after 7800 iterations
 Running MT-DREAMzs, iteration 50001 of 50001 . Current logp -36.48064 -44.17845 -41.44014
 MT-DREAMzs terminated after 1206.558 seconds
Acceptance rate for chain 1 is 22.24%
Acceptance rate for chain 2 is 22.61%
Acceptance rate for chain 3 is 22.93%
           lon      lat  log10_Q             rstart               rstop
0.025 50.11799 55.50922 14.96976 2017-09-25 00:22:32 2017-09-26 23:34:40
0.5   51.09007 55.91466 15.27527 2017-09-25 07:59:14 2017-09-27 18:17:46
0.975 57.88037 60.75305 15.64360 2017-09-25 22:55:13 2017-09-27 23:34:05
mean  51.98106 56.39979 15.28396 2017-09-25 08:46:06 2017-09-27 16:41:15
```

2/ simulation without the snooker step for the unperturbed member and s_i = 0.5

```
 Running MT-DREAMzs, iteration 12300 of 50001 . Current logp -44.62895 -44.45257 -41.44722
Converged after 12300 iterations
 Running MT-DREAMzs, iteration 50001 of 50001 . Current logp -43.13082 -39.71556 -37.64025
 MT-DREAMzs terminated after 1271.046 seconds
Acceptance rate for chain 1 is 25.26%
Acceptance rate for chain 2 is 24.75%
Acceptance rate for chain 3 is 23.59%
           lon      lat  log10_Q              rstart               rstop
0.025 50.09579 55.51231 14.97063 2017-09-25 00:30:47 2017-09-26 23:01:53
0.5   51.03933 55.88155 15.26998 2017-09-25 08:00:29 2017-09-27 18:35:54
0.975 57.74679 60.13044 15.65440 2017-09-25 22:38:03 2017-09-27 23:37:18
mean  51.85432 56.30864 15.27988 2017-09-25 08:42:32 2017-09-27 16:59:44
```

The following information will be added to the revised manuscript:

~~“The algorithm is designed so that a snooker step occurs with a probability of 20 % to allow jumps between different posterior modes (ter Braak and Vrugt, 2008).~~ To enhance efficiency and to obtain more accurate results, randomized subspace sampling is used (Vrugt et al., 2009). This simply means that not necessarily all source parameters are updated at a time, but instead a randomized subset of the source parameters. Furthermore, MT-DREAM (ZS) makes use of multiple try Metropolis sampling (Liu et al., 2000) to enhance the mixing of the chains. This means in practice that, to advance to Markov chain, several proposals are drawn instead of one proposal in traditional Metropolis sampling. Furthermore, the Metropolis acceptance is calculated in a different way (Liu et al., 2000 , Laloy and Vrugt, 2012).”

L 209 – It is written here that “s” is an estimate of “sigma”, but “sigma” is not defined, nor introduced before. Should the reader understand that sigma stands for sigma_mod?

Reply:

Thank you for pointing this out. In L 209, “sigma” should have been “sigma_mod”. Note however that in L 222, “sigma” stands for sqrt(sigma_mod^2 + sigma_obs^2). We will add that to the revised manuscript.

L 215 – In formula (17), “sigma_srs” and “srs” are not defined. What do these notations stand for? Moreover, what is the reason for the multiplicative value of 16 (and not another value) in the same formula? Could the authors comment on this?

Reply:

Thank you for pointing that out. “srs” stands for source-receptor-sensitivities (the model output when Flexpart is run in backward mode), and “sigma_srs” is its (unknown) uncertainty. We will add that to the revised manuscript.

The value of 16 is an empirical number that was found to give a good balance between information obtained from detections versus information from non-detections from an earlier case study described in De Meutter and Hoffman (2020). We will add this information in the revised manuscript.

L 216 – The sentence: “as a consequence, the model uncertainty does not depend on the source parameters” is especially unclear or unprecise. What do the authors call “the model”? Is it the weather prediction or the transport and dispersion simulation or *both? As the source parameters are not considered as uncertain, I do not see why and how they should take part in the model uncertainty. Please, consider rephrase this sentence.*

Reply:

We can calculate the modeled activity concentrations c_mod as a linear relation between the source-receptor-sensitivities (srs) and the release amount Q:

c_mod = srs(release period, release location) * Q

One way of calculating the uncertainty on c_mod (sigma_c_mod) would then be to use sigma_srs, which could be obtained from the ensemble:

sigma_c_mod = sigma_srs(release period, release location) * Q

However, in that case, sigma_c_mod will depend on the source parameters (the release period, the release location and Q). This resulted in undesired effects in the very beginning of the development of the FREARtool (such as: the model selecting very high Q so that the uncertainty became very large, thereby allowing values of c_mod that did not agree at all with c_obs), so that it was decided to make the model uncertainty independent of the source parameters.

To avoid confusion, we propose to omit this sentence.

L 218 – I wonder how "a part of the plume" can be "subject to more atmospheric transport and dispersion processes". All parts of the plume are subject to atmospheric transport and dispersion processes. Small detections may be obtained at the "edge" of the plume or just far from the source of the release. What does a "small" detection mean? It is just a matter of detection method and device. While I globally agree with the ideas contained in this paragraph, I feel that they should be formulated in a different way.

Reply:

We propose the following revision:
L 218: "This is desirable since small detections are caused by a part of the plume of radionuclides that was subject to more atmospheric ~~transport and dispersion processes~~ dilution..."

L 226 – The whole section 4 uses the ECMWF unperturbed weather prediction. This should be mentioned at the beginning of the section.

Reply:

Indeed, thank you for pointing this out.

L 229 – As I understand "s_i" includes the model error and the observation error. I wonder what the respective parts of each kind of errors are. Could the authors comment on this? The authors present the source location probability map for three values of "s_i". Of course, it is difficult to choose this parameter and it is the central question which the paper deals with. Is it possible for the authors to motivate the choice of the three "s_i" values? Finally, it is written that "the same value s_i is used for all observations". I wonder why different values of s_i should be associated to the observations as the *observation error is by assumption the same for each observation and the model error should depend intrinsically on the model and not on the observation.*

Reply:

The interpretation of the different s_i values is straightforward from Eq. 17: it represents a relative error of 30 %, 50 % and 300 % with respect to max(c_det, 16 * L_C). 50 % was our initial “default value”. 300 % seemed a good value to go above that (we also tested other values, such as 100 % and 1 000 %). The choice for the lower value is limited by the observation error (lower values for sigma_total would imply an imaginary model error). Furthermore, some members had troubles with convergence when very small s_i values were chosen (10 %).

The observation error is different for each observation as it depends on the background radiation, the sampled volume of air etc… We believe that the model error should also be observation-specific, please see our reply further below to a comment regarding L 347.

Figure 3 – The figure 3 as the following figures seem to me a bit small.

Reply:

We will increase the figure size in the revised manuscript.

L 237 – I do not see what is an “unknown error”? There are observation errors, representativeness errors or model errors including among others the atmospheric processes not resolved by the model. What is “unknown” is not the type of error, but the value to be attributed to the error.

Reply:

In the revised manuscript, we will make the following change:

“~~Besides being an alternative model error, multipliers could also be used to take into account unknown errors (such as errors due to local atmospheric features not resolved by the model).~~”

“Besides being an alternative model error, multipliers could also be used to take into account errors that were not fully captured by the model (such as errors due to local atmospheric features not resolved by the model, measurement errors due to sample inhomogeneity, etc.)”

L 270 – Increasing the value of the parameter s_i results in a shift and an enlargement of the posterior distribution. I wonder why introducing multiplier only results in a shift of the posterior. I suppose that it acts as another way to adjust the posterior without any increase in the level of model uncertainty. Could the authors comment on this?

Reply:

That sounds certainly plausible. The model uncertainty is indeed not affected by the multipliers. The multipliers allow a better match between “m * c_mod” and the observations “c_obs”. This better agreement can in theory be obtained with the same source parameters when no multipliers are used (thus, no shift will be seen), or it can be obtained with different source parameters (so that a shift will be seen if the source location is affected).

L 272 – I presume that forcing the model uncertainty with a high value of the parameter s_i predominates against the influence of the multipliers. Do the authors have the same explanation?

Reply:

If the model uncertainties (determined by s_i) are larger than "|c_mod – c_obs|", then indeed the multipliers will have less impact on the posterior.

L 281 – As for me, it is not so obvious that the errors arising from the meteorological input data have the "largest contribution" to the total model error. Would the atmospheric transport and dispersion model be a "bad model" (what is probably not the case of FLEXPART), the dispersion model error would not be negligible. The authors should perhaps moderate their assessment in L 281.

Reply:

Please see also our reply to a related comment concerning L 44. We propose the following (minor) moderation but remain open to consider further moderation if the Reviewer could share findings or literature that shows its necessity.

"While this type of error ~~arguably~~ likely adds the largest contribution to the total model error, other sources of model error are not included."

L 285 – How the data of all grid boxes is aggregated should be more explained. For me, it is not an obvious process.

Reply:

We will add the following in the revised manuscript (in green):

"In order to obtain the error structure, the data of all spatial grid boxes is aggregated into an uncertainty distribution."

Furthermore, we will add to the list:

> "4. The remaining data points are used to make an uncertainty distribution (as in Figure 4)."

L 298 – The probability density function of the SRS members should be presented not only for "an arbitrary observation and an arbitrary time" as in Figure 4, but for other observations and times or all distributions should be considered and their moments computed.

Reply:

It is not feasible to plot the distribution for each time and each observation (288 in total) in one figure, but we will add this as supplementary information.

Figure 4 – There is a typo in the caption: "distributed" versus "distribution".

Reply:

Thank you for noticing this. We will correct this in the revised manuscript.

L 321 – I wonder about the generality of the method presented by the authors, especially in case 4 when the parameters are fitted for each observation and time. As a matter of fact, it means that just adding or removing a detection will not only influence the source term estimate, but also the uncertainty on this estimate (and this with the same meteorological fields). Could the authors comment on this?

Reply:

It is not only the meteorological fields that determine the uncertainty, but also the trajectories that particles follow along these meteorological fields. As a result, the model uncertainty is observation-specific, and indeed, adding or removing observations can alter the uncertainty on the inferred parameters. See also our next reply.

L 347 – Considering "observation-specific" uncertainty parameters is an ad hoc (and interesting) way to fit the model (and observation) error, but it should not be forgotten that the model error should be an intrinsic feature of the model and not depend on the set of observations which is taken into account. I suggest that the authors argue on this.

Reply:

We do not agree that the model error should be an intrinsic feature of the model and does not depend on the observation: the model uncertainty depends on the trajectory of the retro-plume (= the plume that goes from the sampling station backward in time). Observations of a plume that are made three weeks after the release should have higher model uncertainty than observations made two days after the release. Also, depending on the weather conditions along the trajectory, the model error can be observation specific (consider transport associated with a frontal system versus transport associated with the calm conditions found in an anticyclone).

To clarify this, we will add (see text in green) to the revised manuscript:
L341: "In this subsection, it is assessed how the fitted uncertainty parameters vary among different observations and different times. The motivation for this is as follows: first, and somewhat trivial, we can expect the model uncertainty to increase as a function of simulation time. Second, uncertainties are expected to be observation-dependent, since observations are made on different times and at different distances from the source; uncertainties on the trajectories between the receptor and the source will also be affected by the atmospheric conditions along the trajectory, which are expected to be observation-specific."

L 350 – That the model uncertainty grows when going backwards in time is somewhat trivial. At least, the contrary would be surprising.

Reply:

We agree, but it is always good to confirm that our ensemble of atmospheric transport simulations replicates evident features.

L 353 – It is worth noticing that the oscillations have a circadian period. Is it possible to relate them with the day and night alternation of the boundary layer?

Reply:

We believe that L 351 made that notice:
"Also interesting to note is that there is an oscillatory behaviour with a period of eight time steps, corresponding to the diurnal cycle (since SRS fields were produced every three hours). The oscillations are likely associated with boundary layer processes, which often follow the diurnal cycle."

L 365 – It is quite optimistic to assert that both maps in Figure 7 roughly agree. There are many differences. Would the location of the release be the aim of the study, the authors would be certainly quite embarrassed to designate it using one map or the other.

Reply:

Indeed, but we assume the output of the inference will be interpreted by an expert, who is aware that models have uncertainties, and that even the uncertainties are uncertain. Also take into account that we zoom in into the area of interest. If we would plot the full domain, the differences will appear smaller.

L 390 – I would like to point out that there is an interesting result in L 390. As a matter of fact, using the ensemble only to fit the uncertainty parameters or running all members of the ensemble to figure out the uncertainty seems to be equivalent.

Reply:

We believe L 389 in the original manuscript mentions this, but we will try to make it more explicit by adding (in green):

"It seems that overall, a similar picture is obtained when running the Bayesian inference for each ensemble member separately, compared to the procedure explained in Section 5. This suggests that if we use the ensemble only (i) to fit the uncertainty parameters and (ii) to calculate the ensemble median SRS for running the inference as was done in order to obtain Fig. 7, no crucial information from the ensemble is lost with respect to the source location. As a consequence, it is equivalent to running the inference with all members of the ensemble separately to determine the uncertainty."

L 410 – As a conclusion, I would suggest to the authors to apply the different approaches and methods presented in their paper to situations in which the source characteristics (especially the location) is known unambiguously (because in the Ru-106 case the source location was not really recognized). In a situation with a clearly identified location of the emission, it would be interesting to see what results (good or less good) are obtained using the inference in different ways, and also what is the most efficient approach.

Reply:

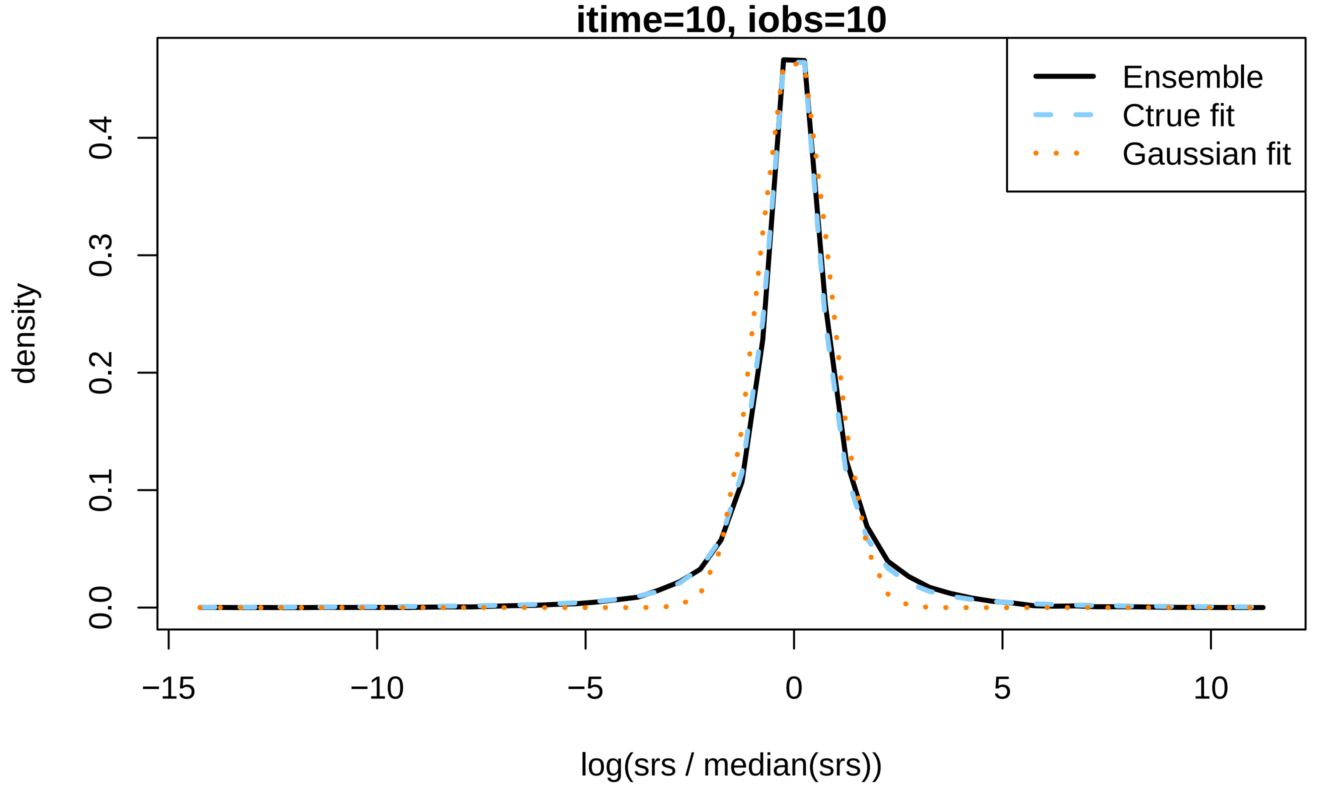
Thank you for this suggestion, which is in line with the comments made by Reviewer 1. We will add to the conclusions:
“In a future study, we will apply the different approaches and methods presented in this paper to situations in which the source characteristics are known unambiguously. This will help to better evaluate the different approaches proposed in this paper.”

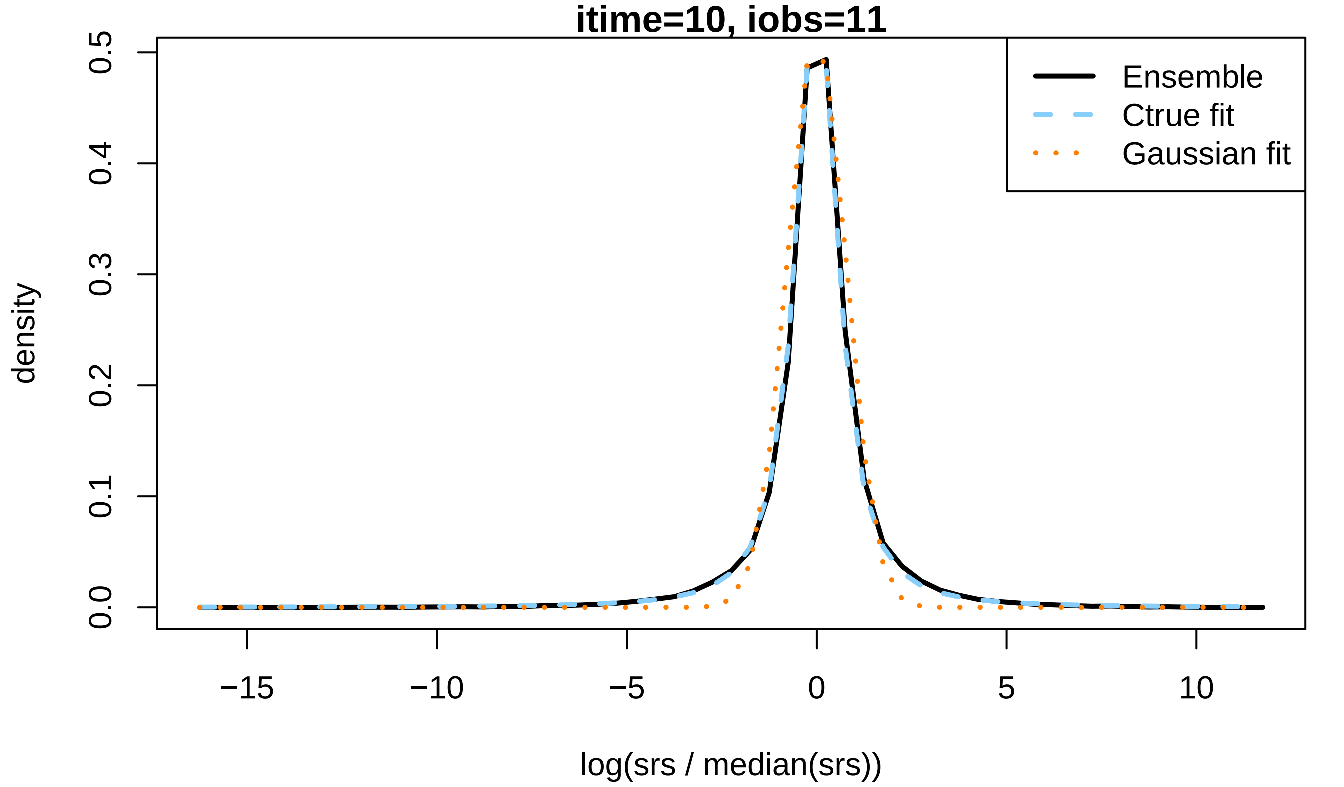
L 435 – As argued by the authors, it seems that using the members of an ensemble in the source term estimate gives more robust results with regard to the choice of the uncertainty parameter as opposed to not using any ensemble. It seems to me quite logical as the ensemble introduces a kind of uncertainty (which is certainly not all the uncertainty, but a “rigorously built” uncertainty). This uncertainty may predominate against the uncertainty arbitrarily fixed by choosing the uncertainty parameter.

Reply:

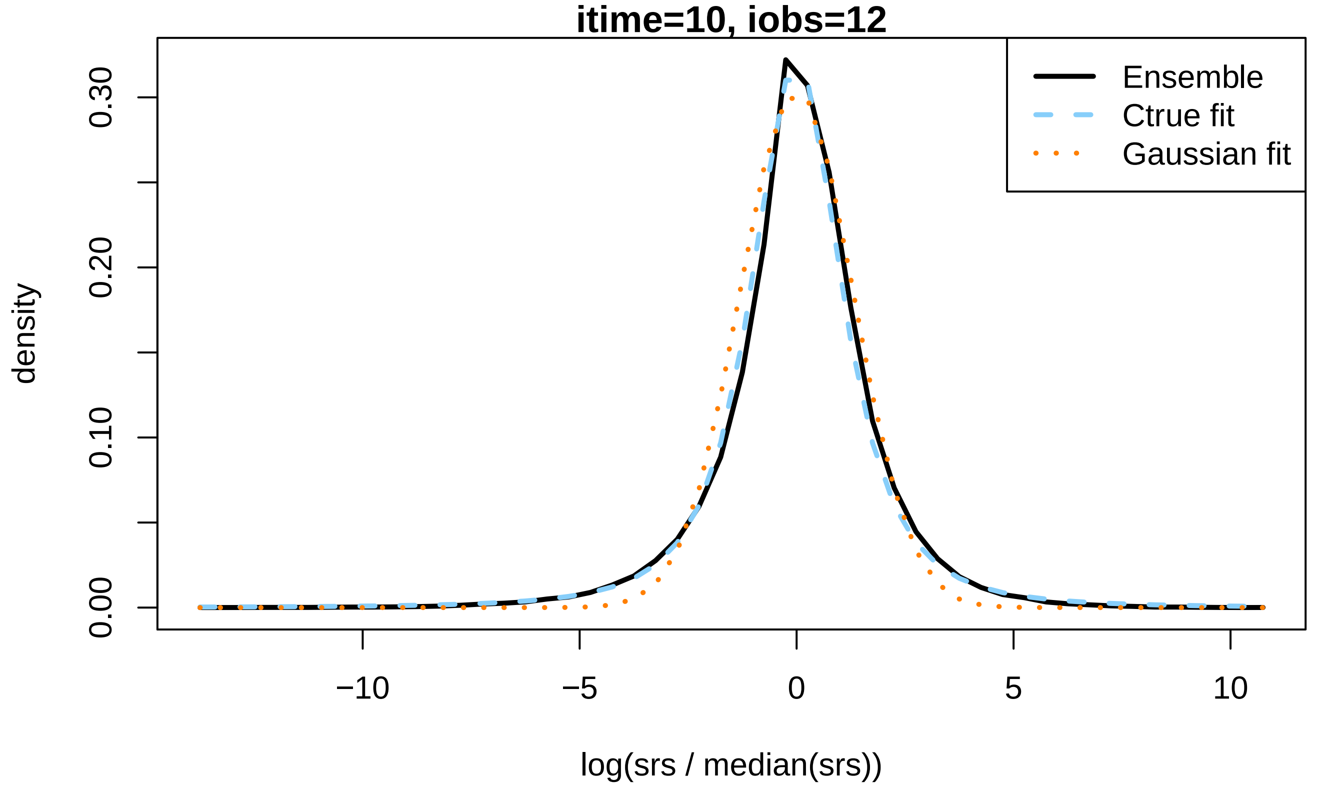
We agree with that, and propose to add that in the revised manuscript (in green):

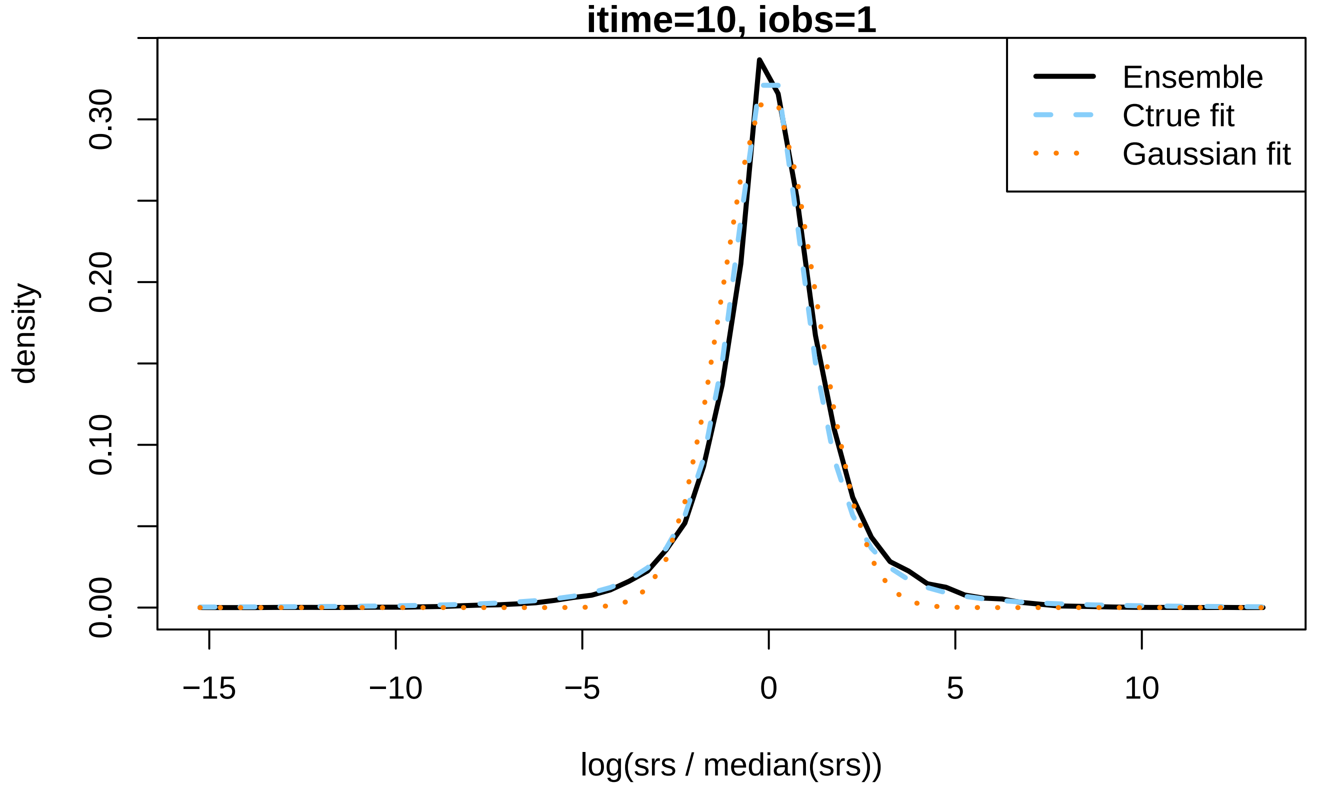
“A scenario-based approach (where each ensemble member is used as input for the Bayesian source reconstruction, instead of using the ensemble to fit the uncertainty parameters) gives results which are more robust against the choice of the uncertainty parameters but is more costly compared to directly fitting the uncertainty parameters. This is because the ensemble introduces model uncertainty that may predominate against the uncertainty prescribed by arbitrarily choosing the uncertainty parameter.”

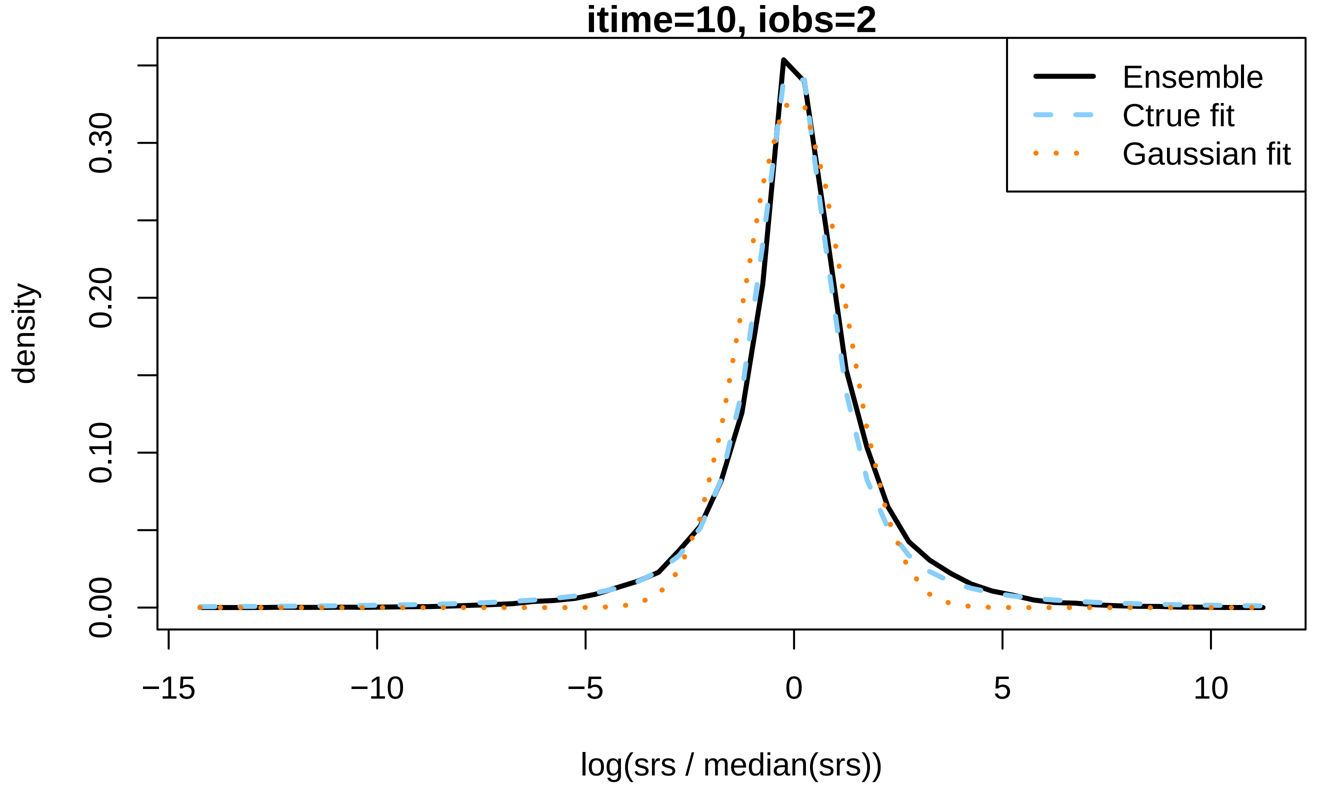

itime=10, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))

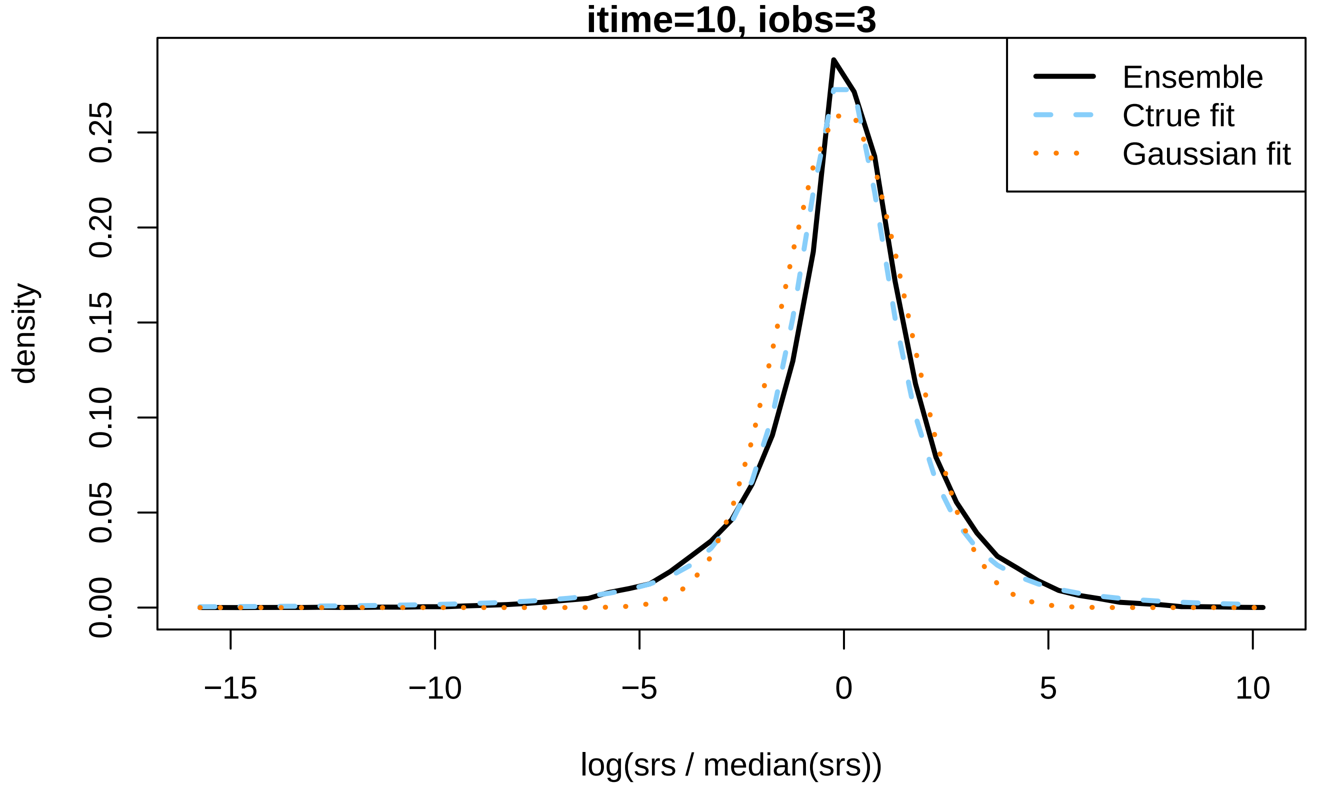


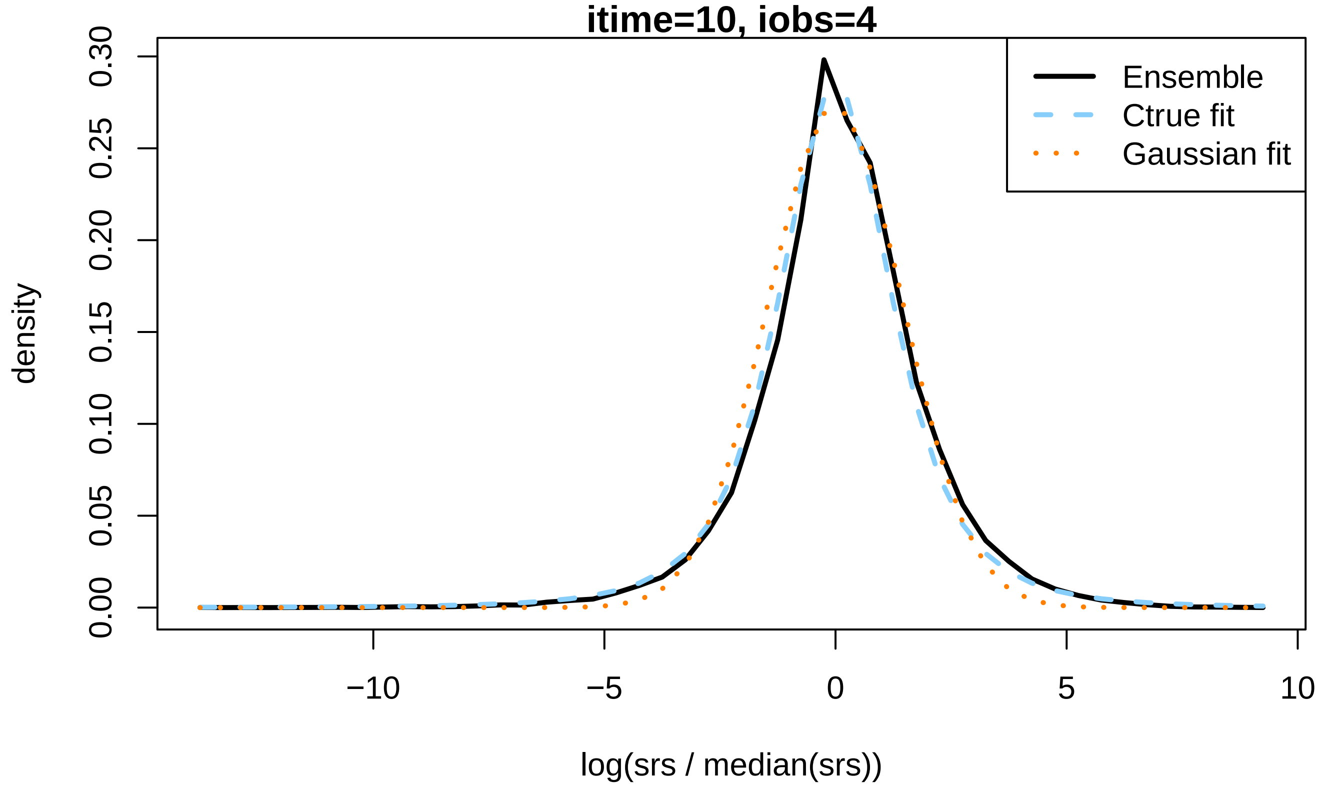
itime=10, iobs=11
density
0.0
0.1
0.2
0.3
0.4
0.5
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

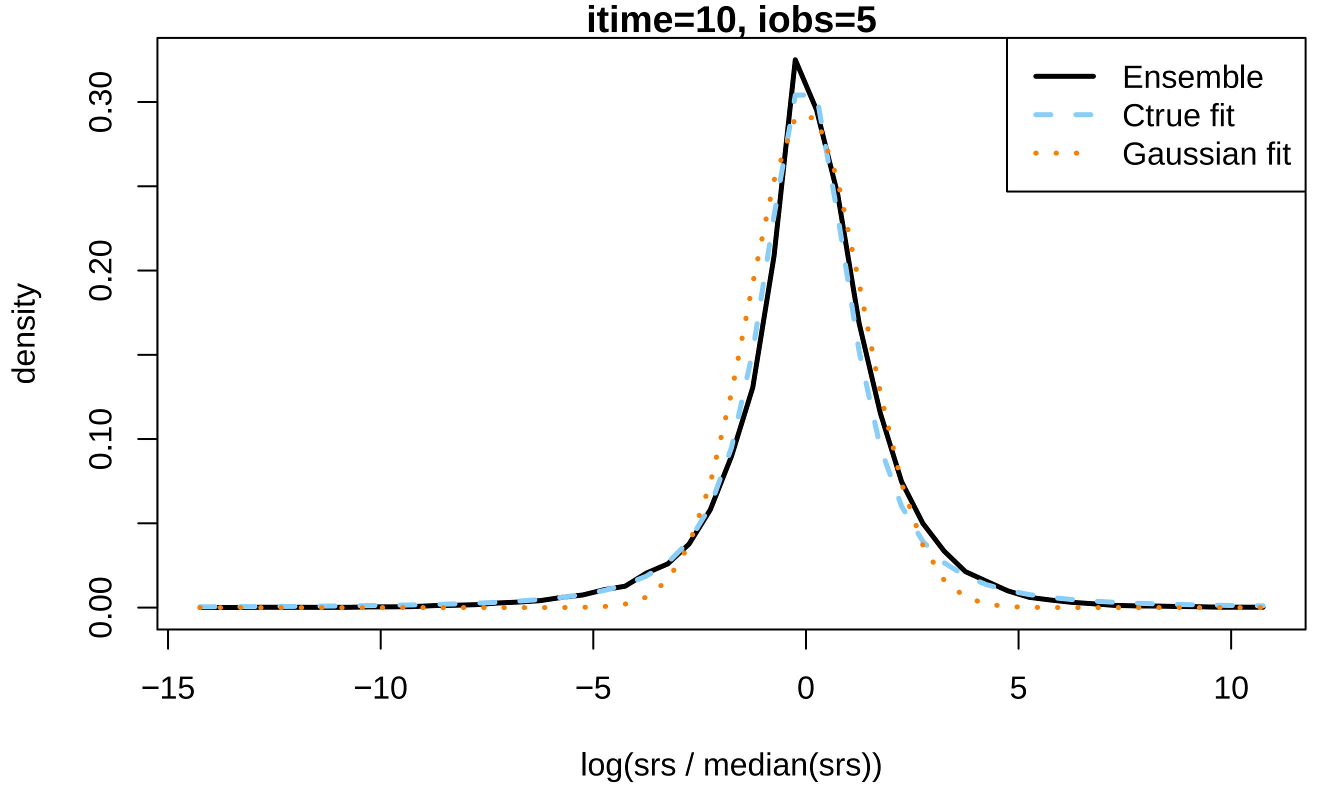

itime=10, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))

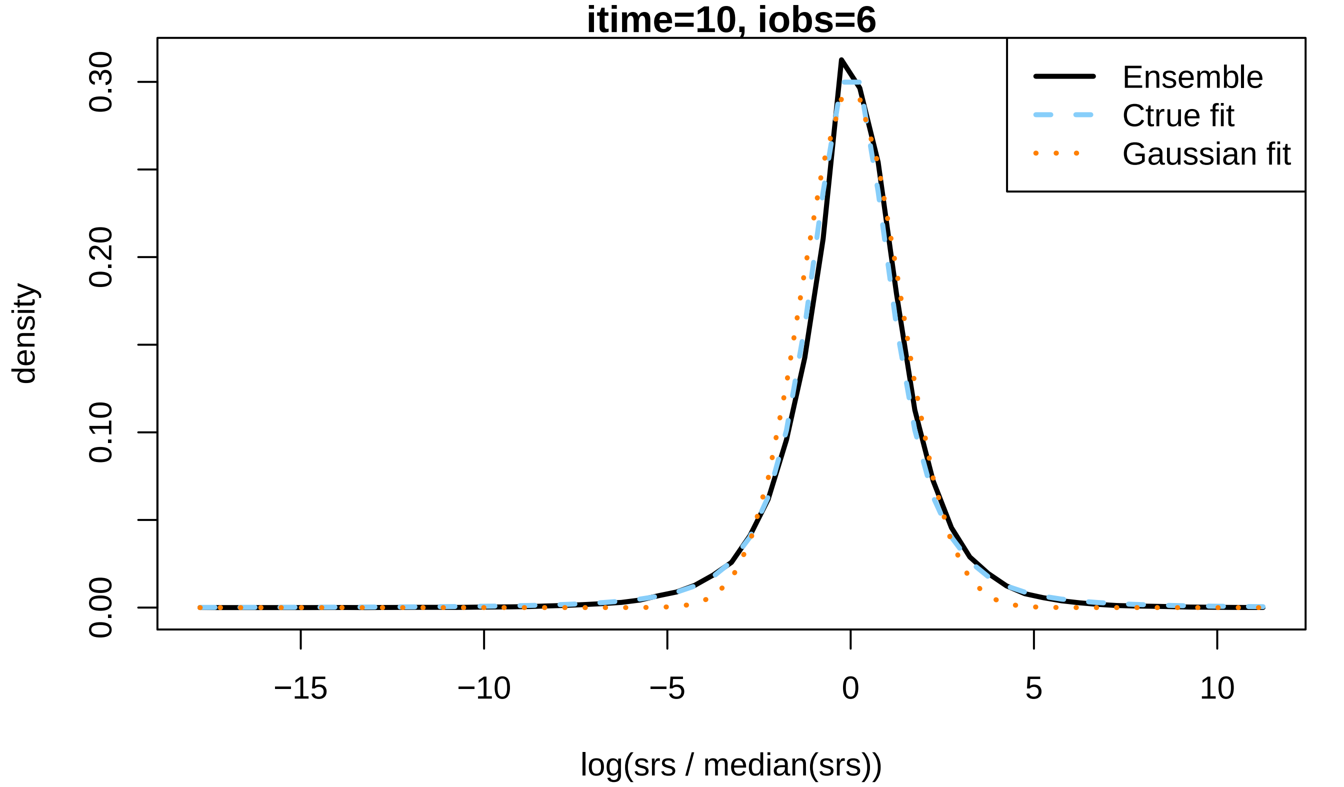

itime=10, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

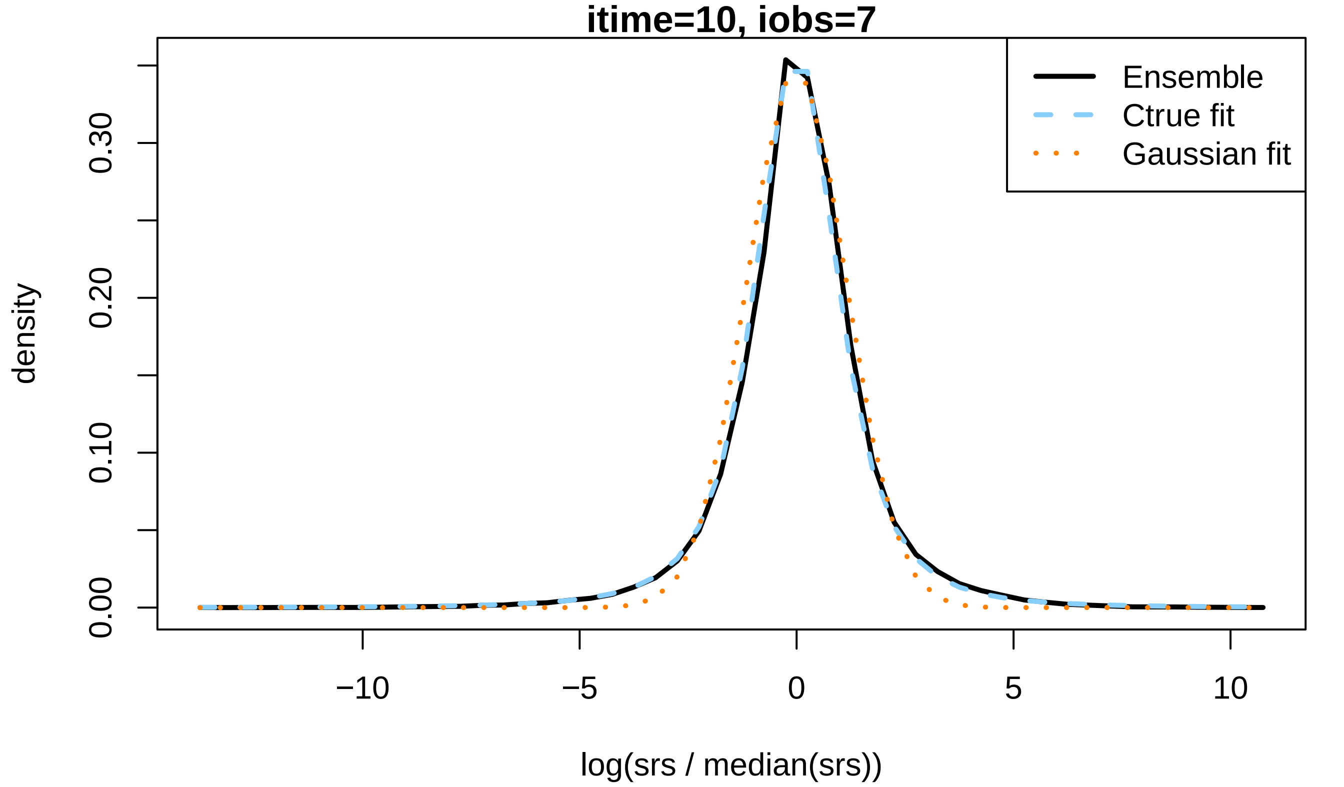

itime=10, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

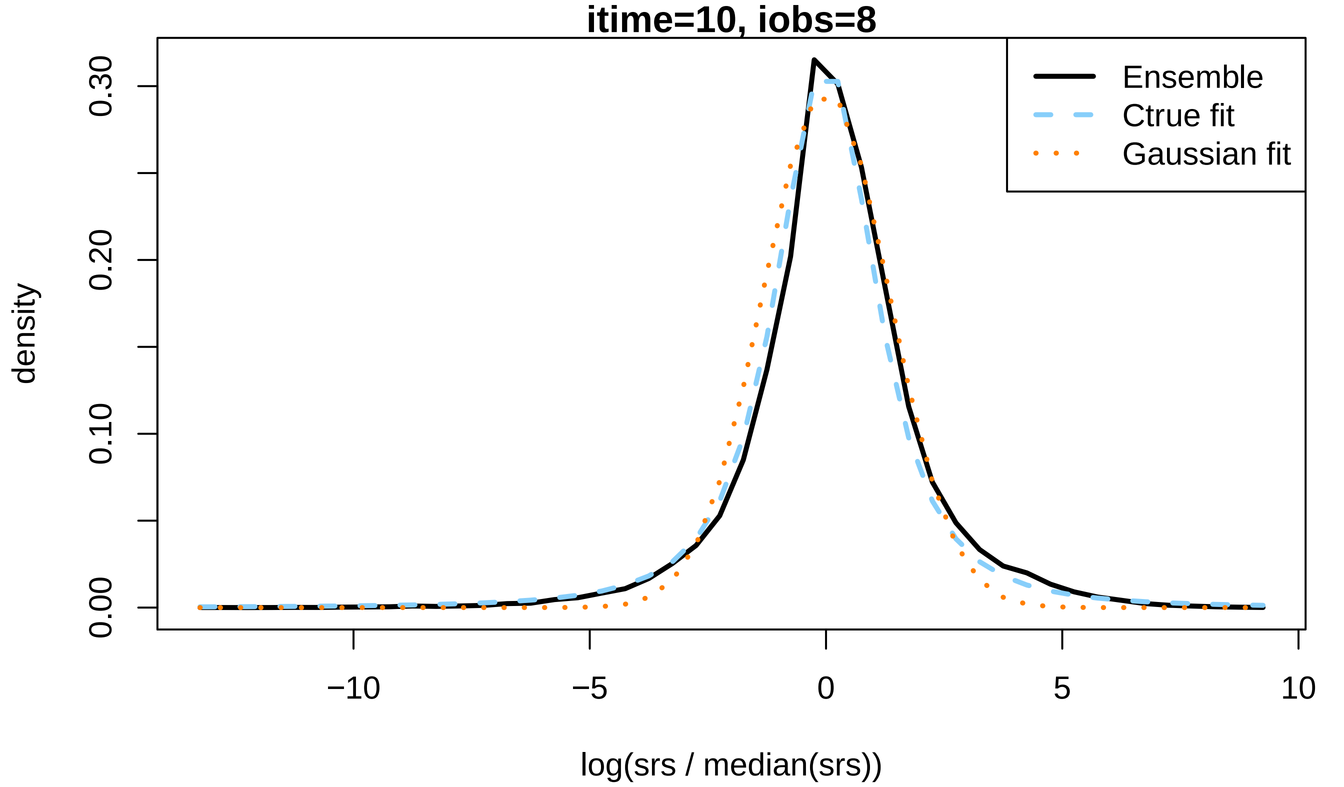

itime=10, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

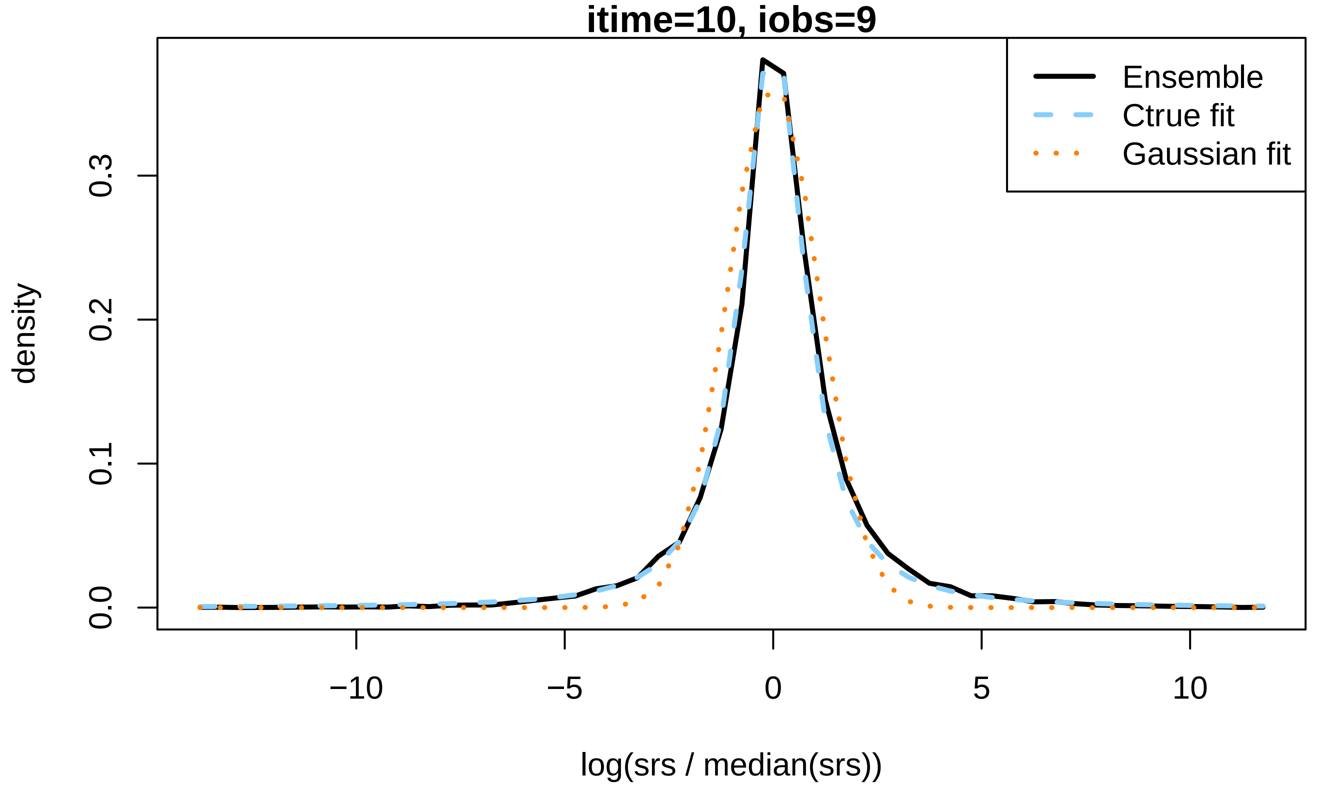

itime=10, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−10
−5
0
5
10
log(srs / median(srs))


itime=10, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

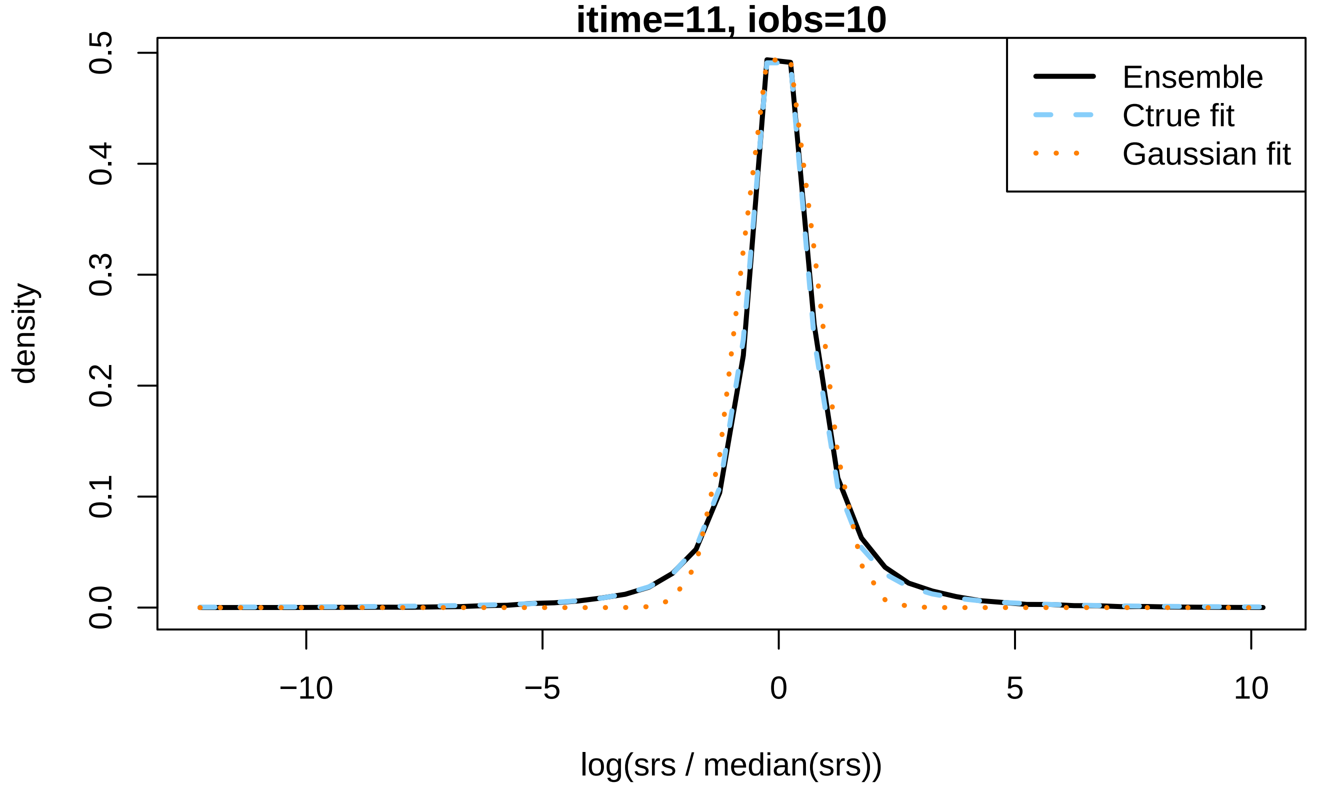

itime=10, iobs=6
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit


itime=10, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))

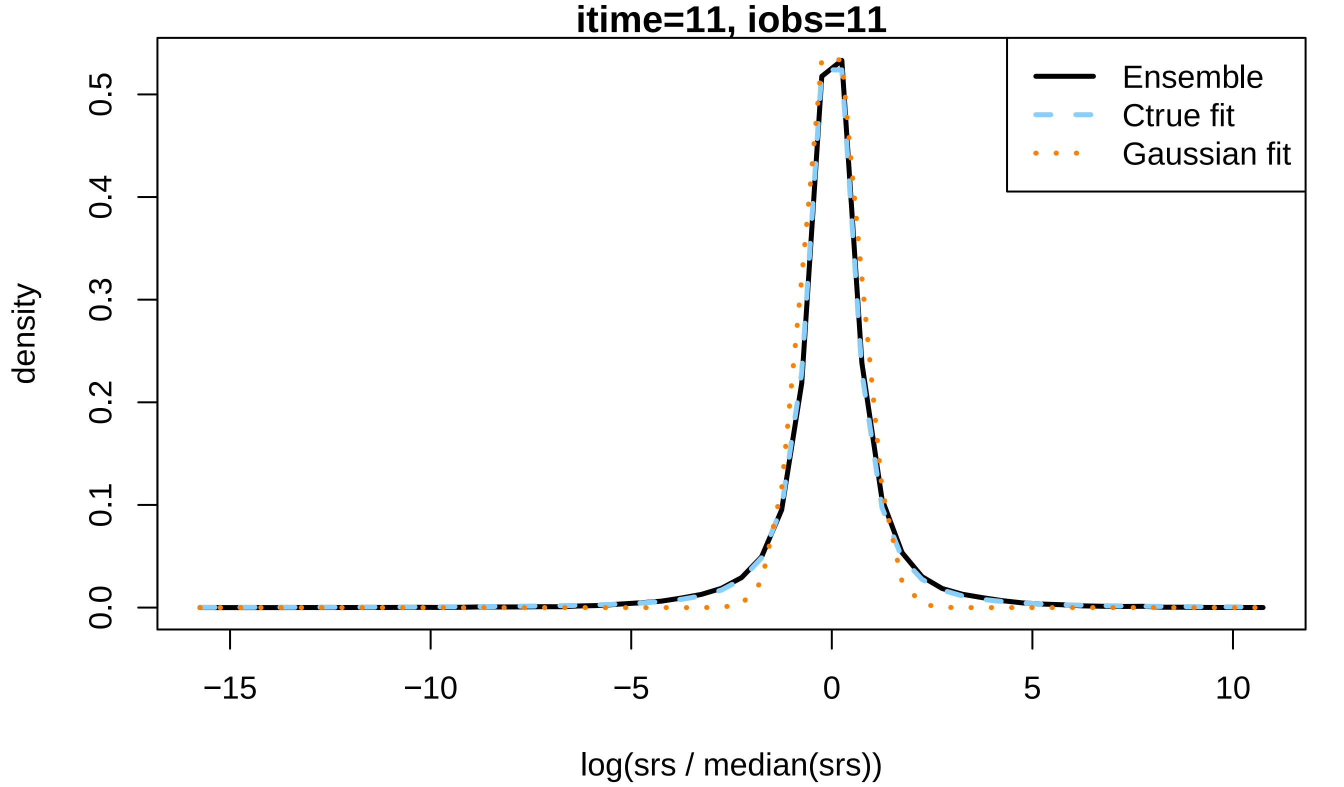

itime=10, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))



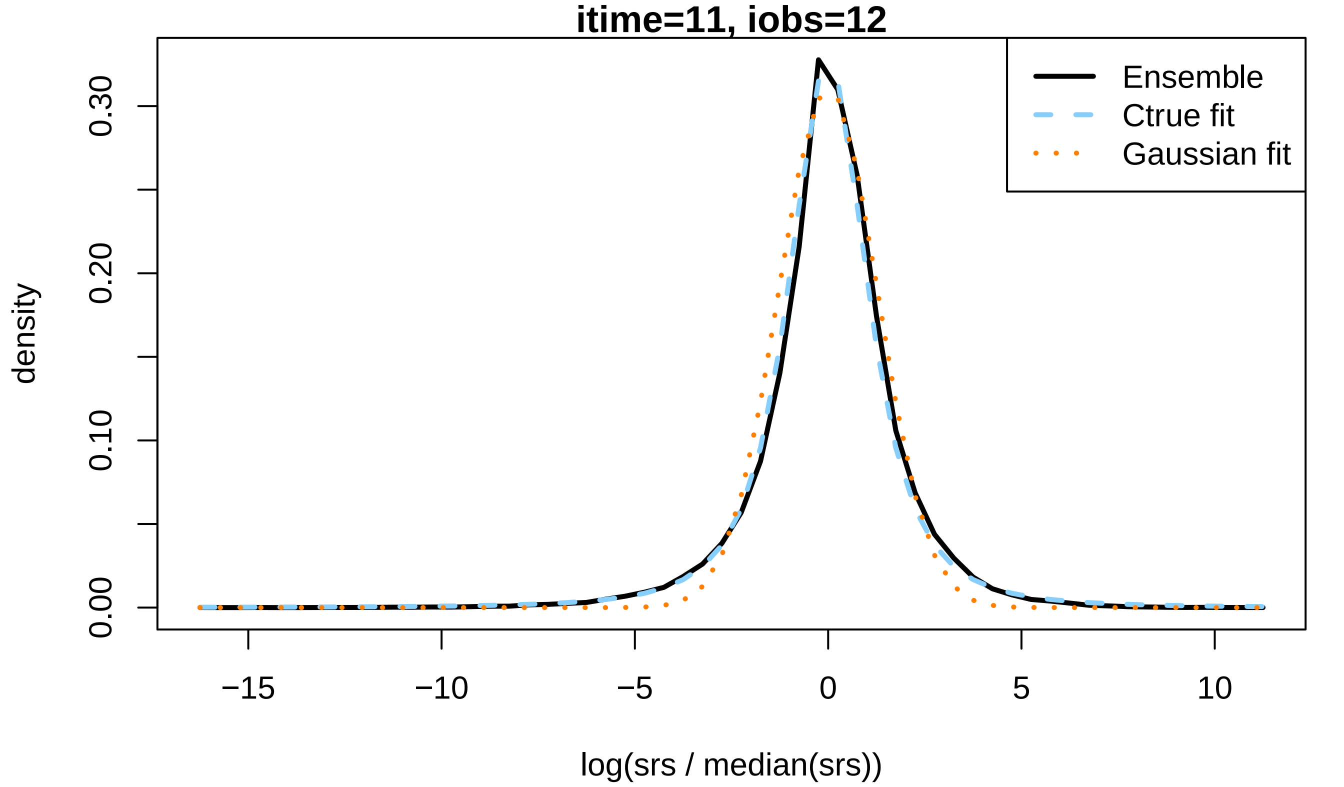
itime=10, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−10
−5
0
5
10
log(srs / median(srs))

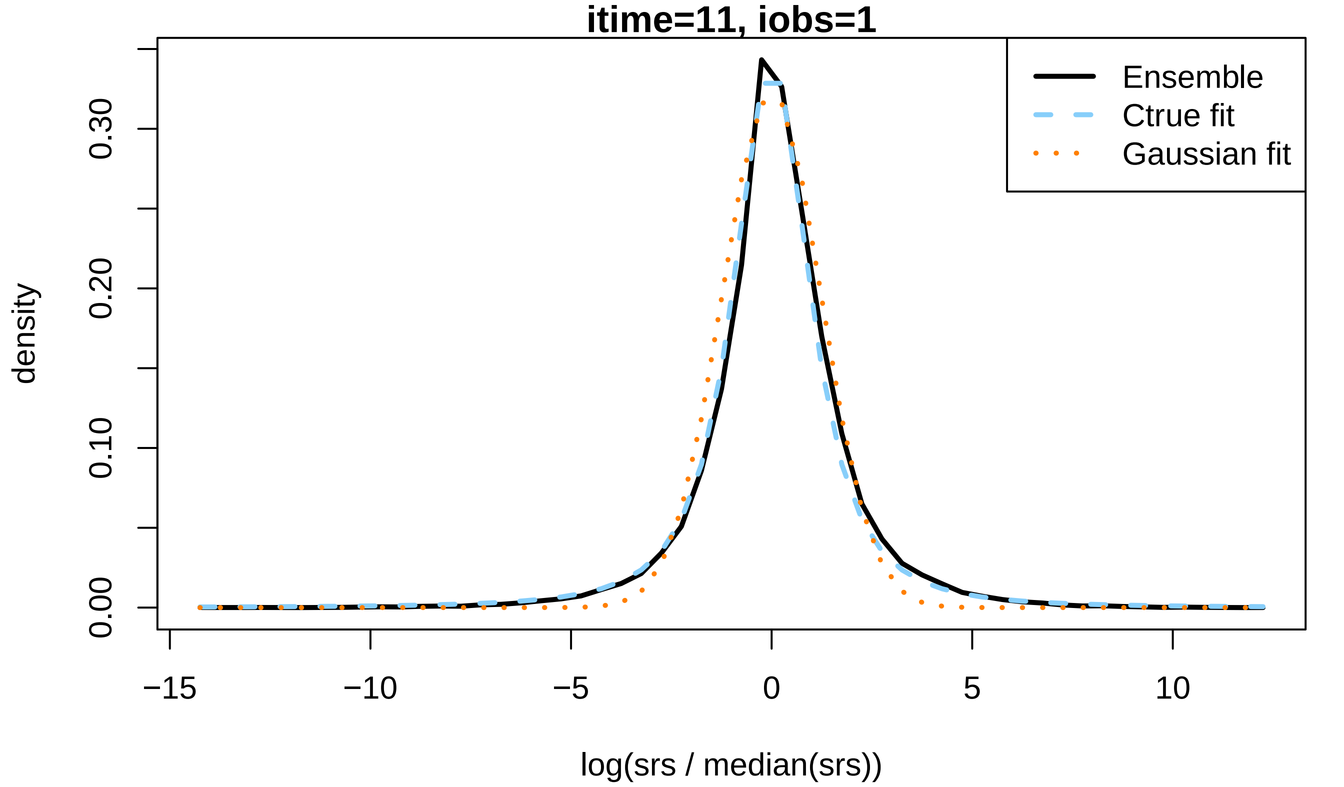


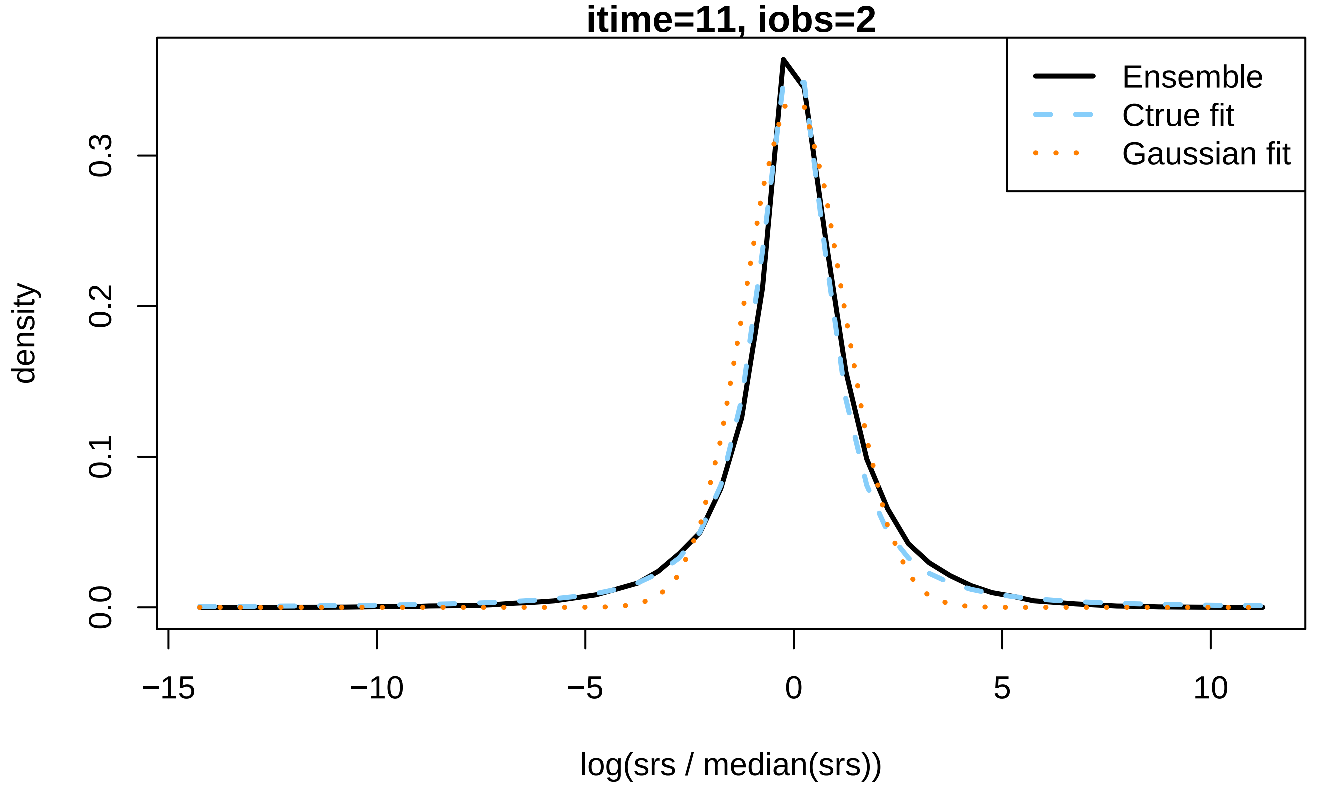
itime=11, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−10
−5
0
5
10
log(srs / median(srs))

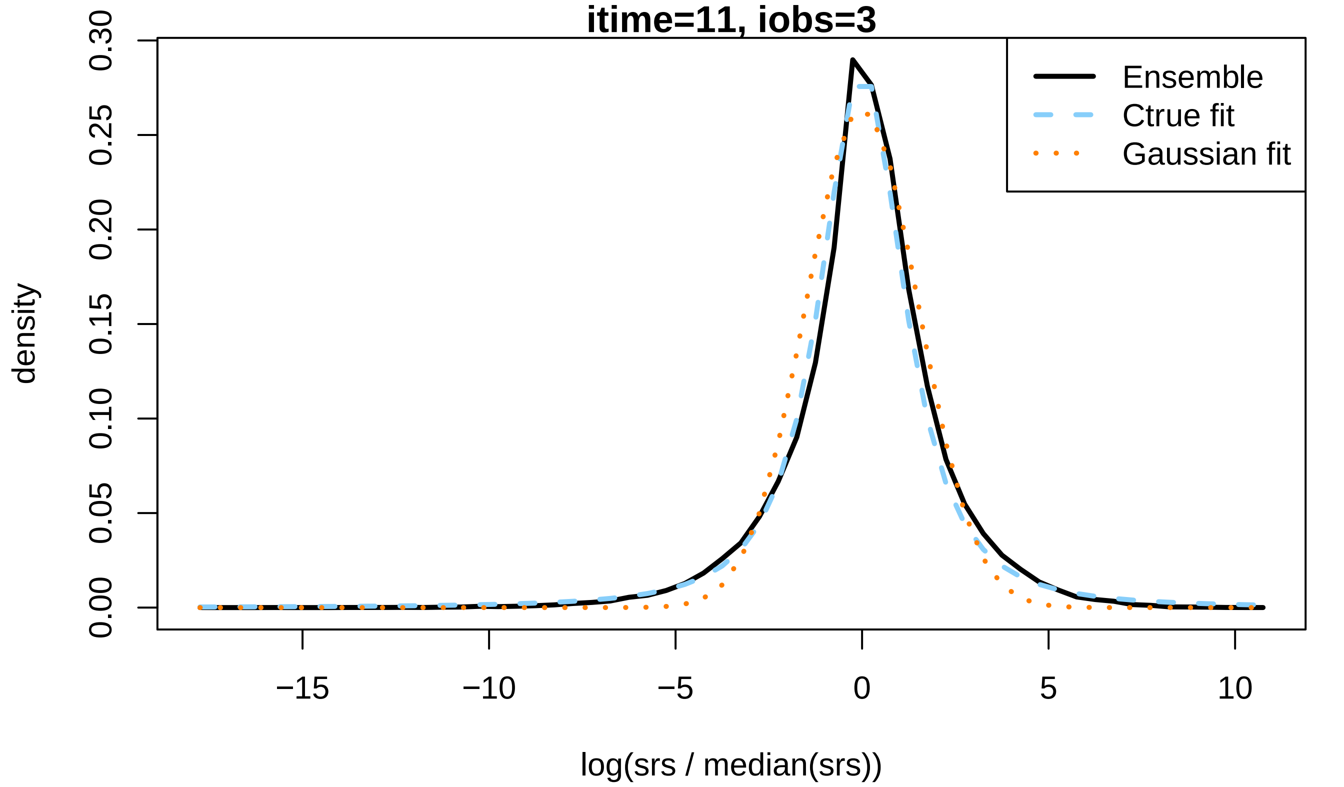


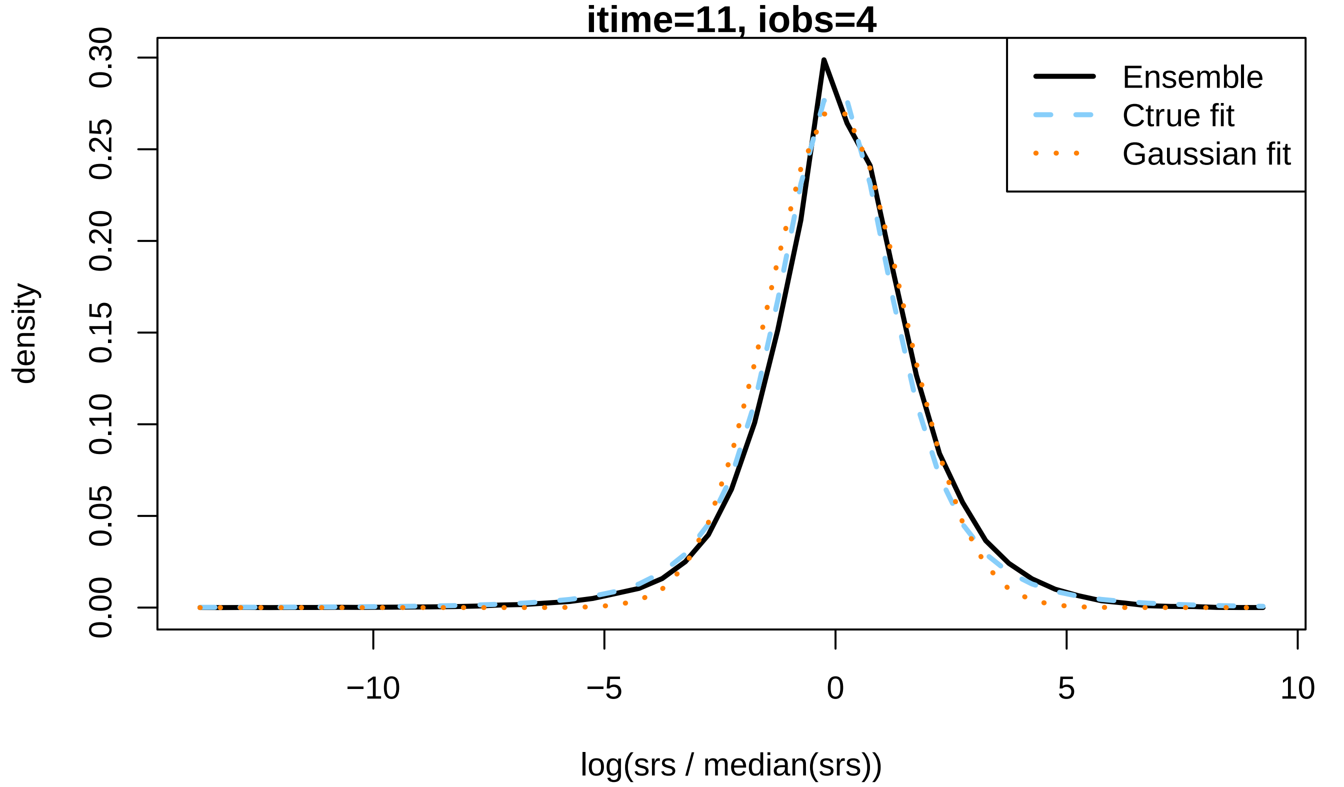
itime=11, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=11, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

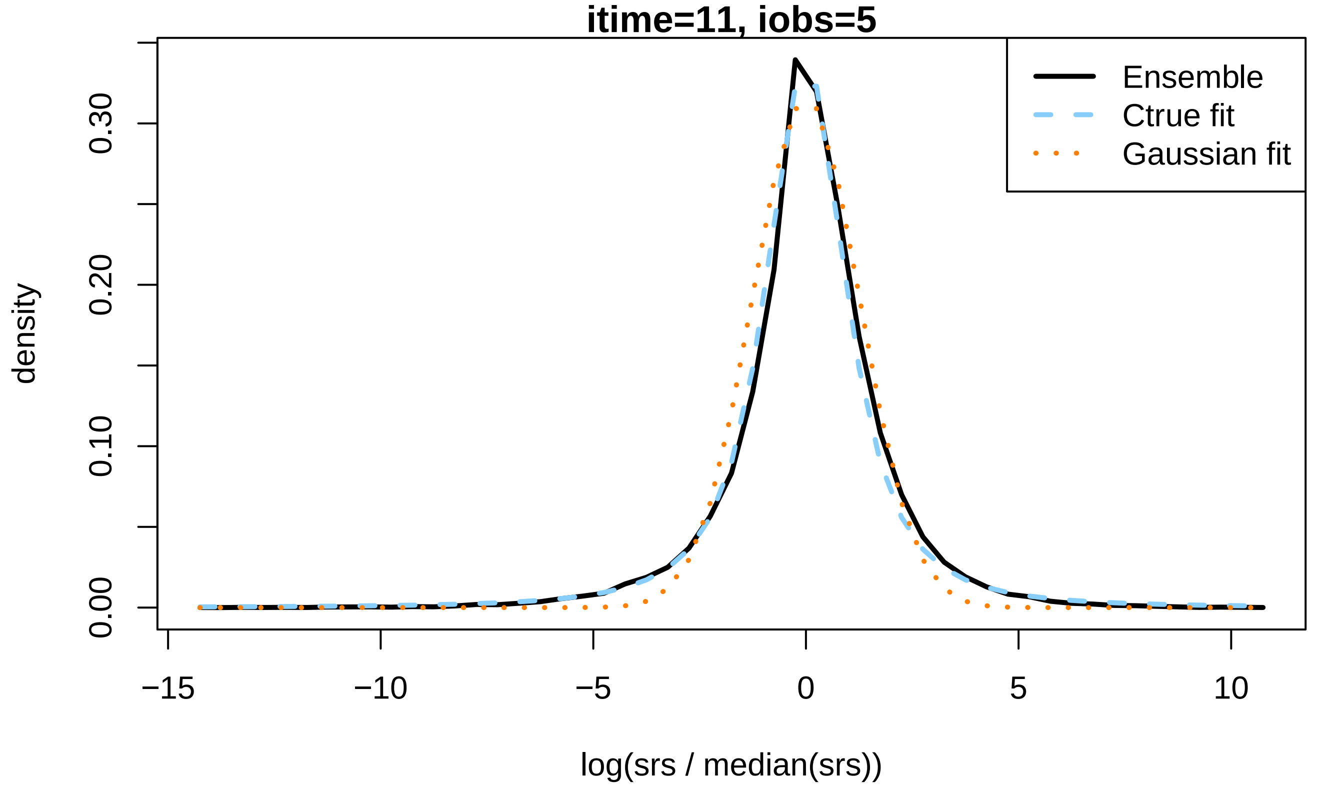

itime=11, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=11, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

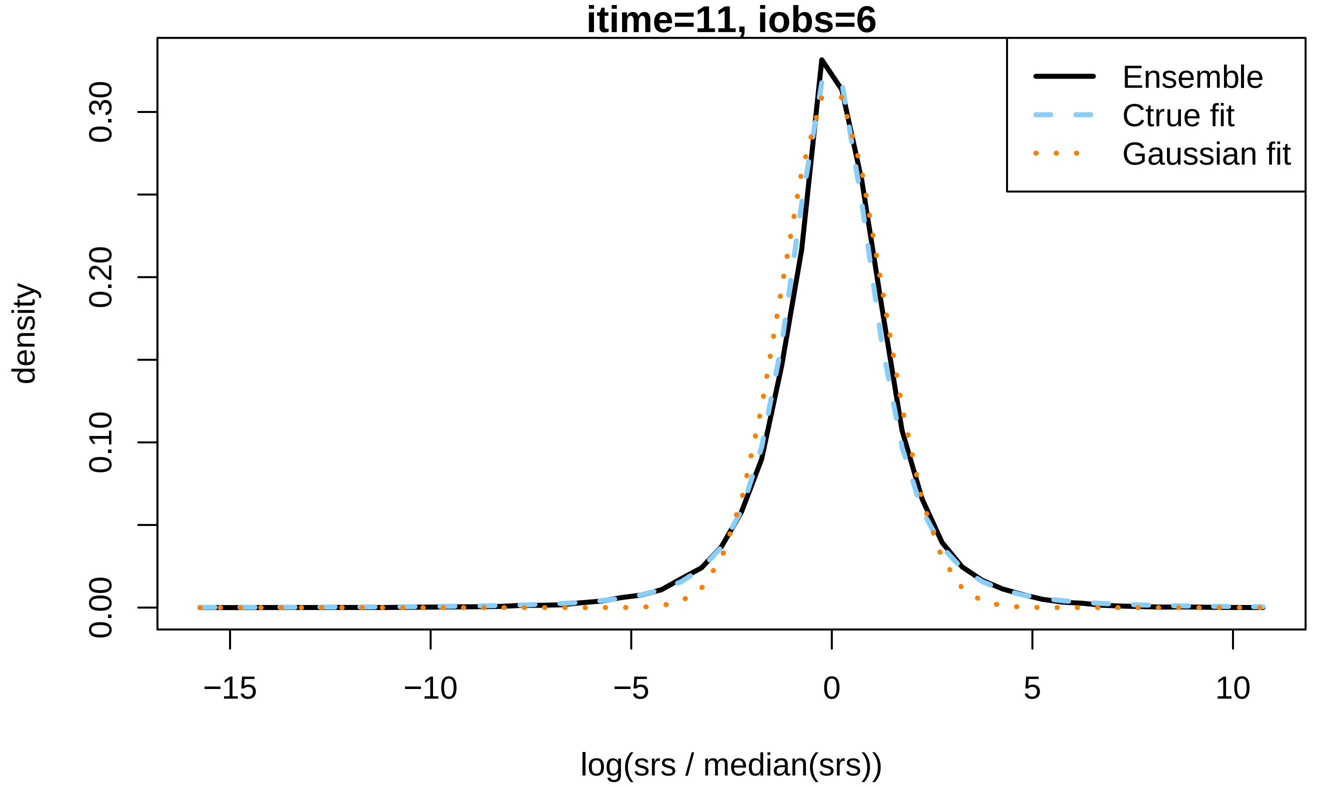

itime=11, iobs=3
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

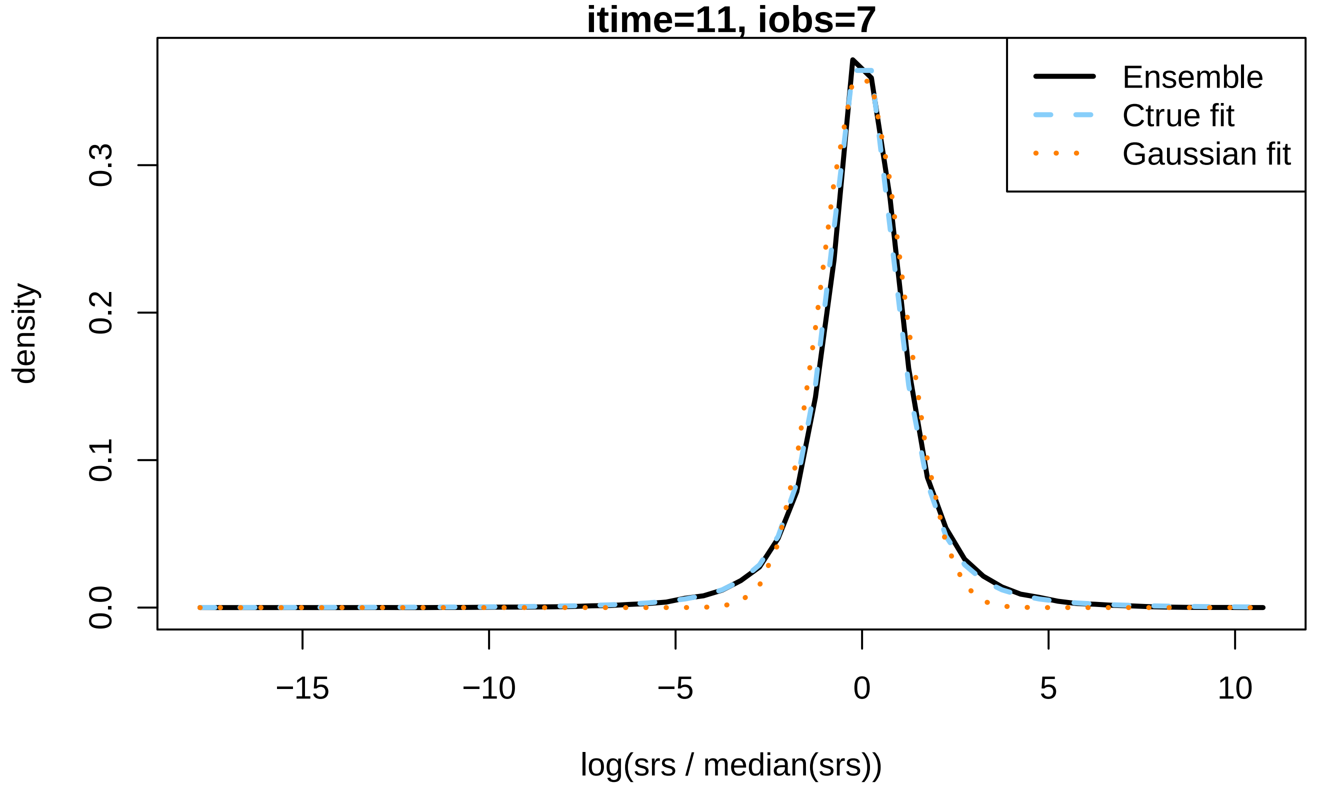


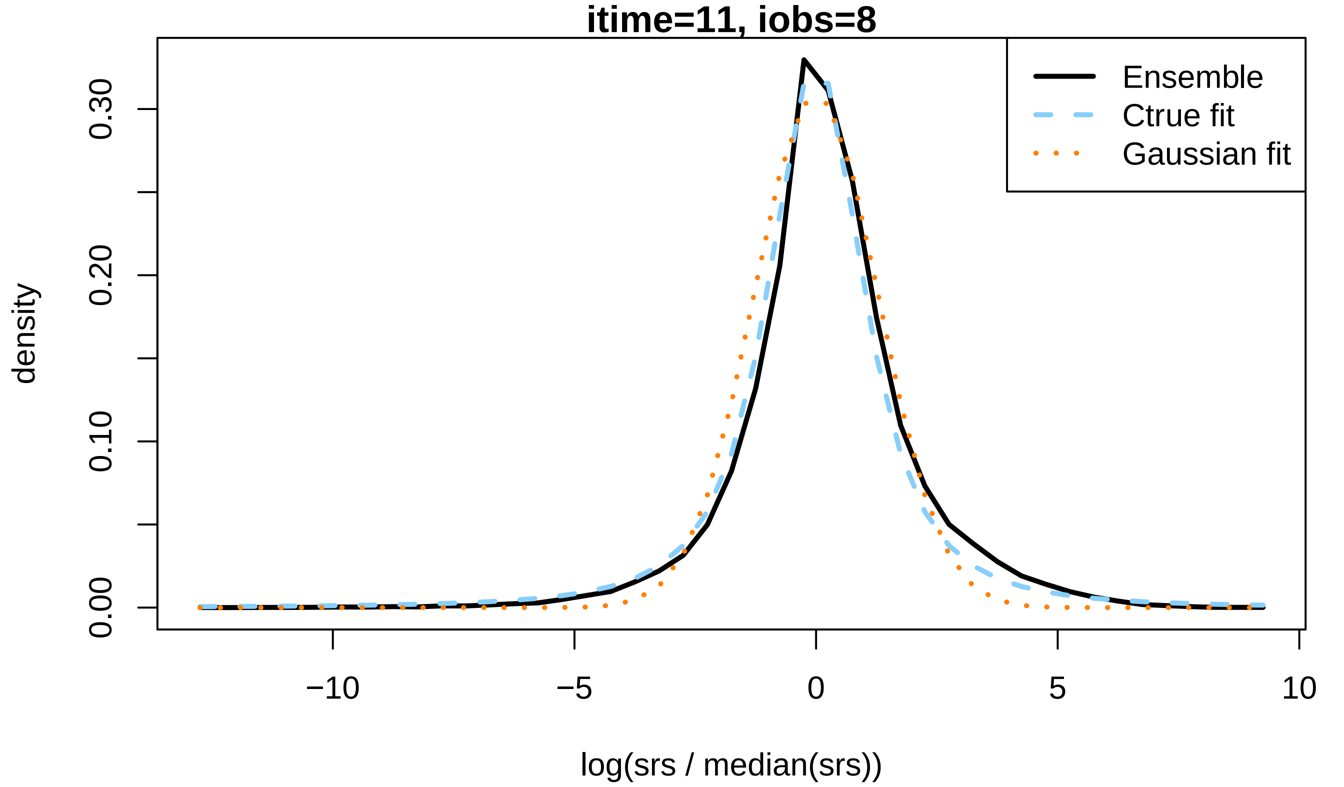
itime=11, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−10
−5
0
5
10
log(srs / median(srs))



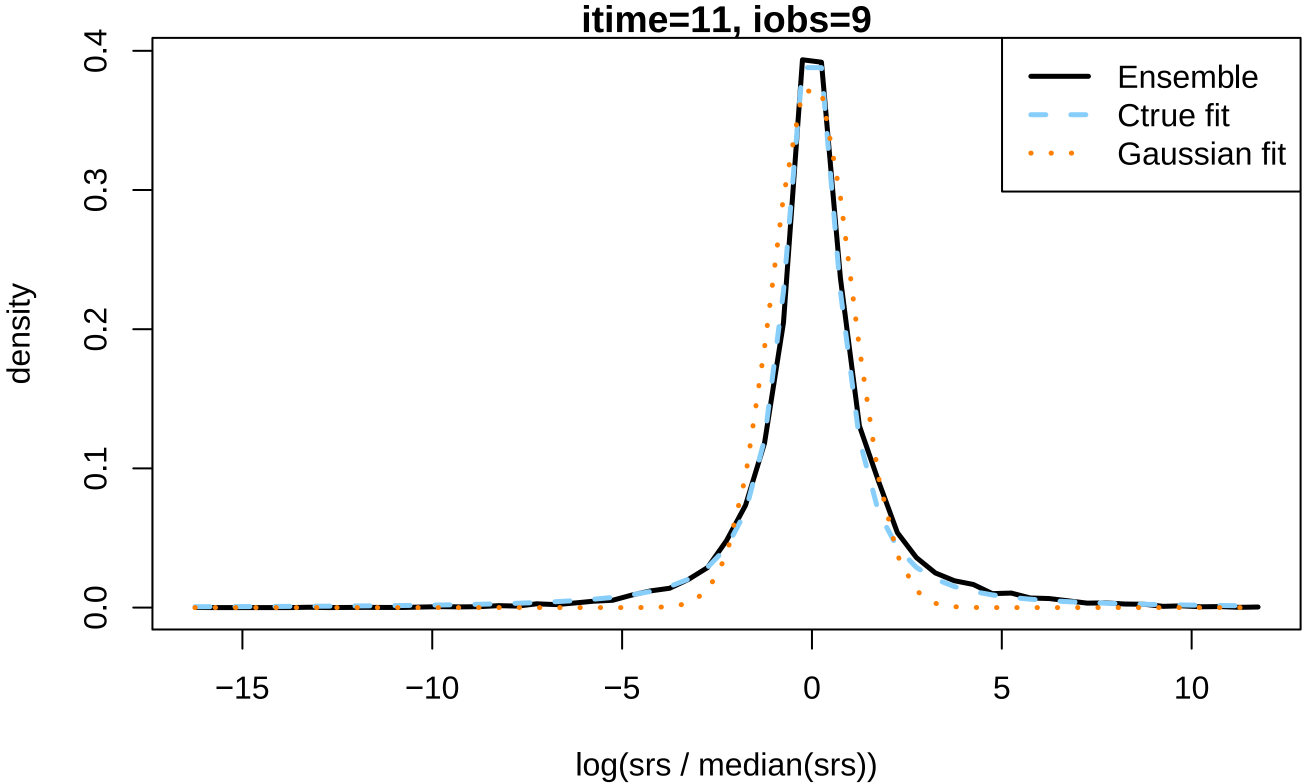
itime=11, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=11, iobs=6
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

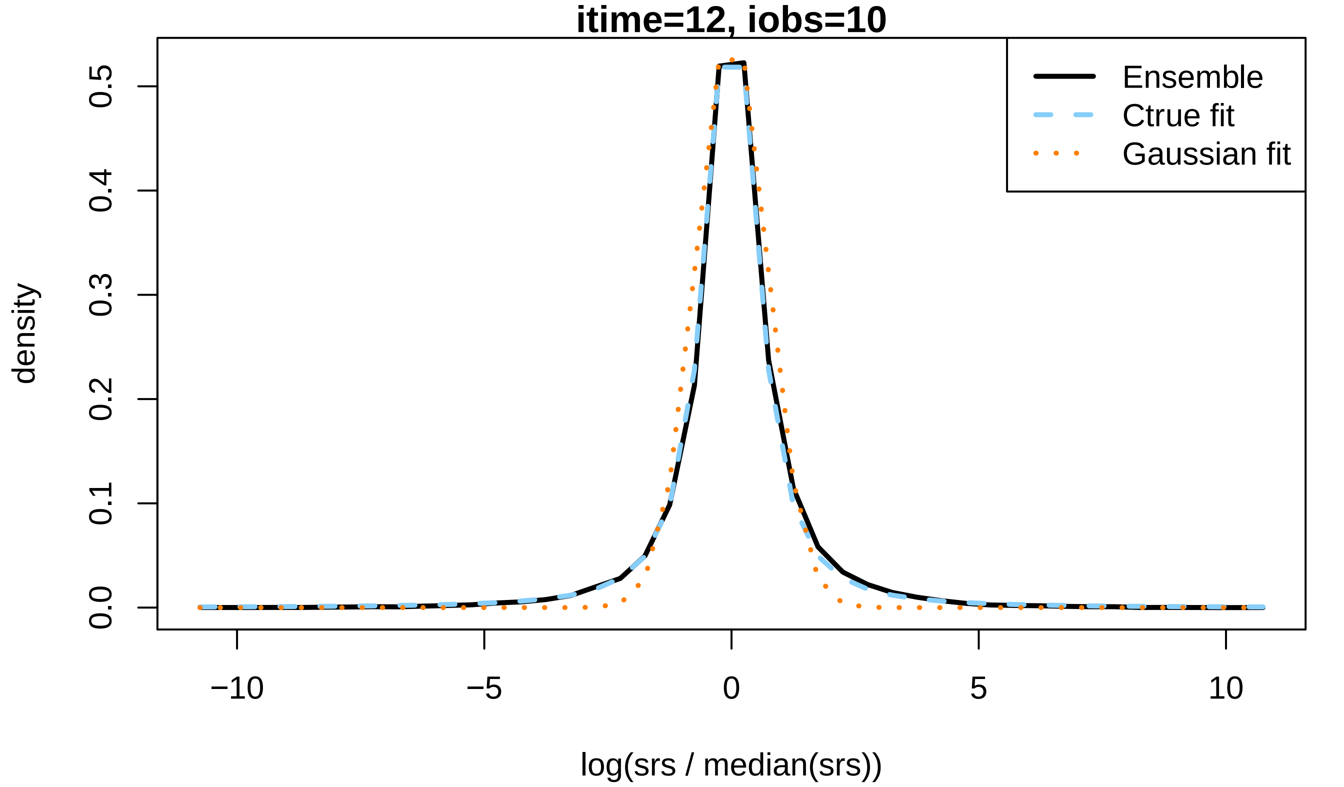

itime=11, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

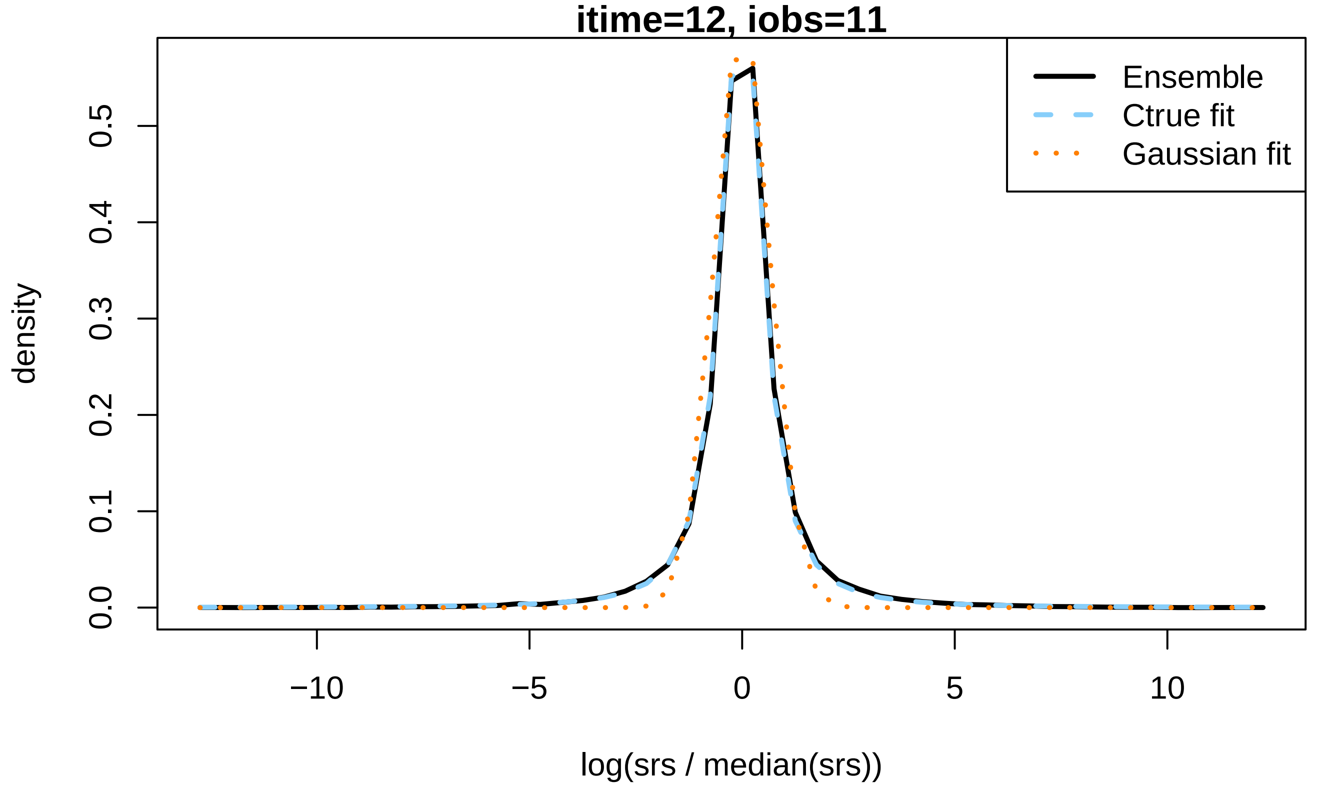


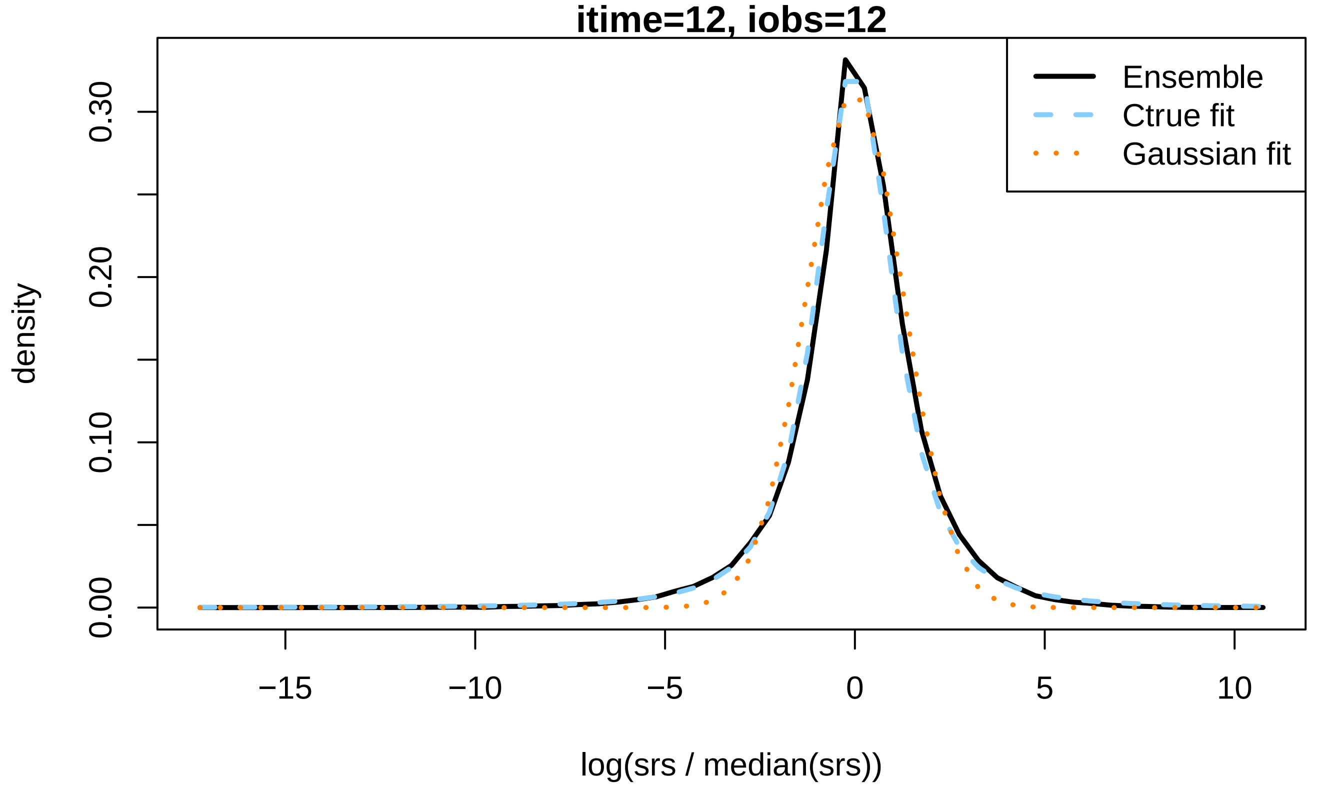
itime=11, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))



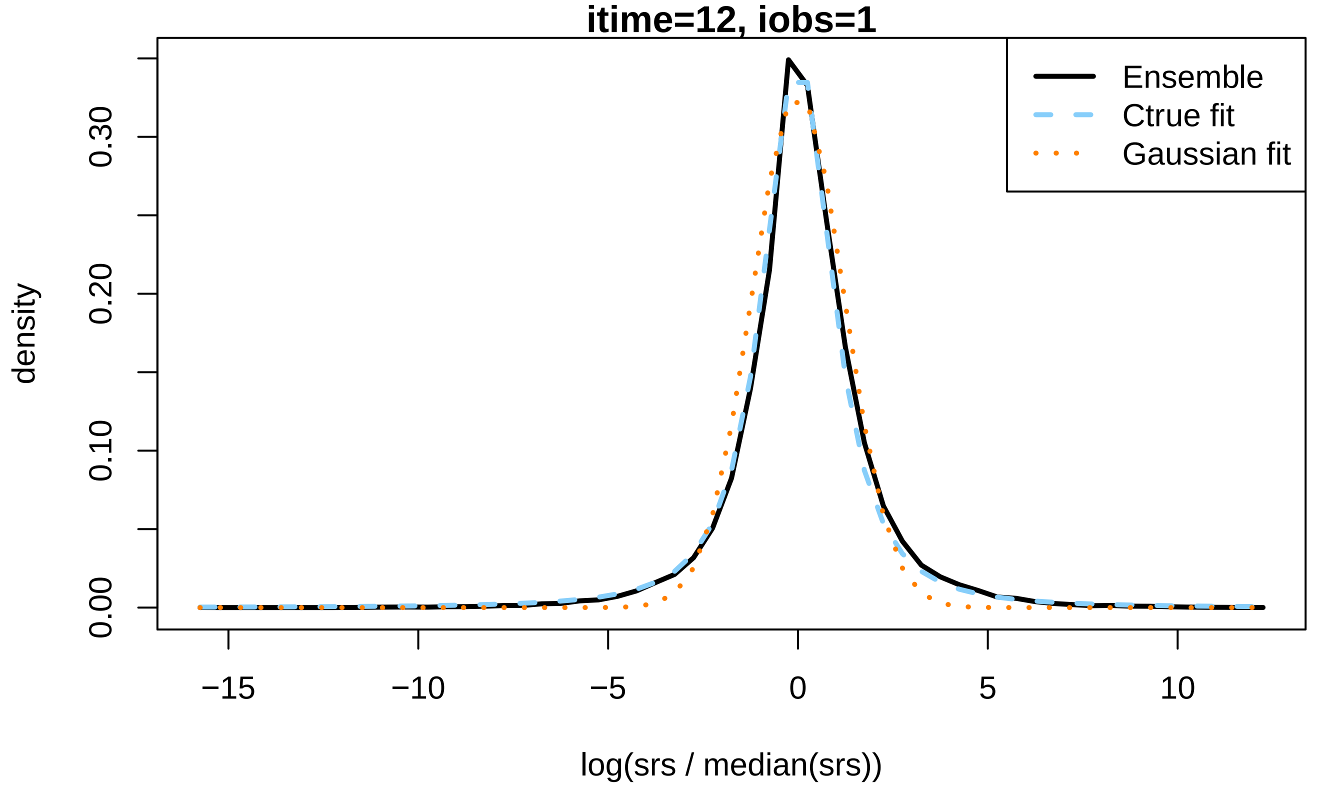
itime=11, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=12, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−10
−5
0
5
10
log(srs / median(srs))

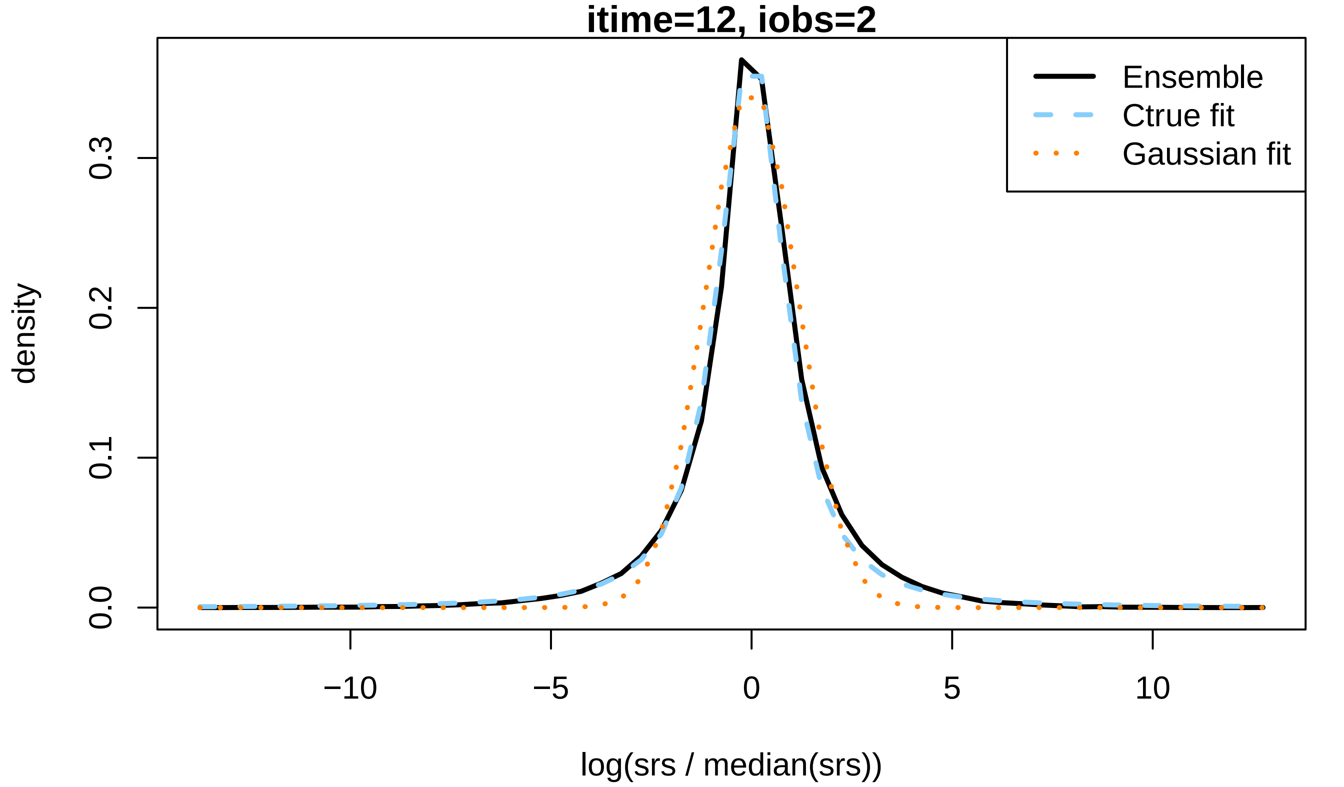

itime=12, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−10
−5
0
5
10
log(srs / median(srs))

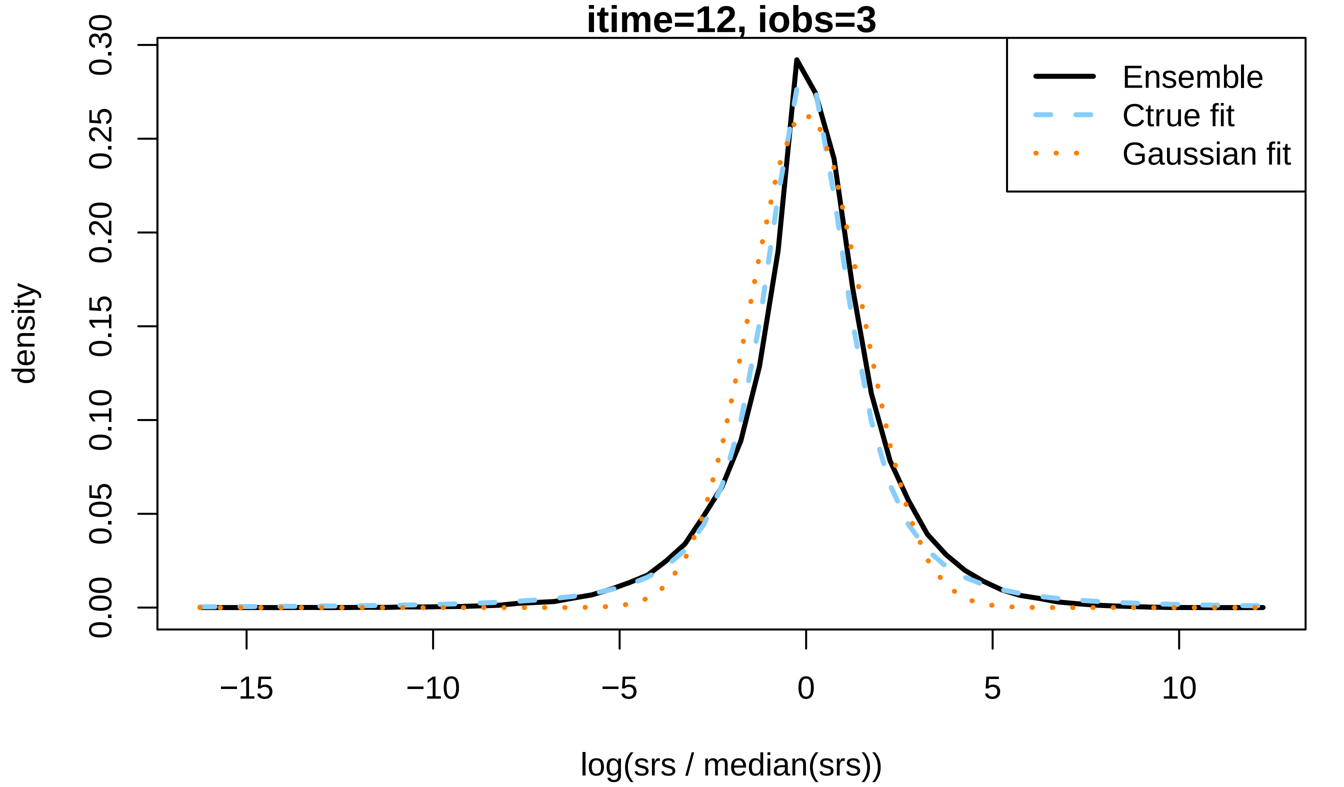


itime=12, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

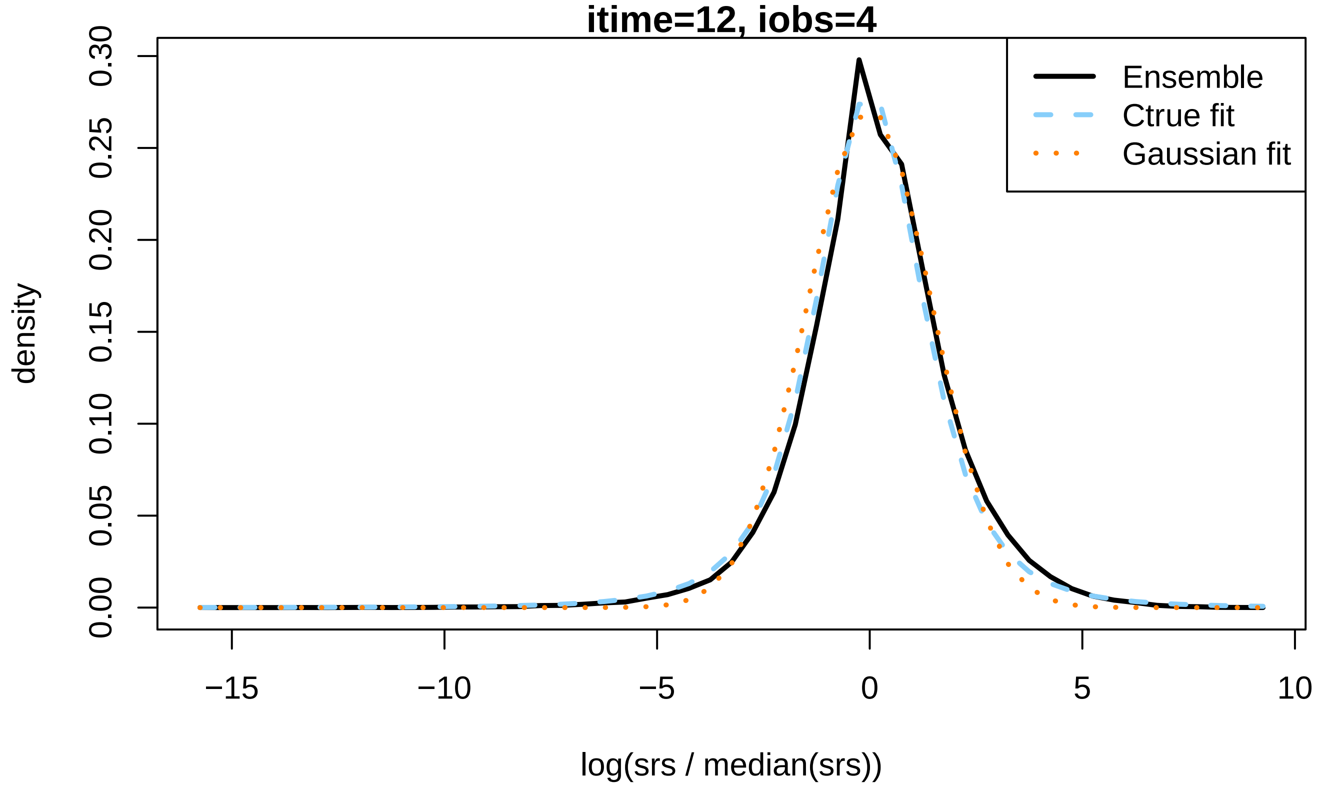


itime=12, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

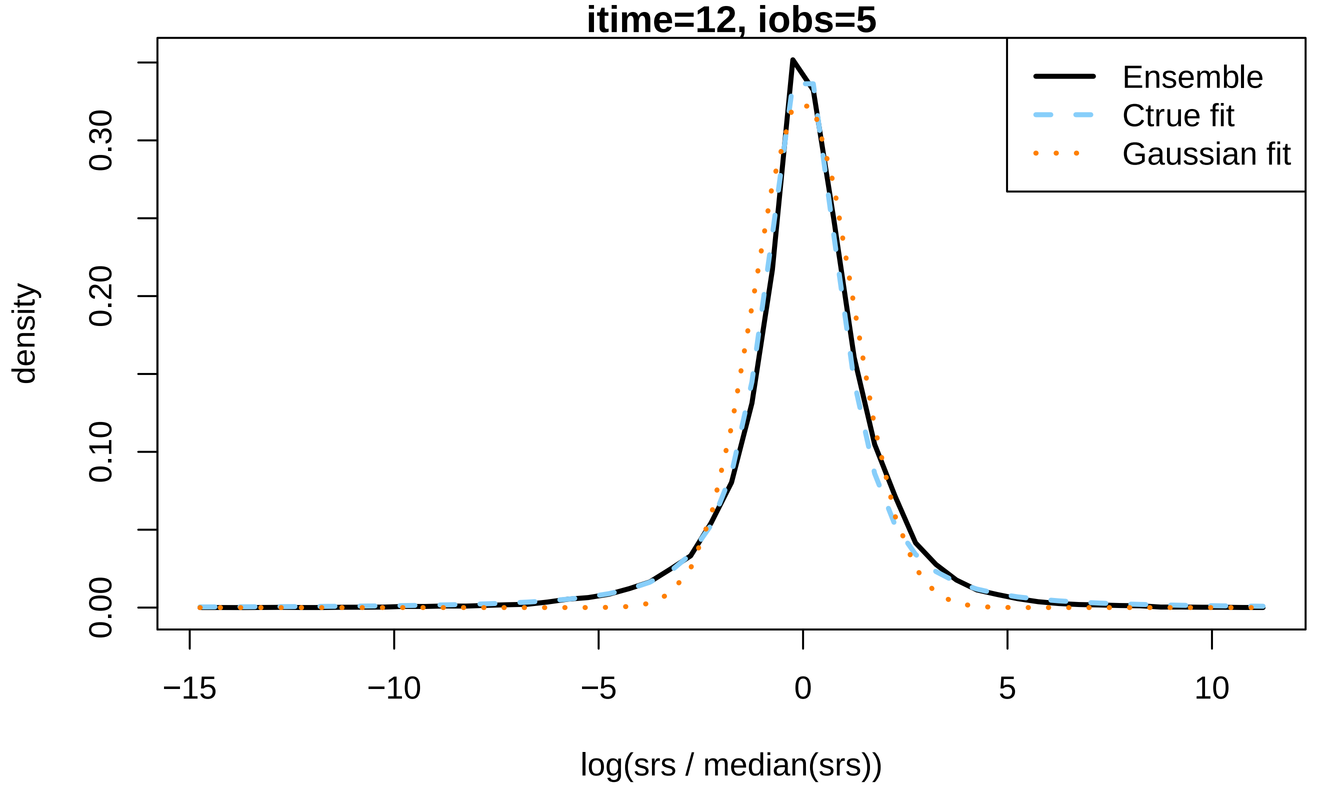

itime=12, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−10
−5
0
5
10
log(srs / median(srs))

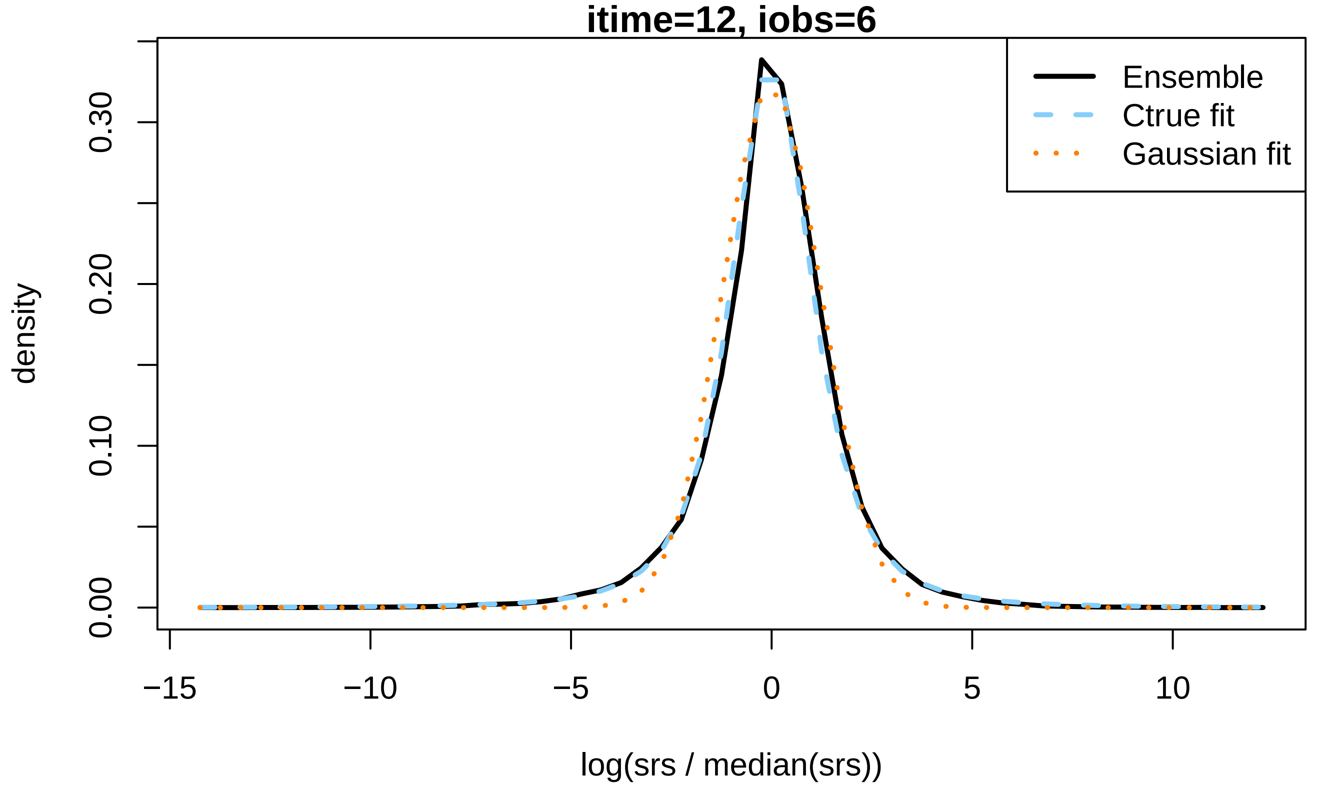


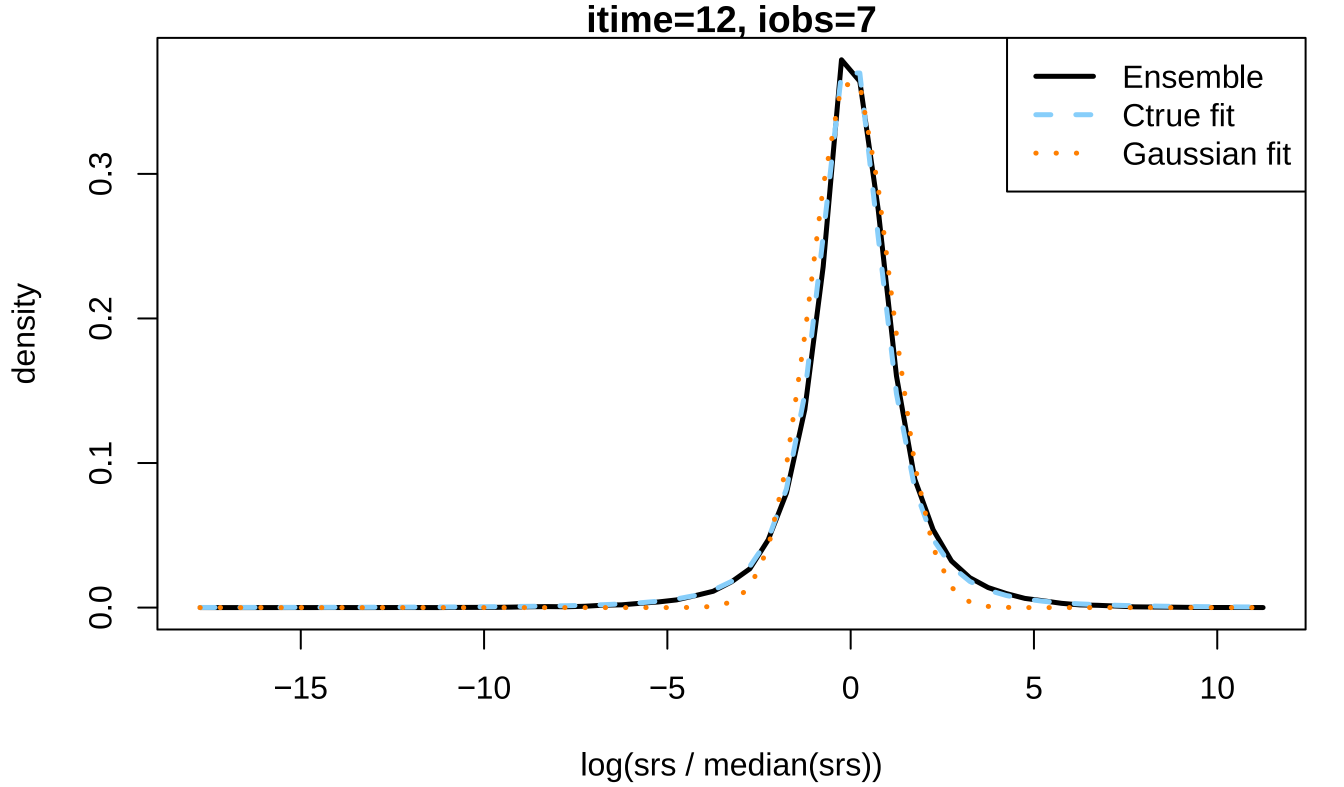
itime=12, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

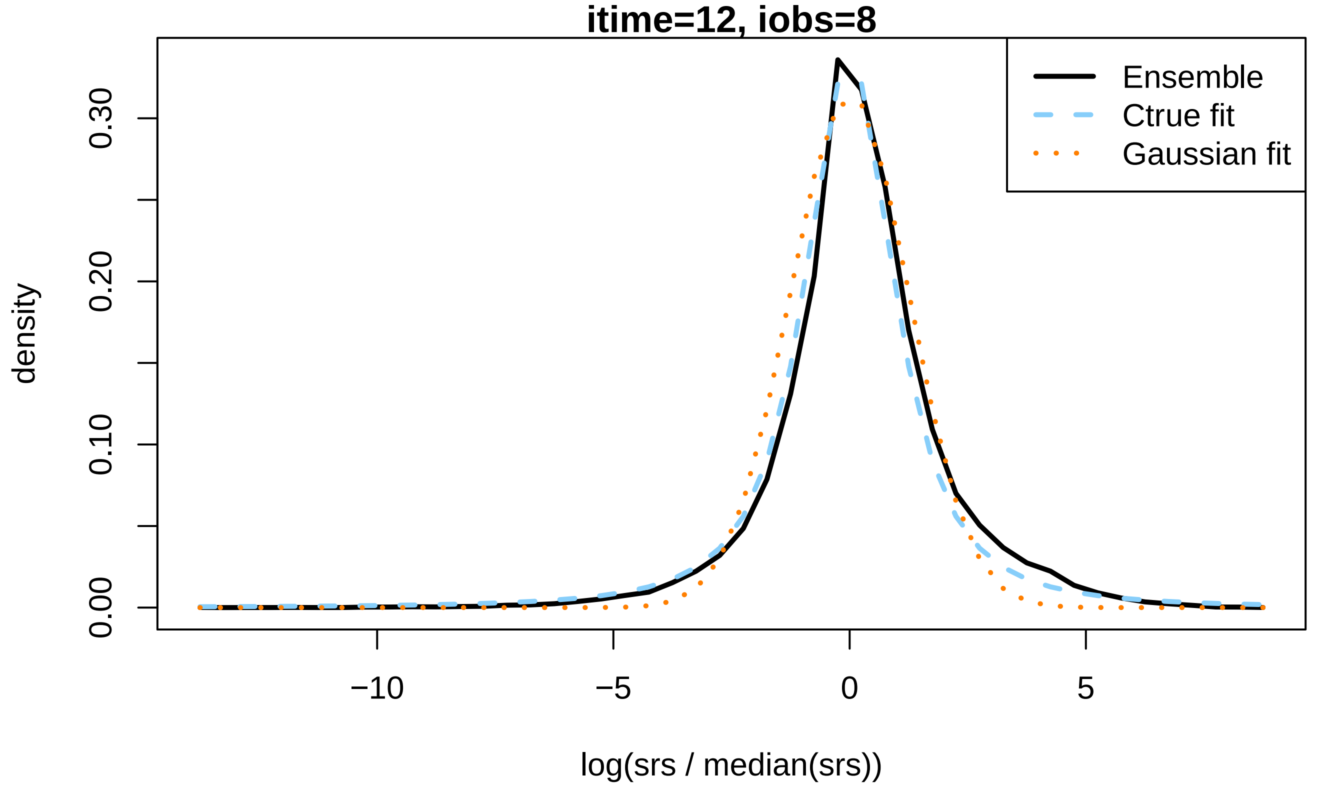


itime=12, iobs=4
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

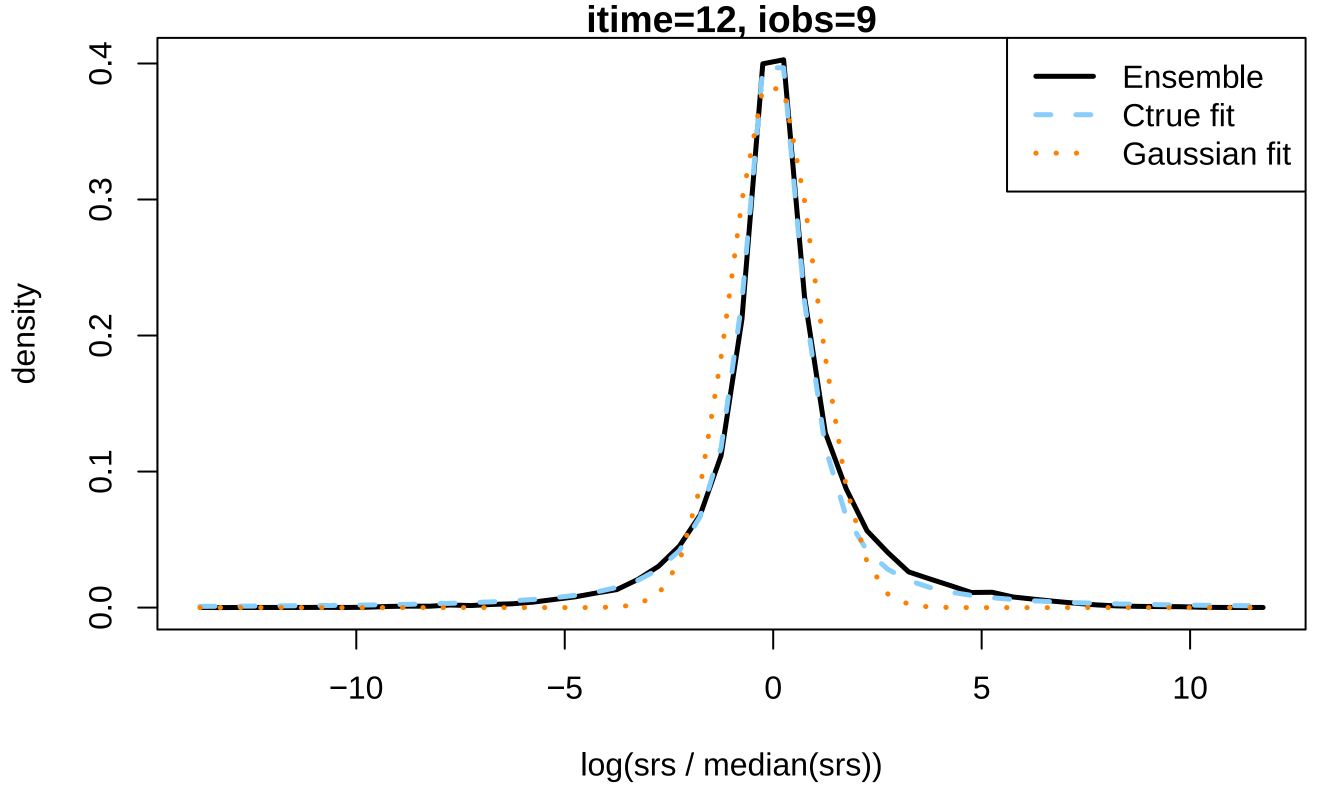

itime=12, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

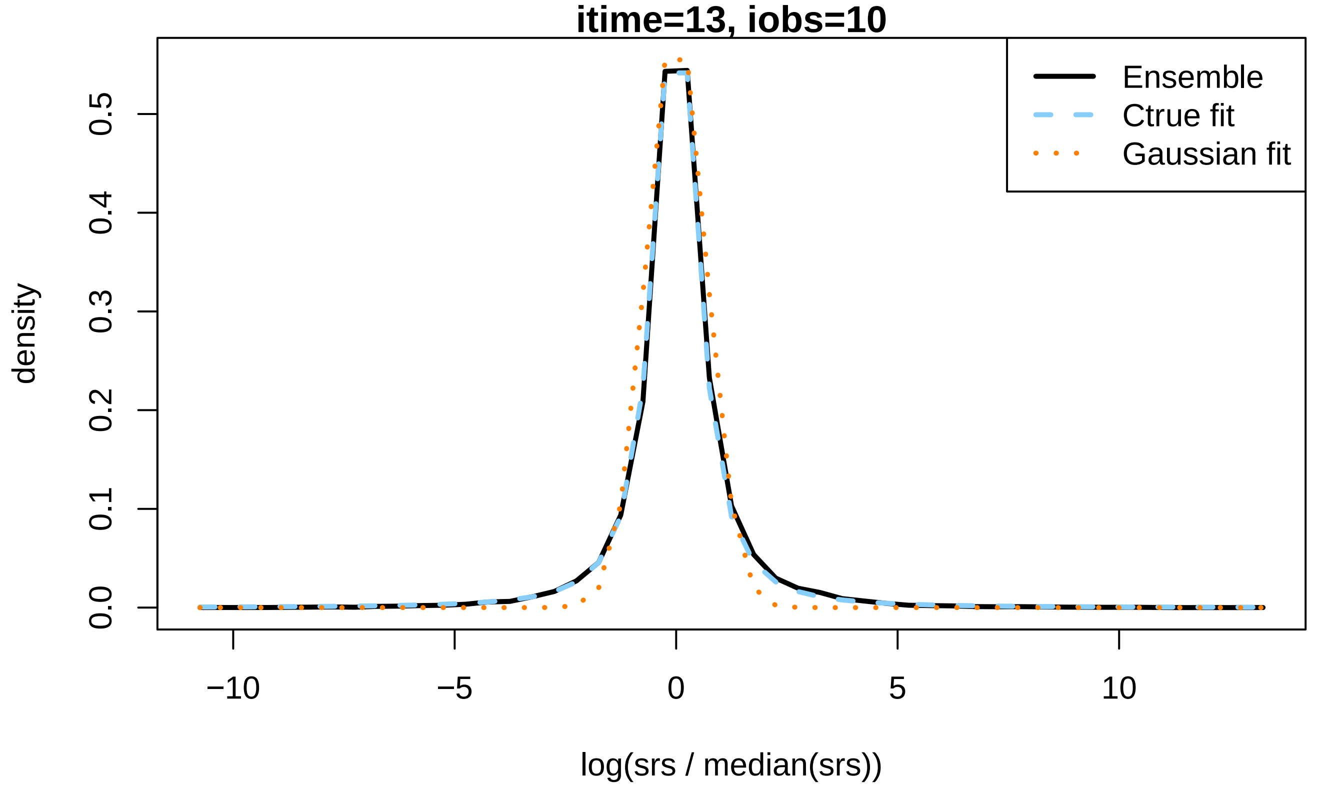

itime=12, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=12, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

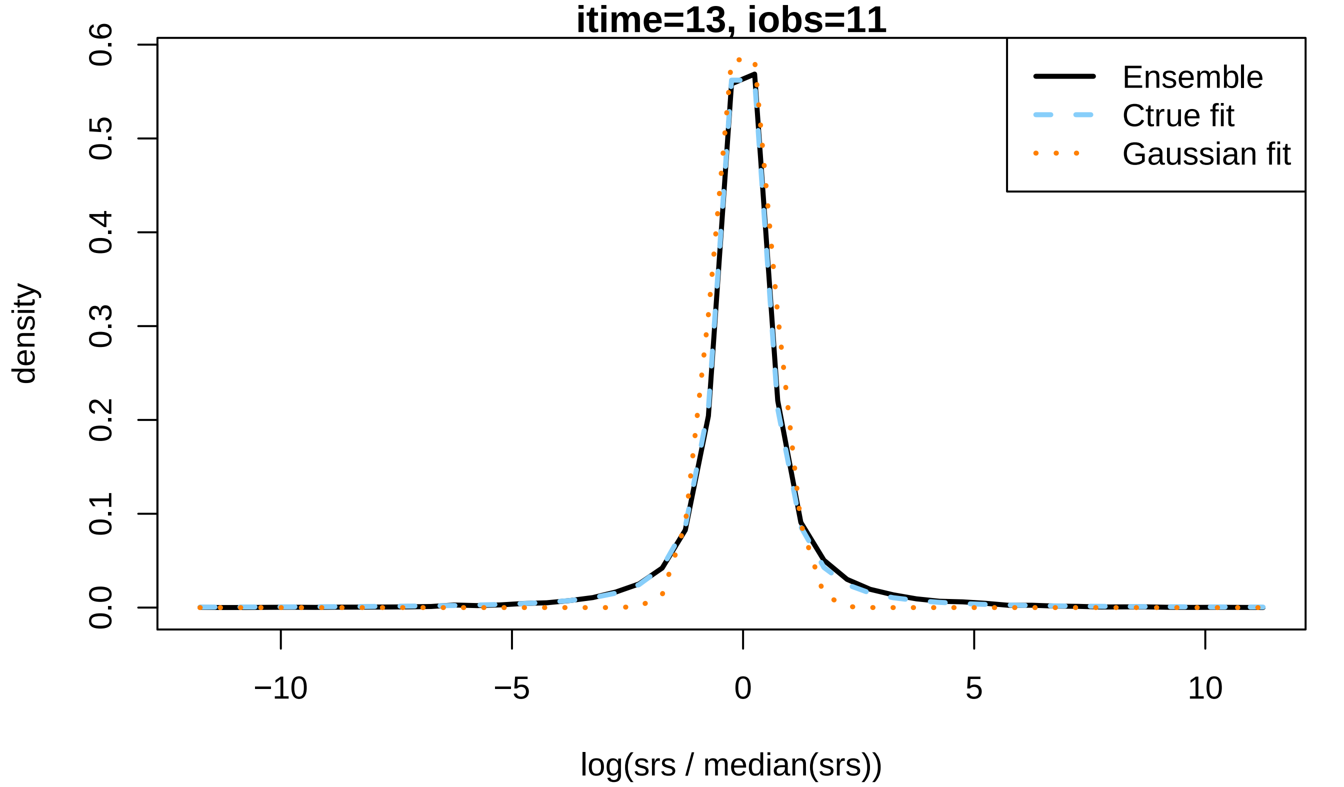


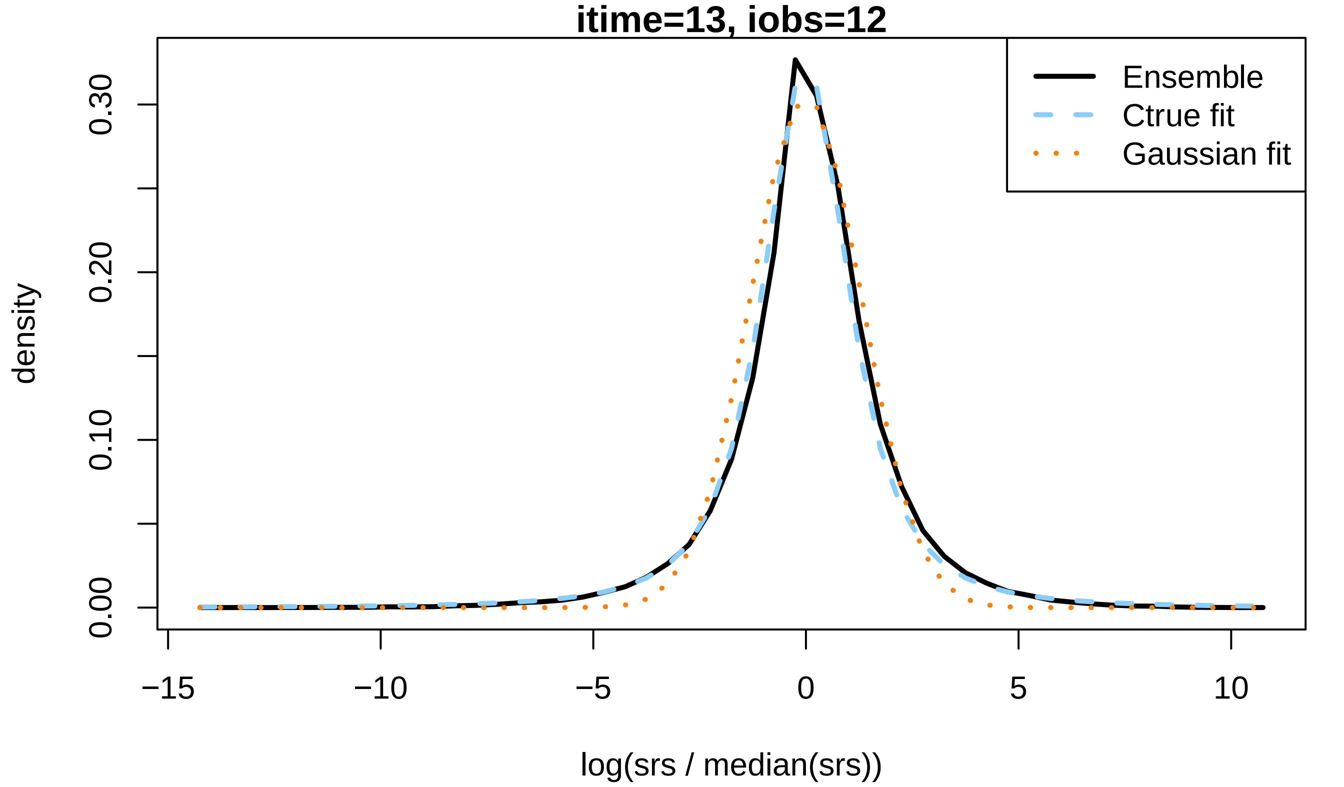
itime=12, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
log(srs / median(srs))

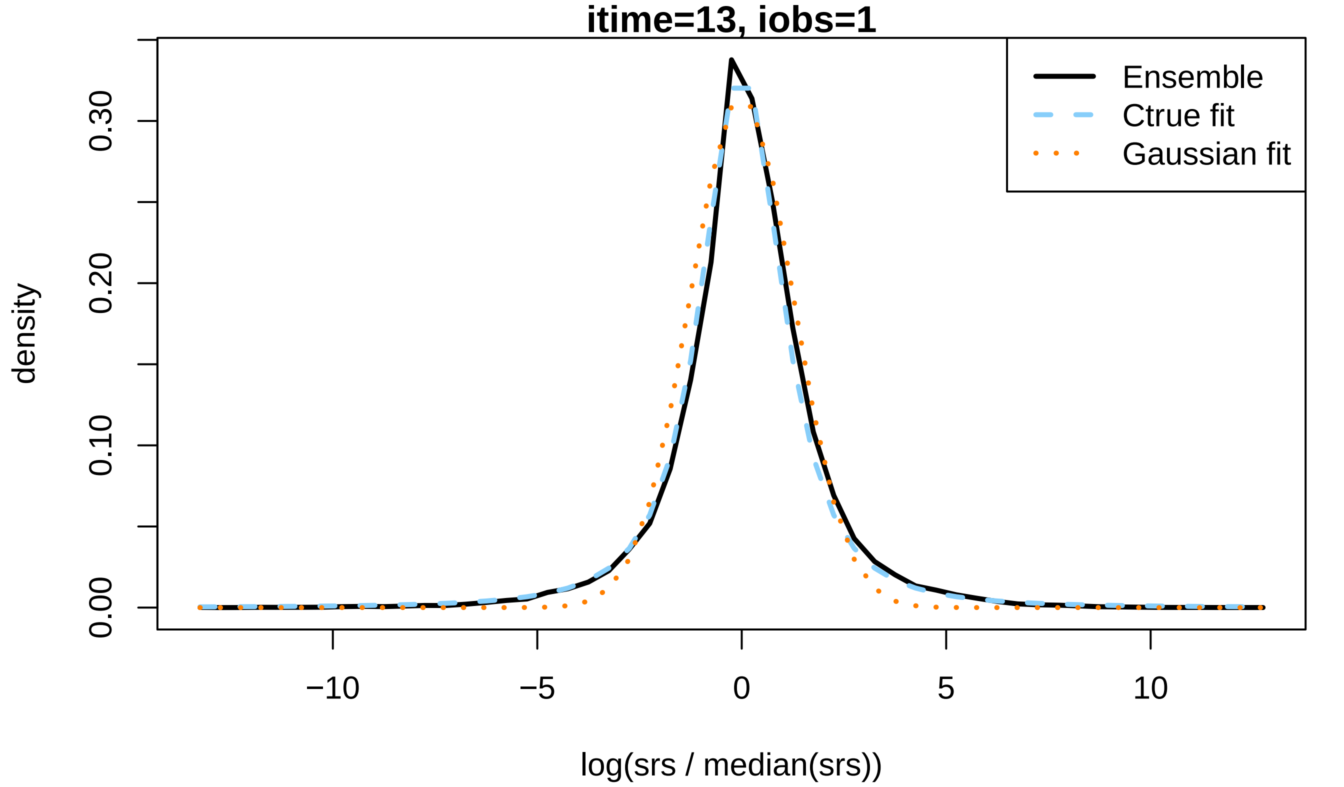

itime=12, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
log(srs / median(srs))

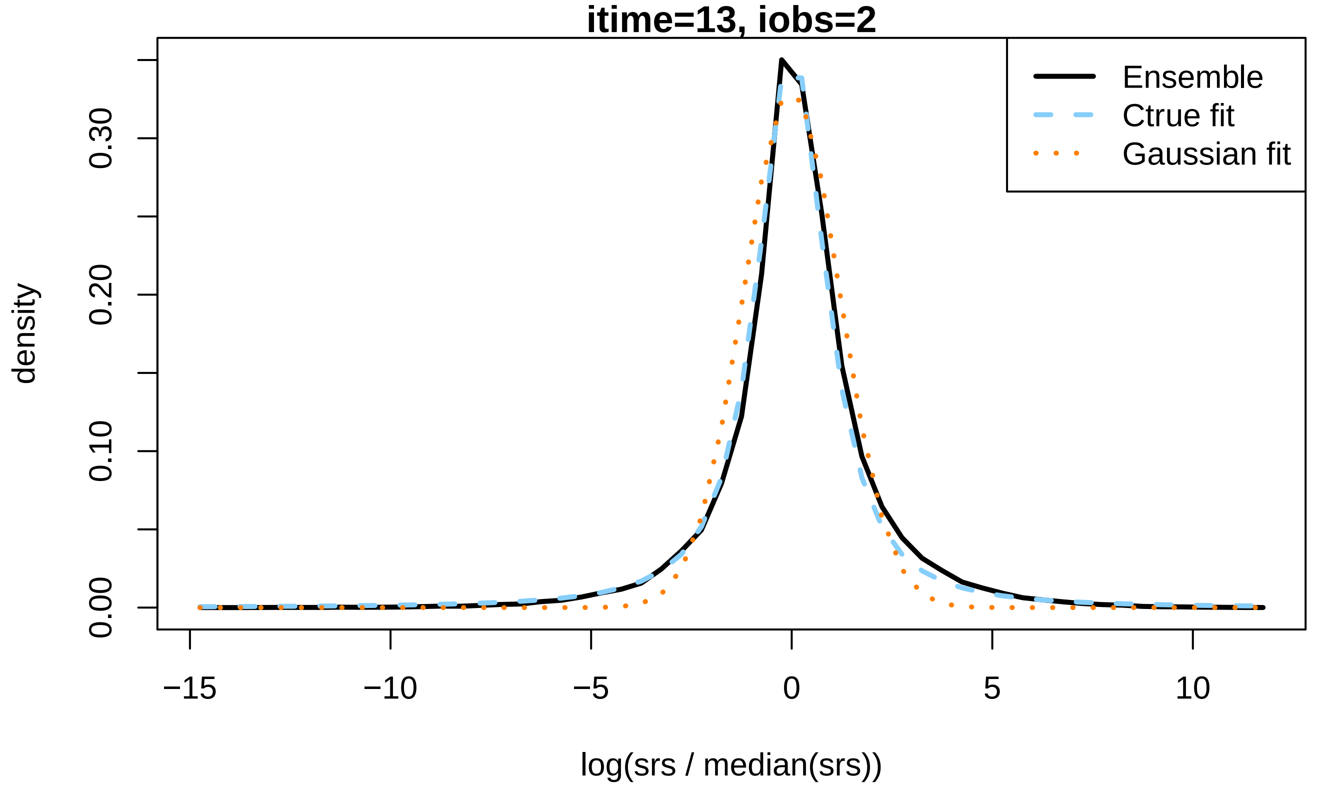


itime=13, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−10
−5
0
5
10
log(srs / median(srs))

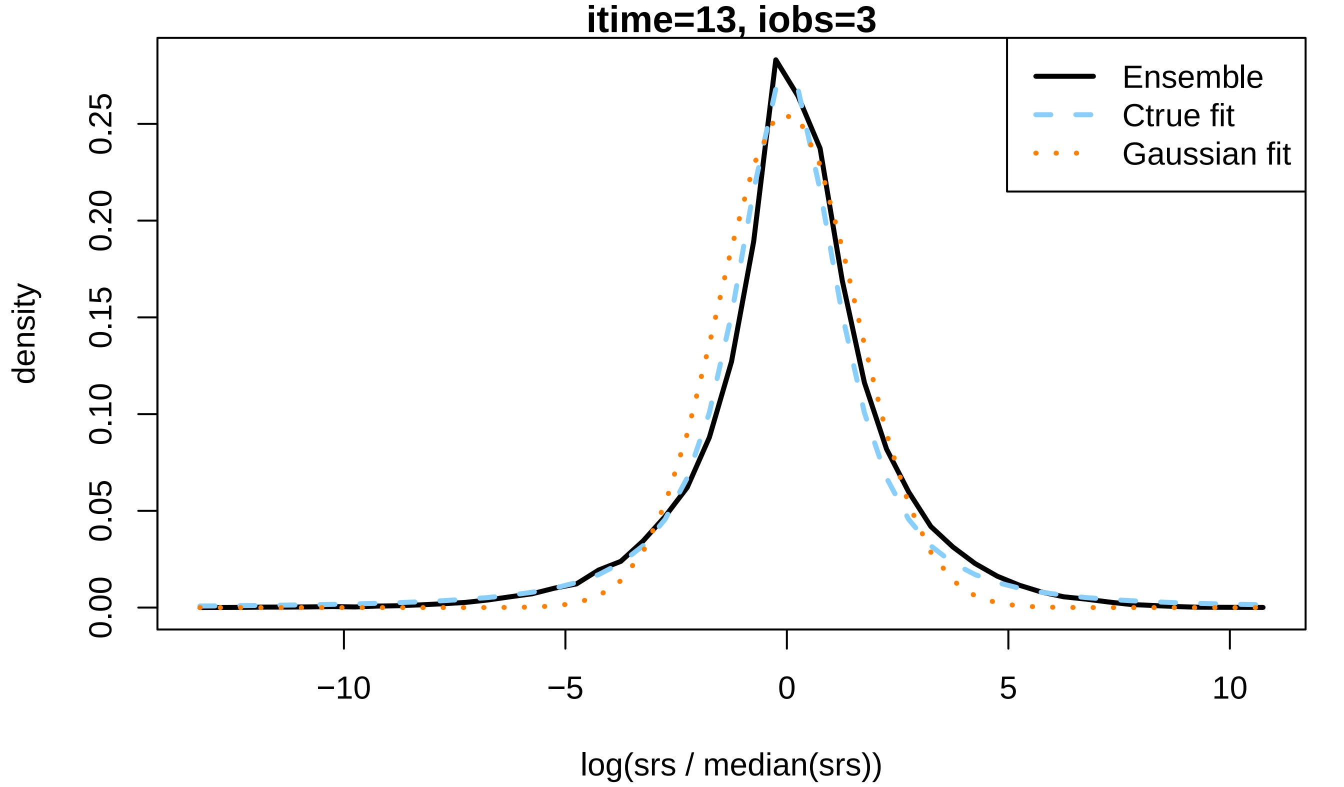

itime=13, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−10
−5
0
5
10
log(srs / median(srs))


itime=13, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

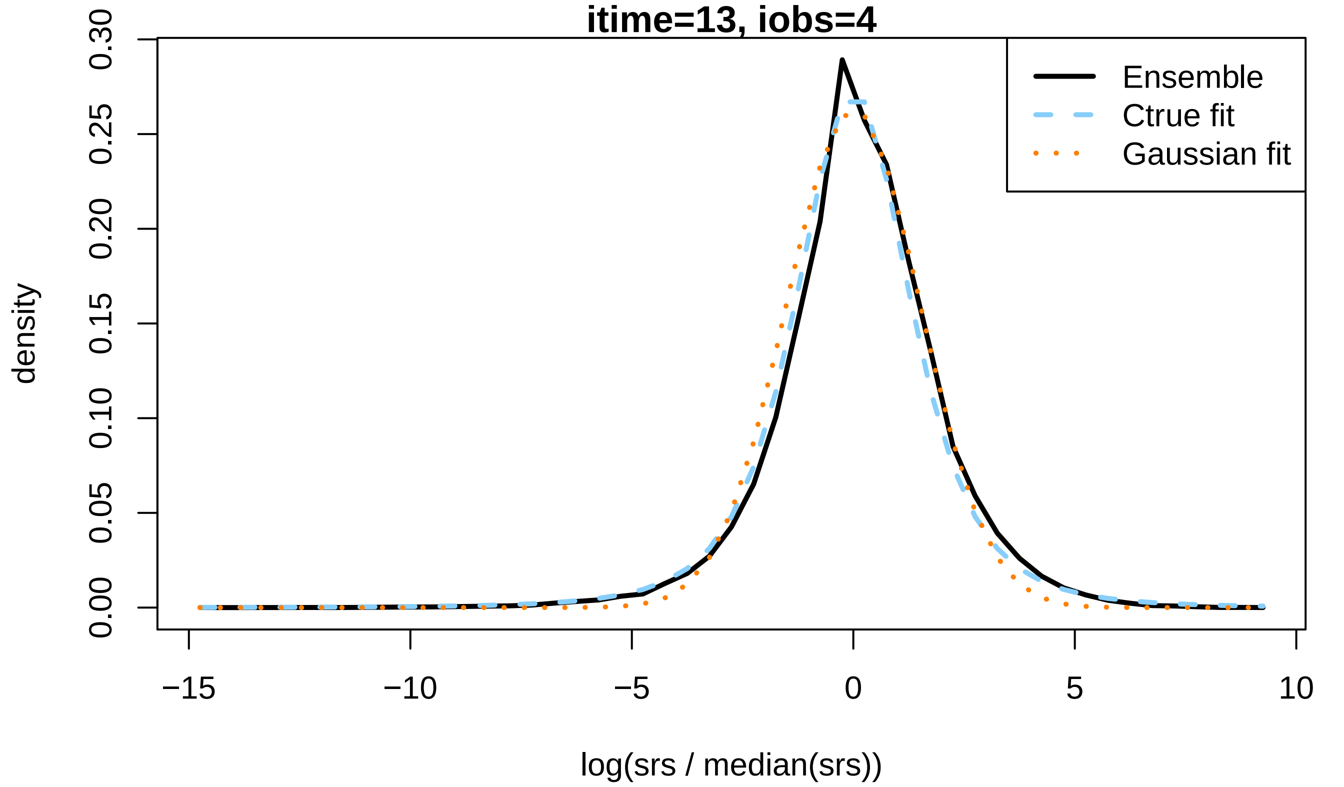

itime=13, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))



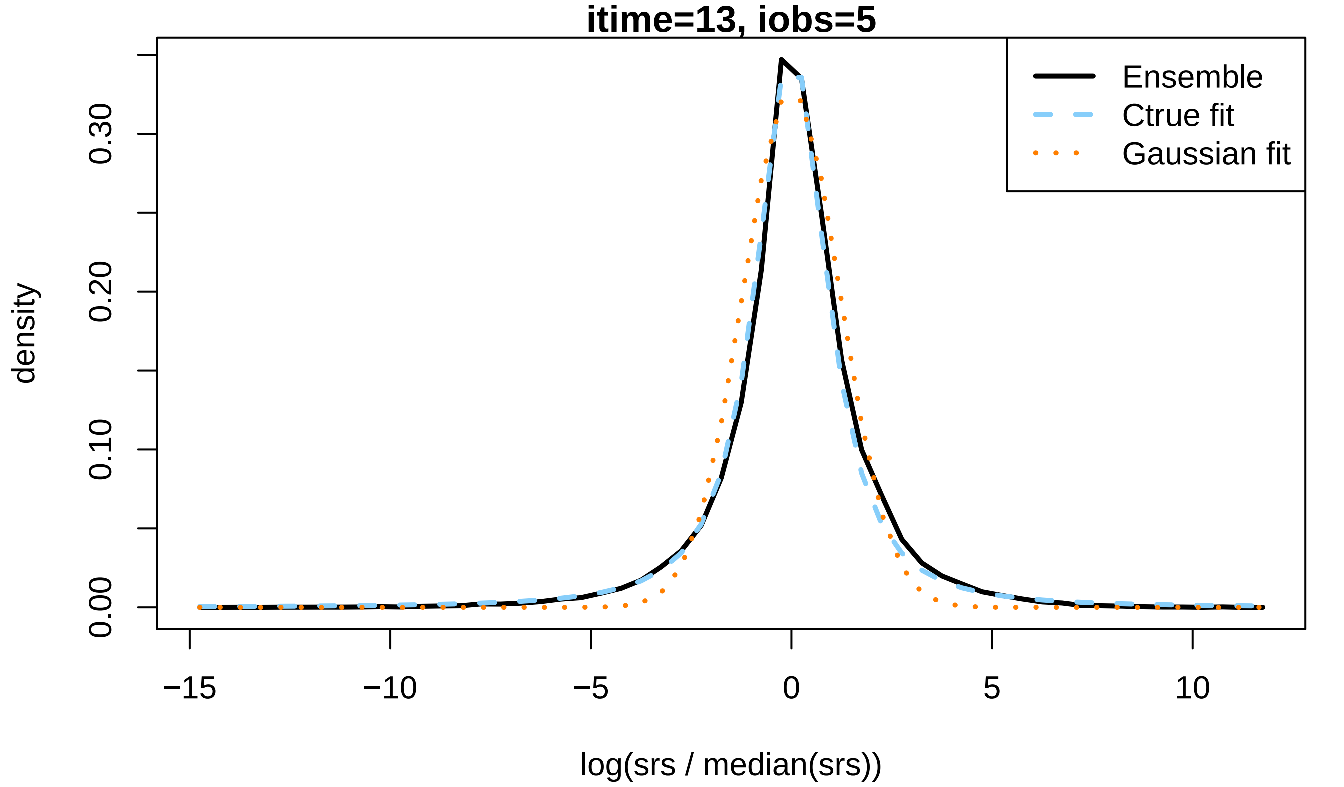
itime=13, iobs=2
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

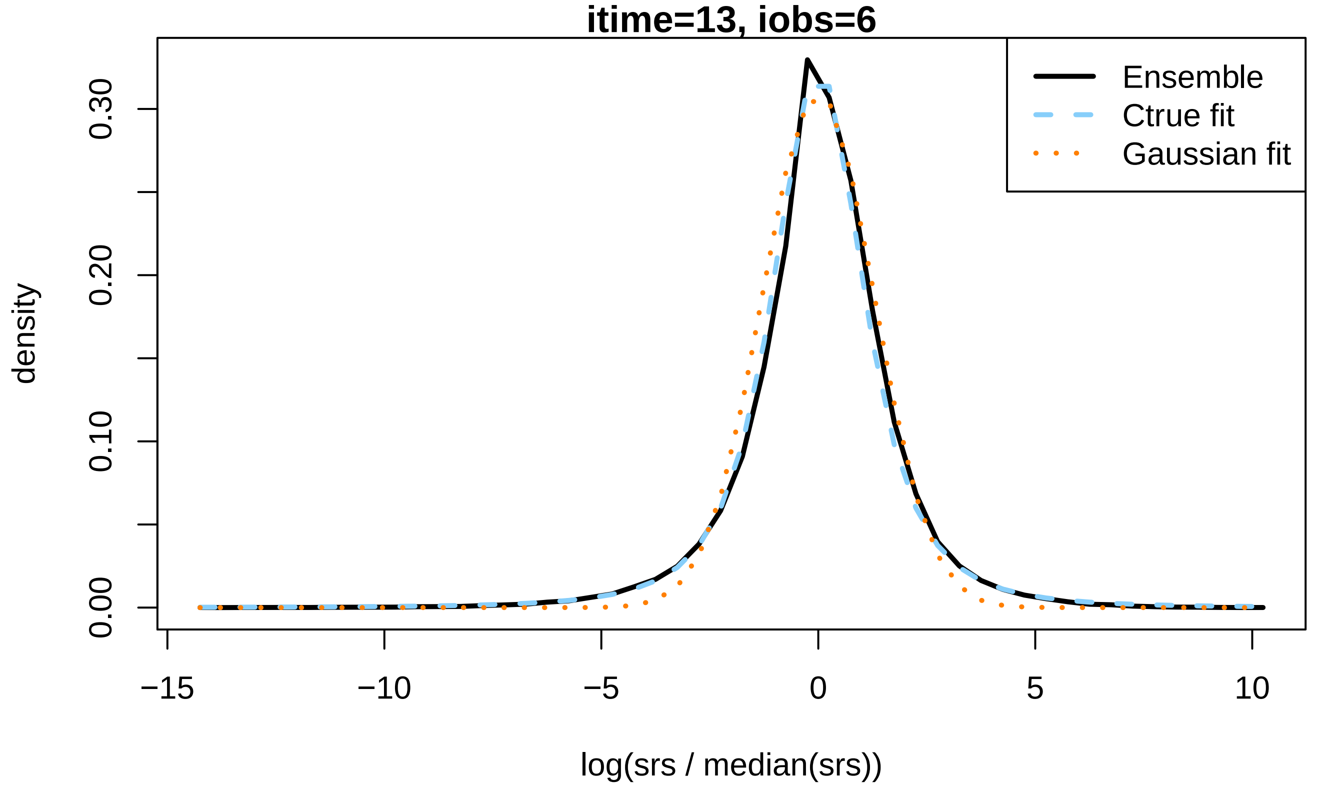


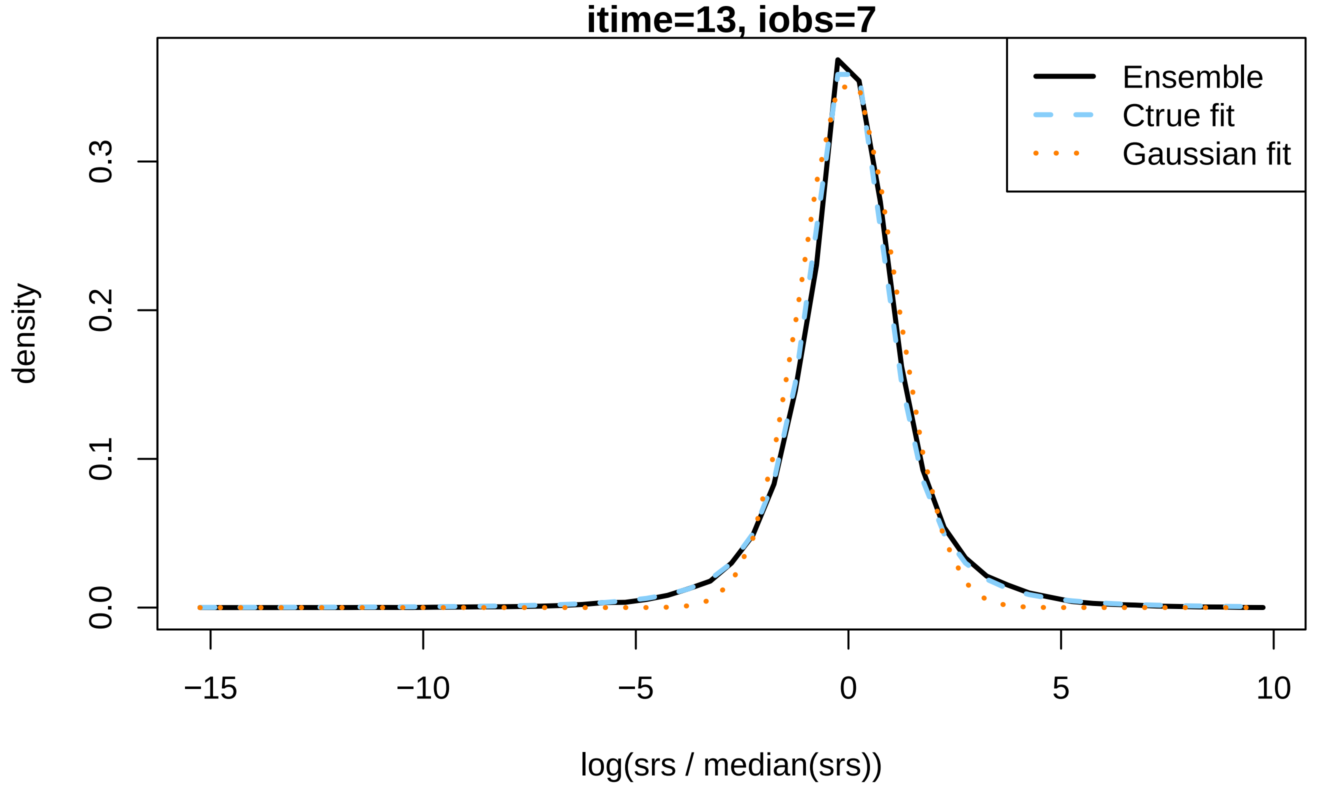
itime=13, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−10
−5
0
5
10
log(srs / median(srs))

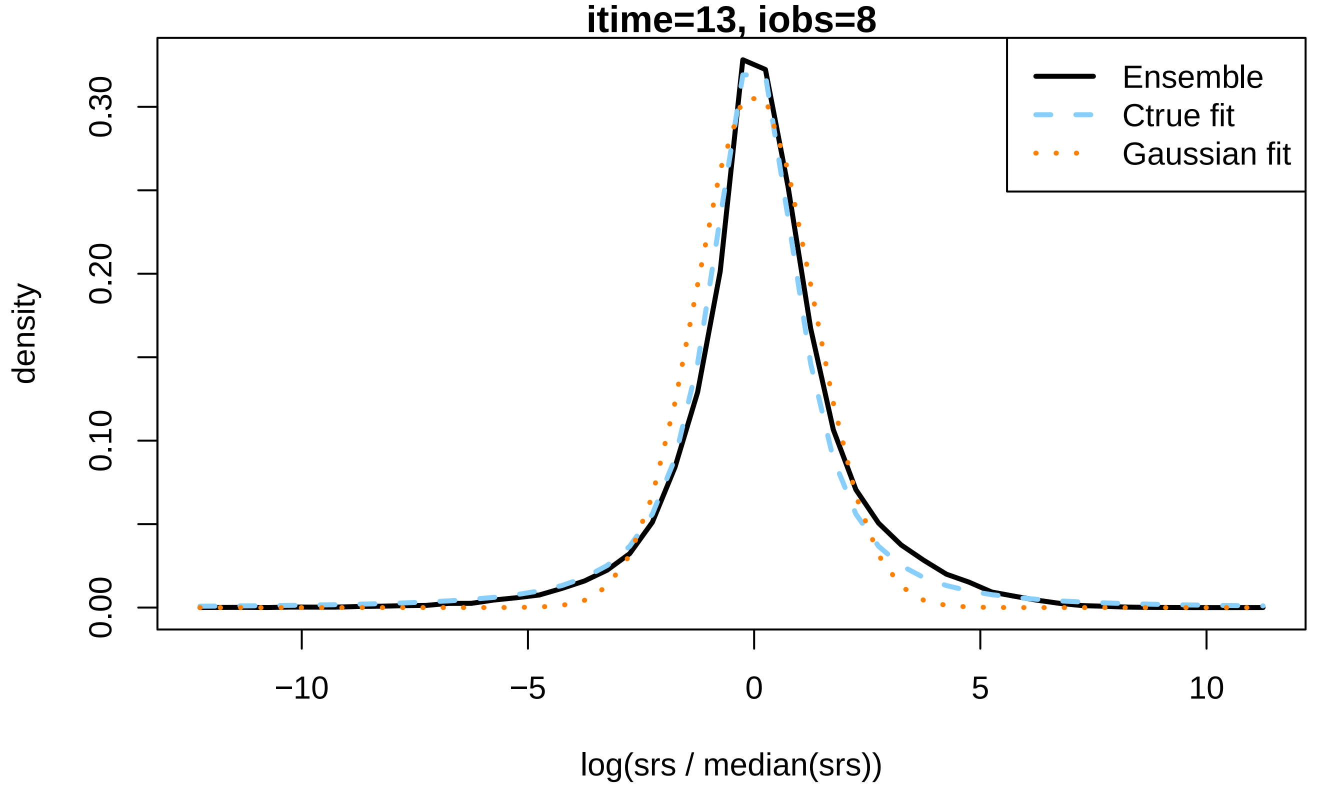


itime=13, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

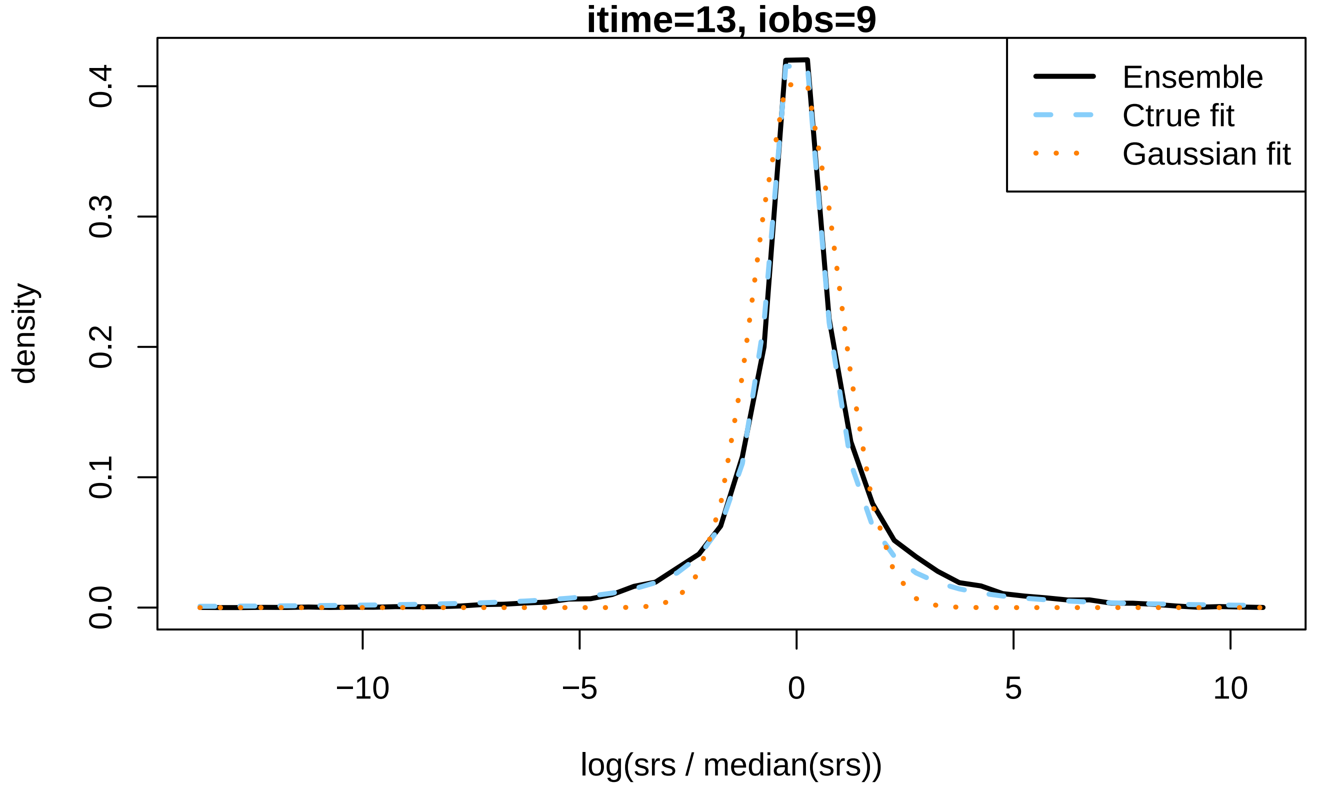

itime=13, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

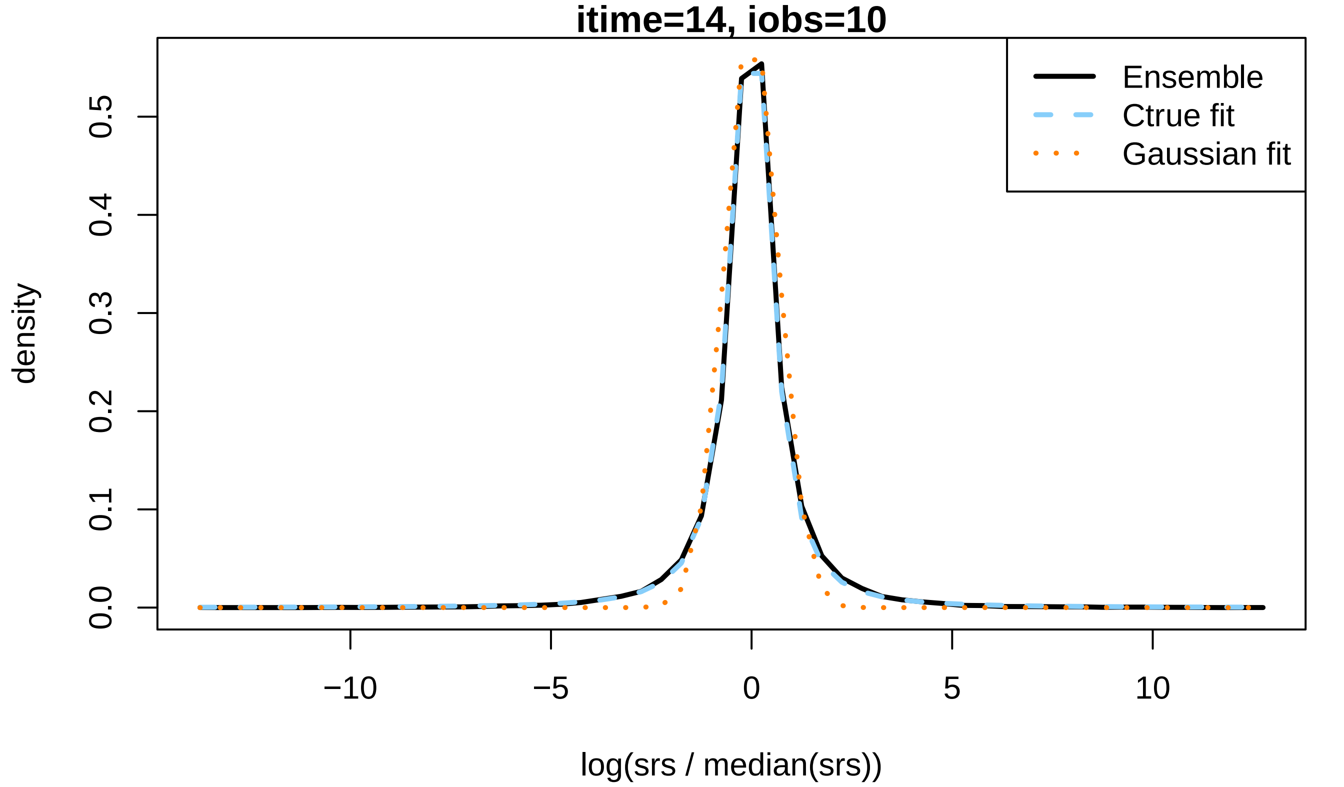

itime=13, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

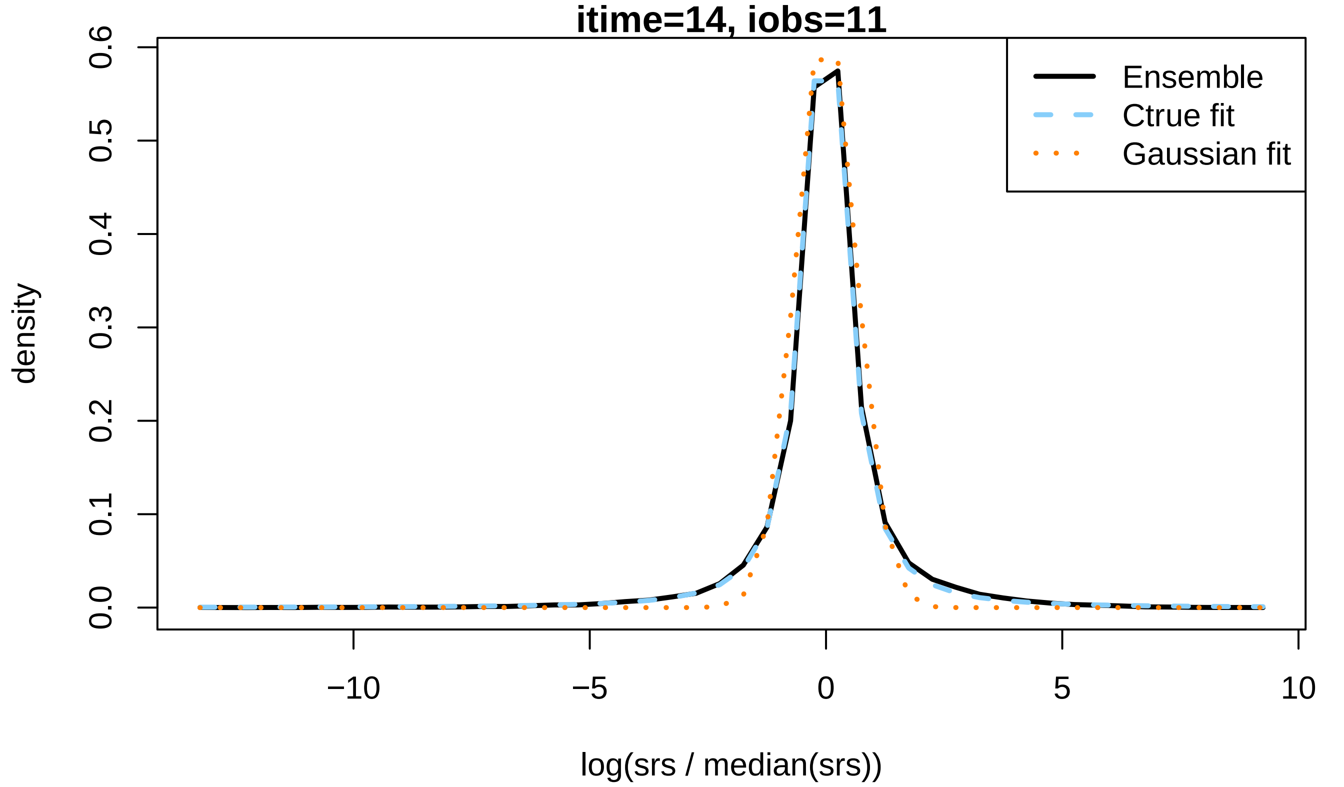

itime=13, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

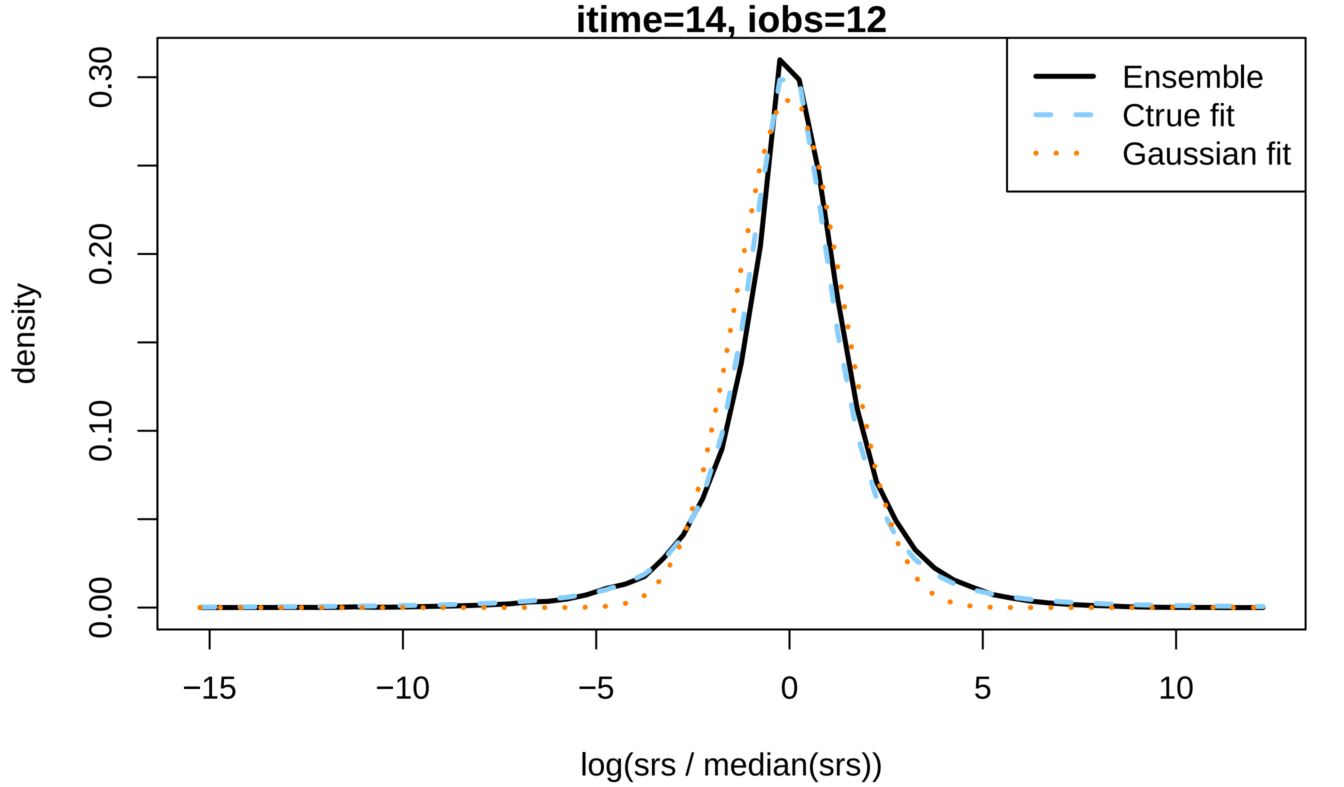


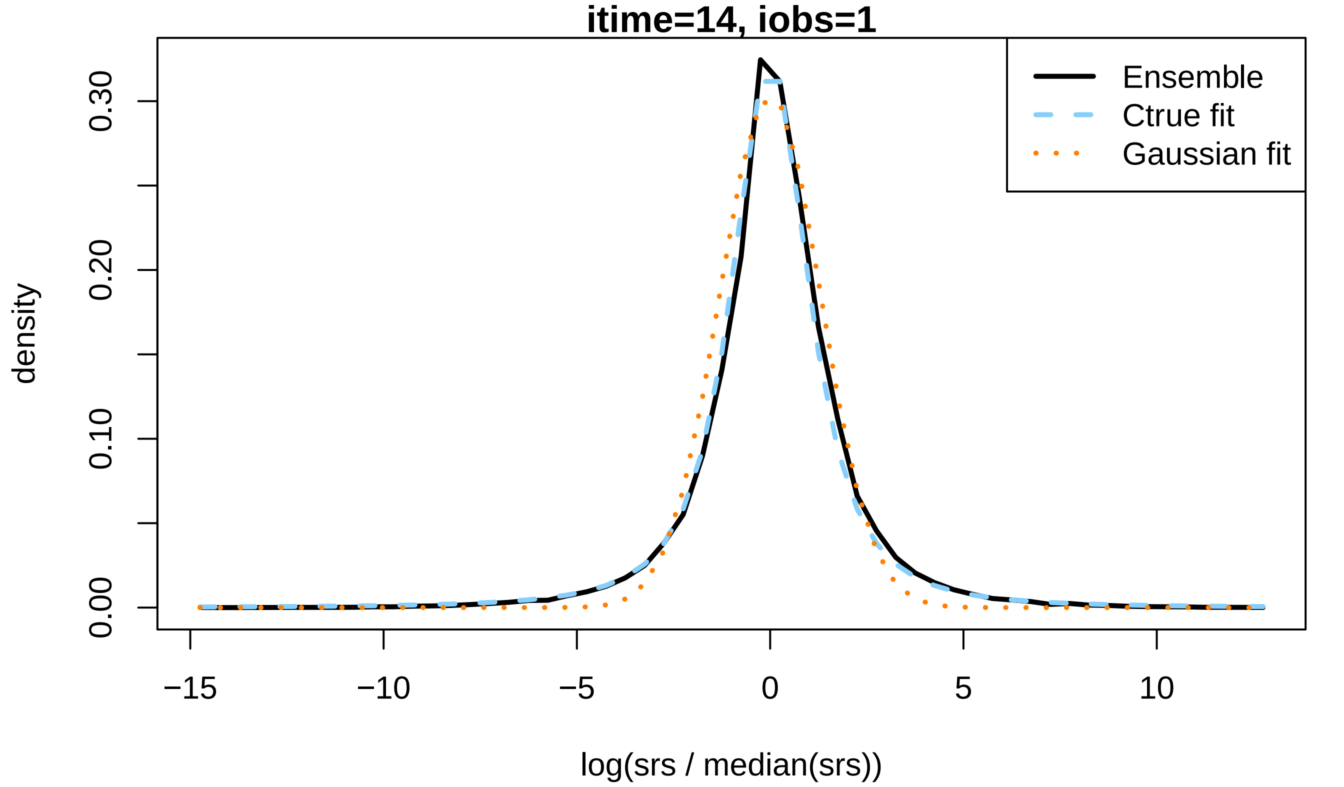
itime=13, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))

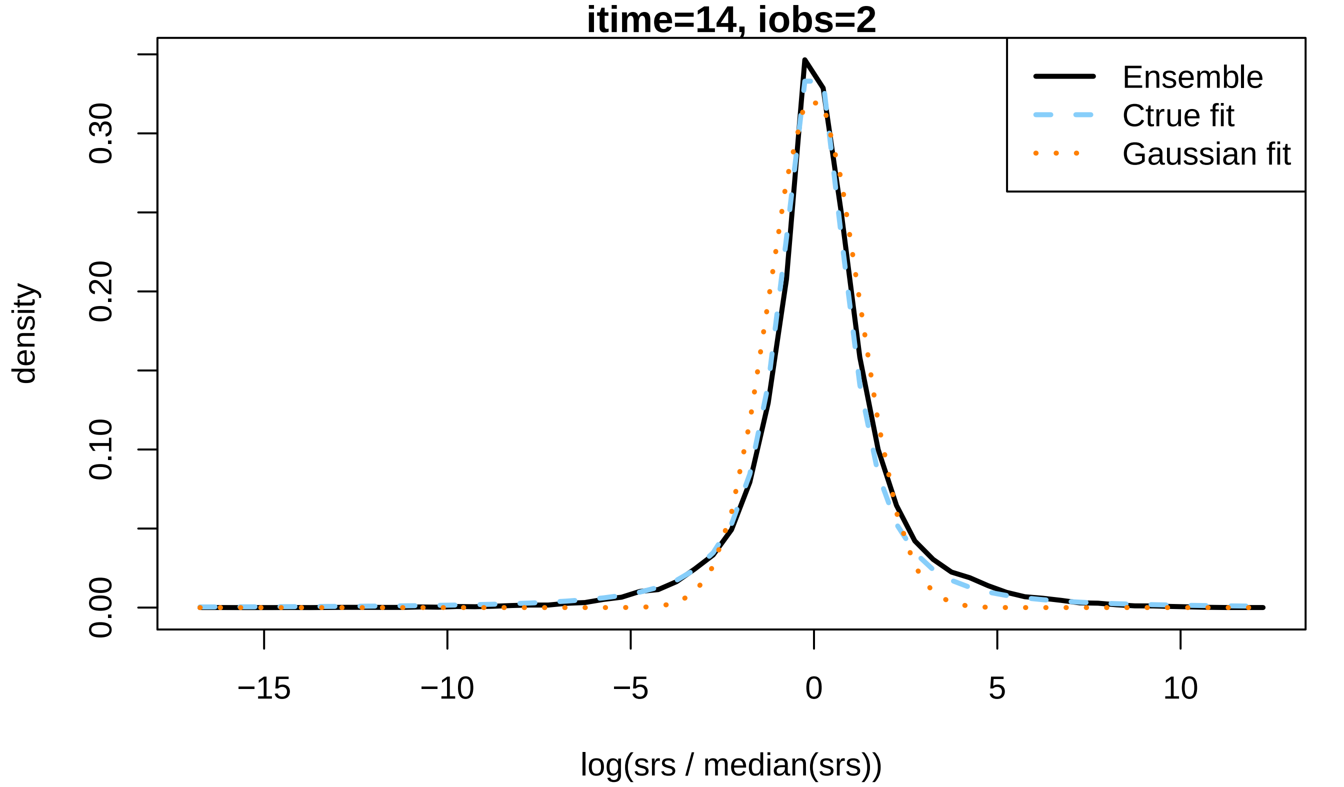

itime=13, iobs=9
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit


itime=14, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−10
−5
0
5
10
log(srs / median(srs))

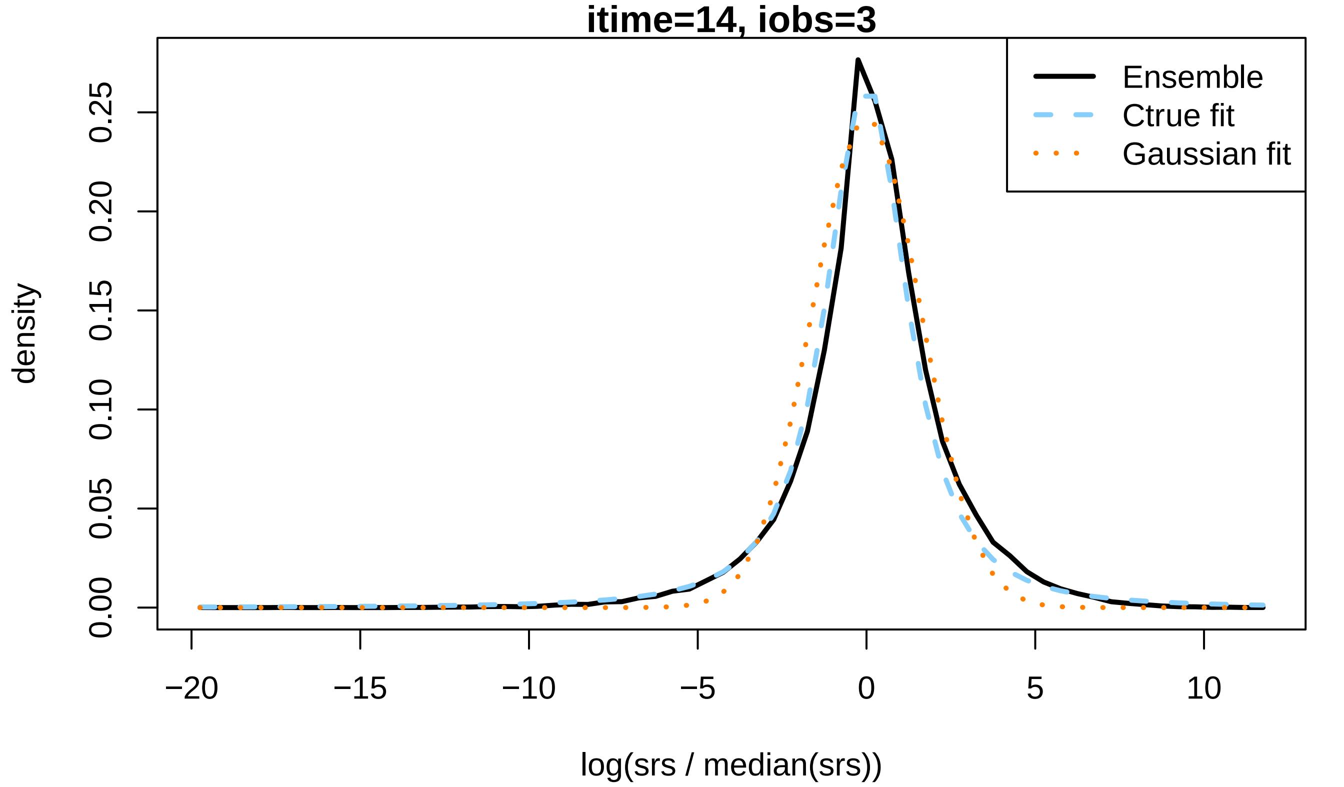

itime=14, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−10
−5
0
5
10
log(srs / median(srs))

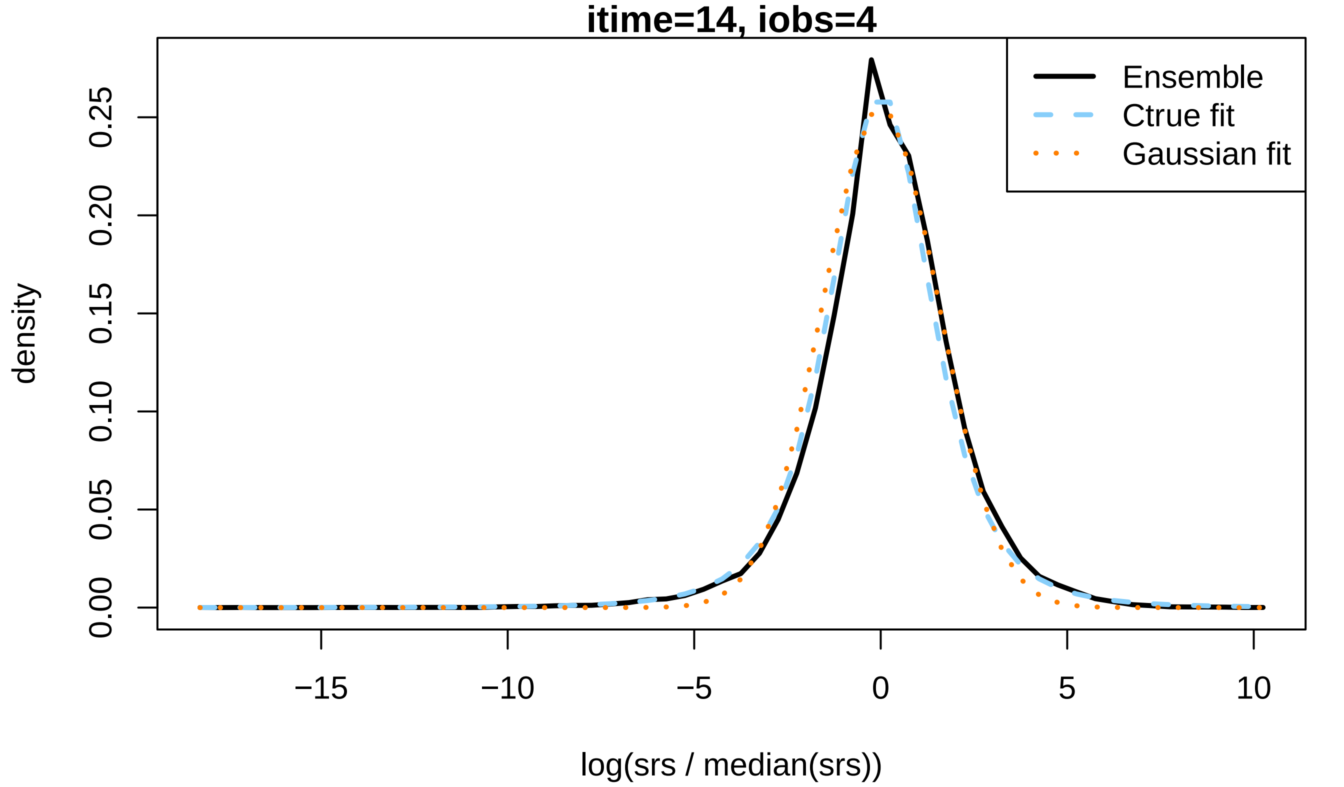

itime=14, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

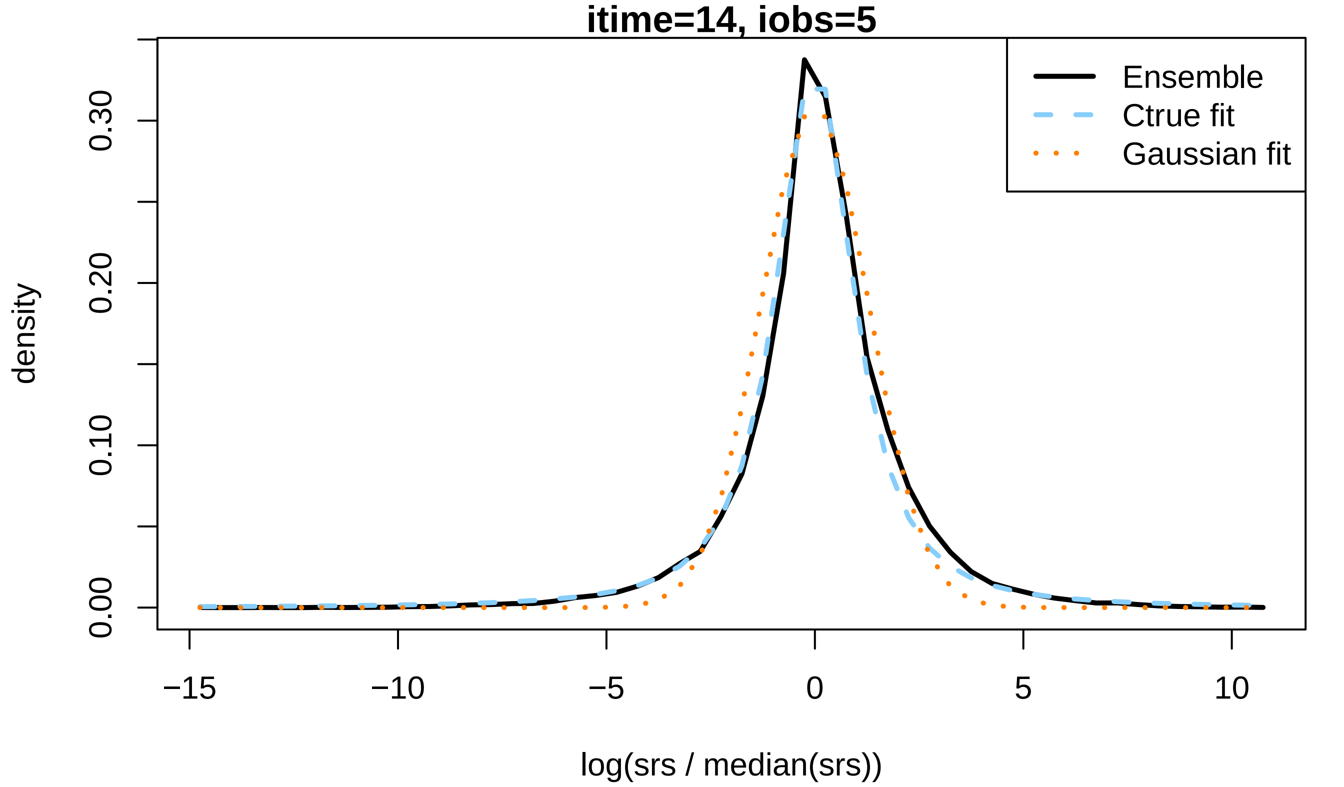

itime=14, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



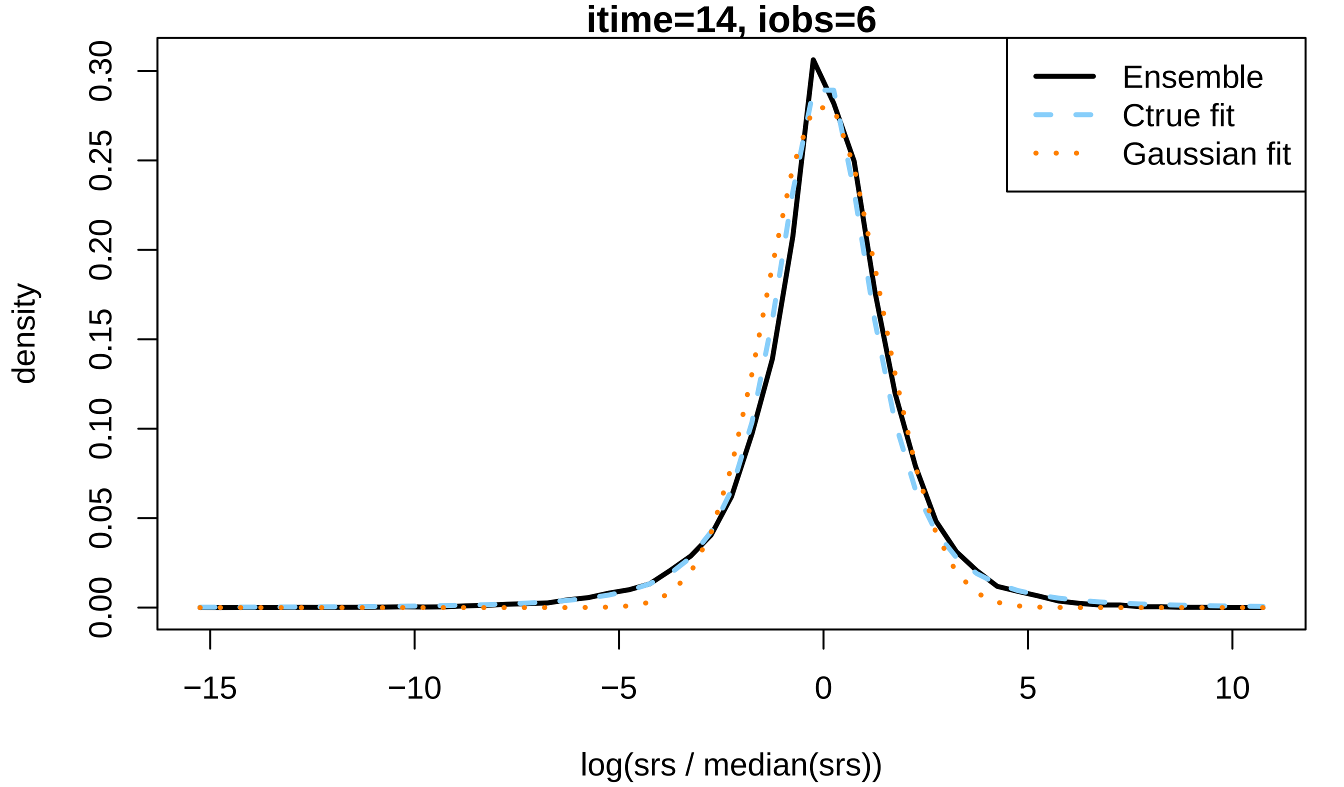
itime=14, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=14, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−20
−15
−10
−5
0
5
10
log(srs / median(srs))

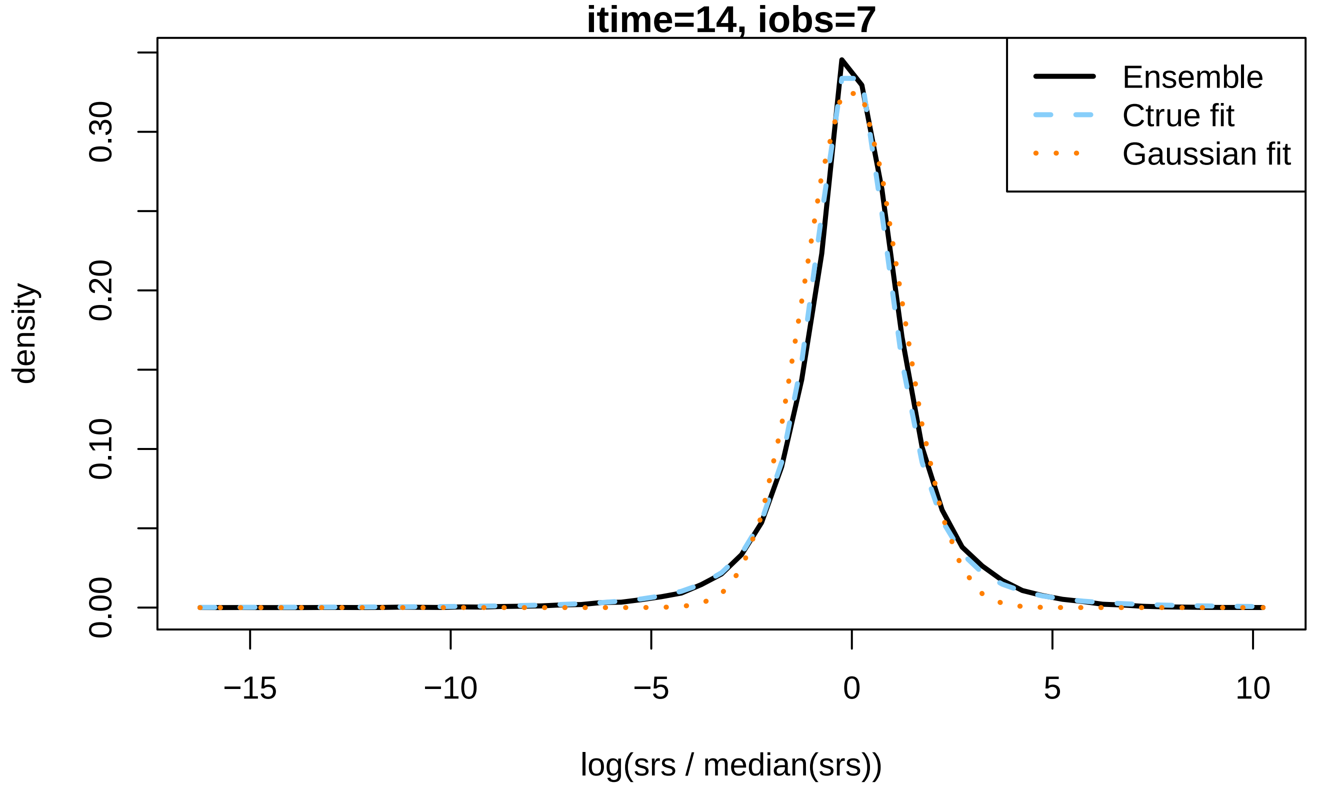

itime=14, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))



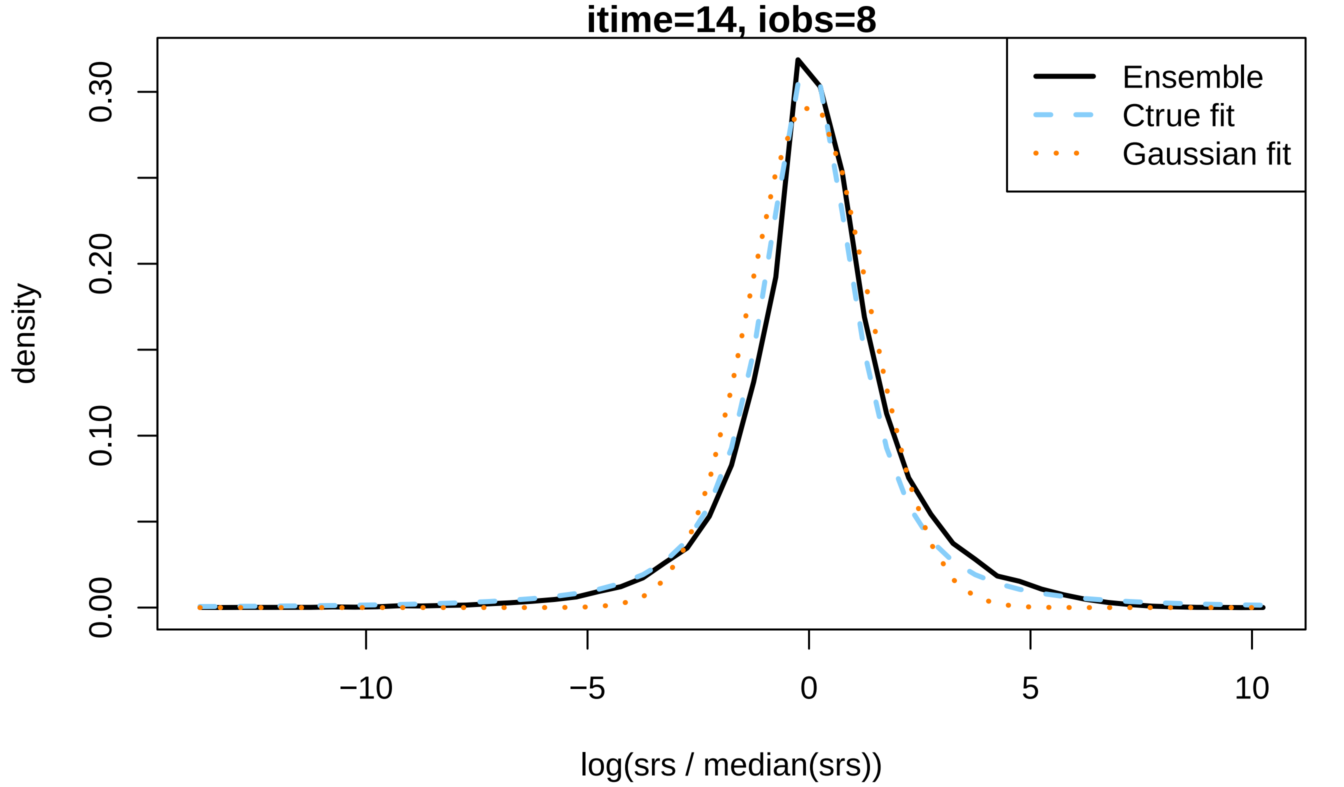
itime=14, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

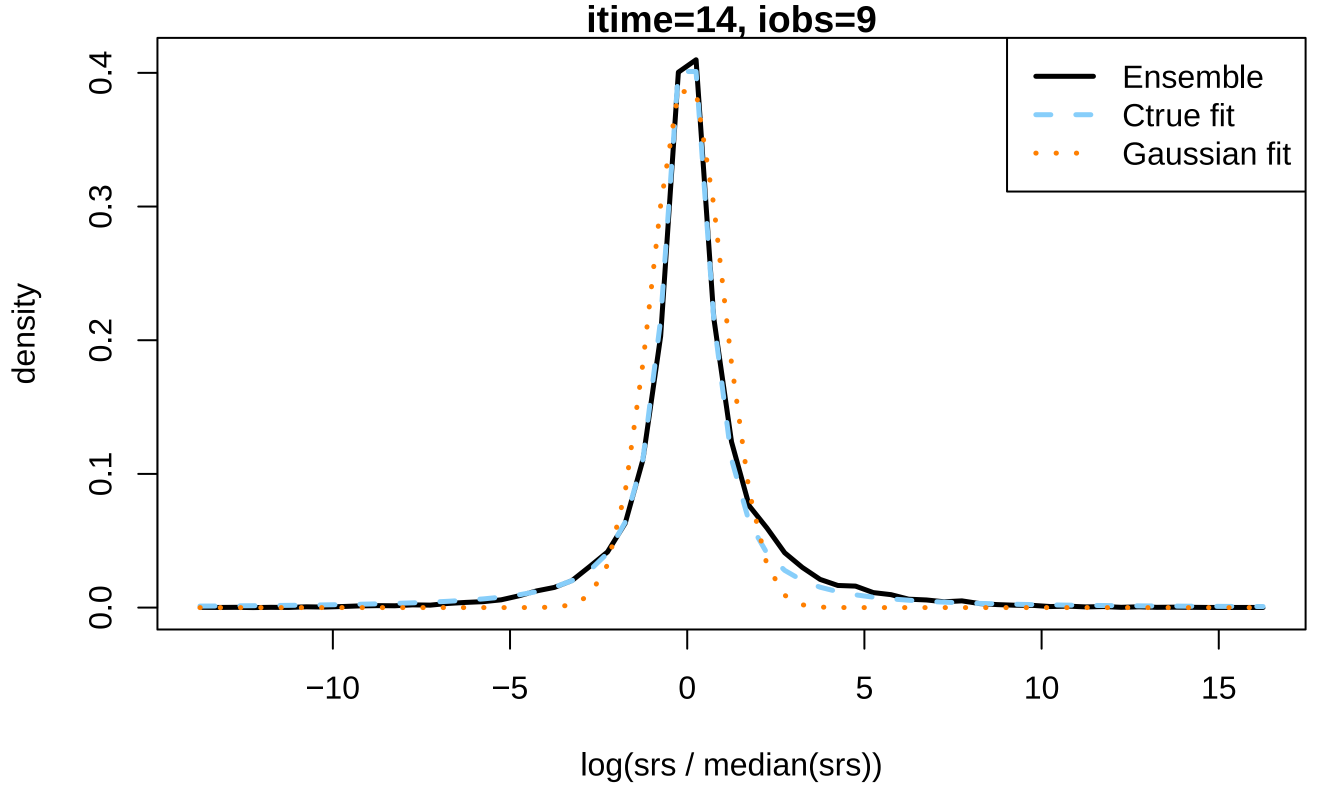


itime=14, iobs=6
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

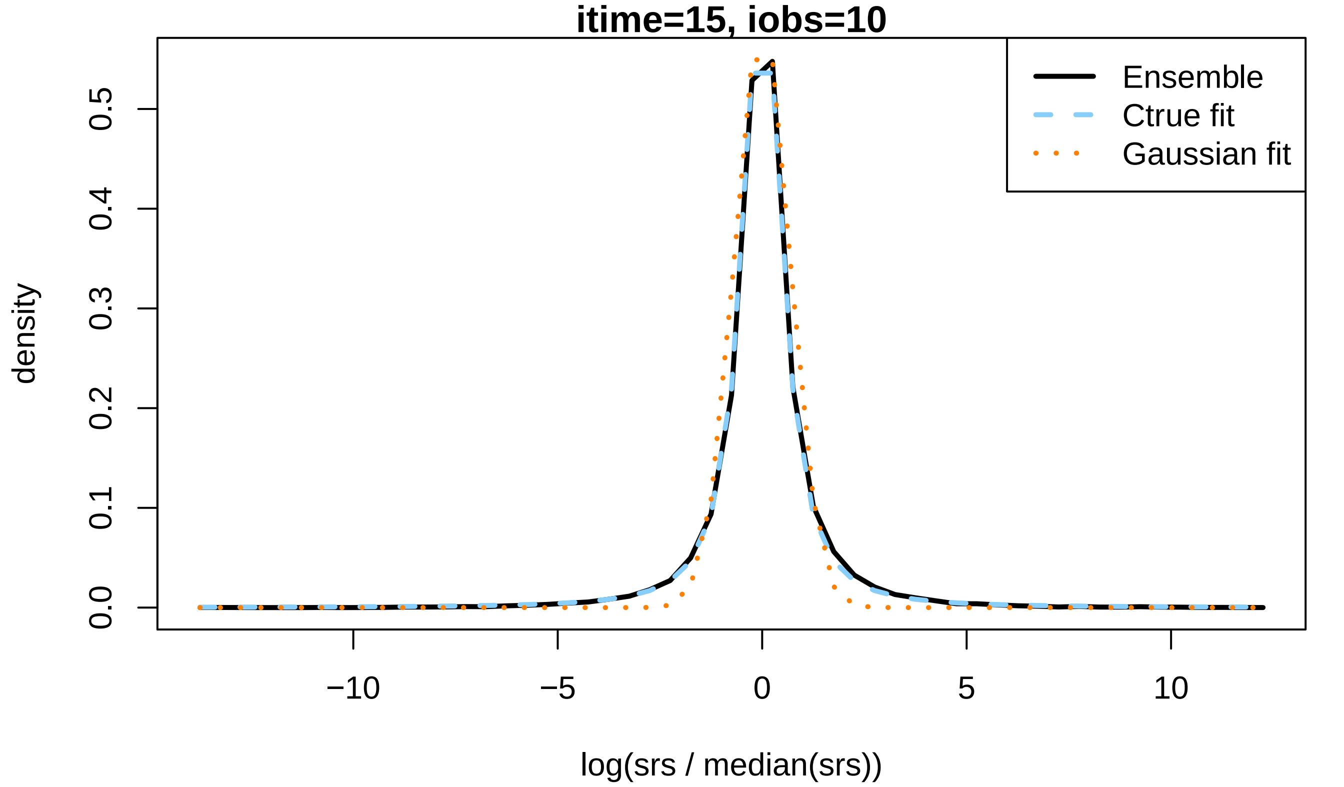


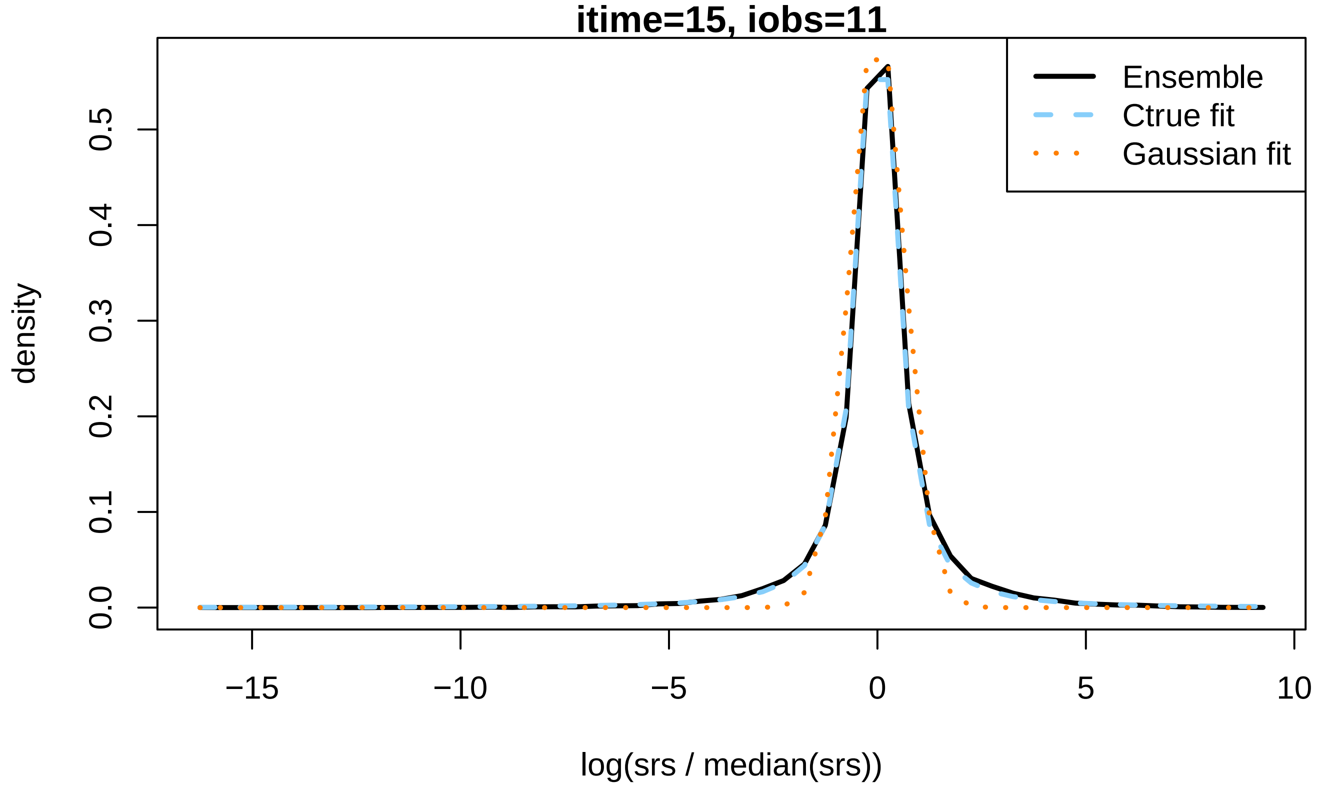
itime=14, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=14, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))

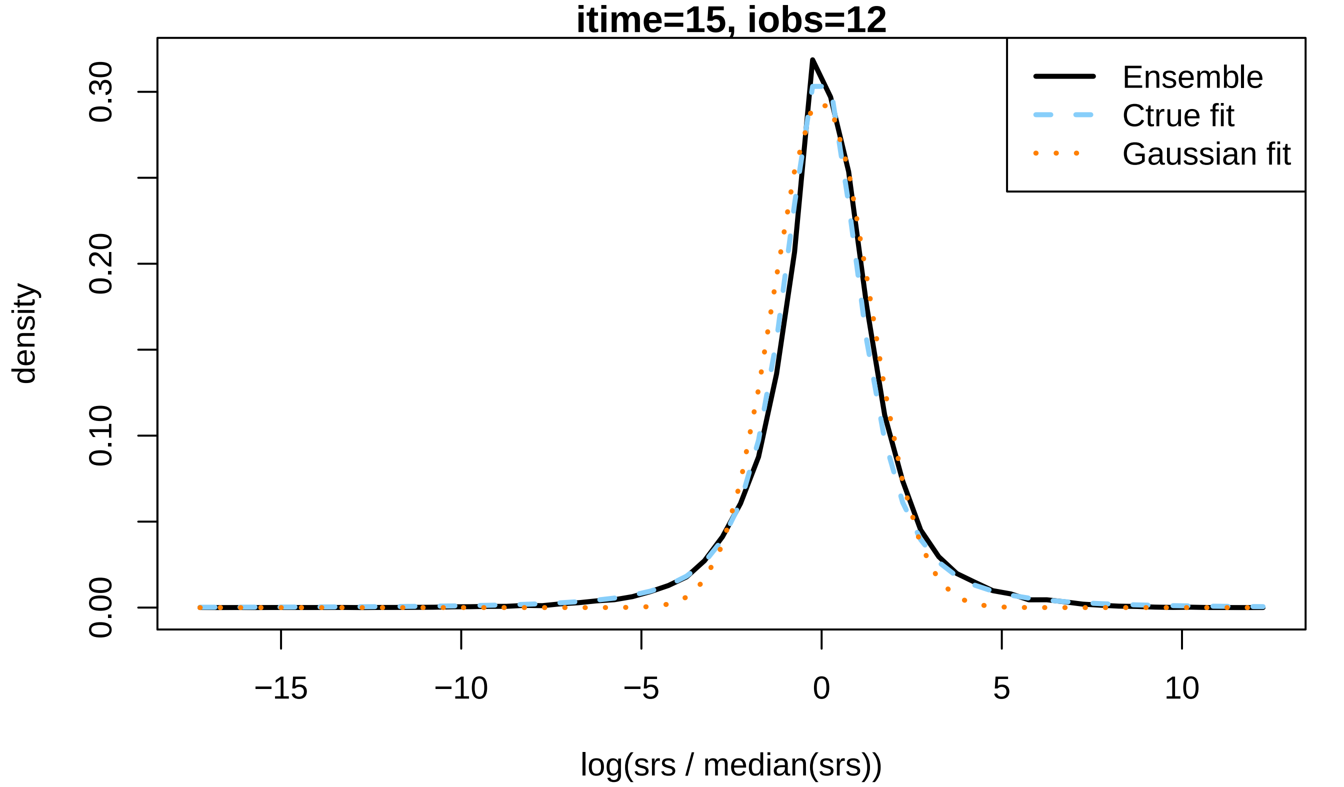


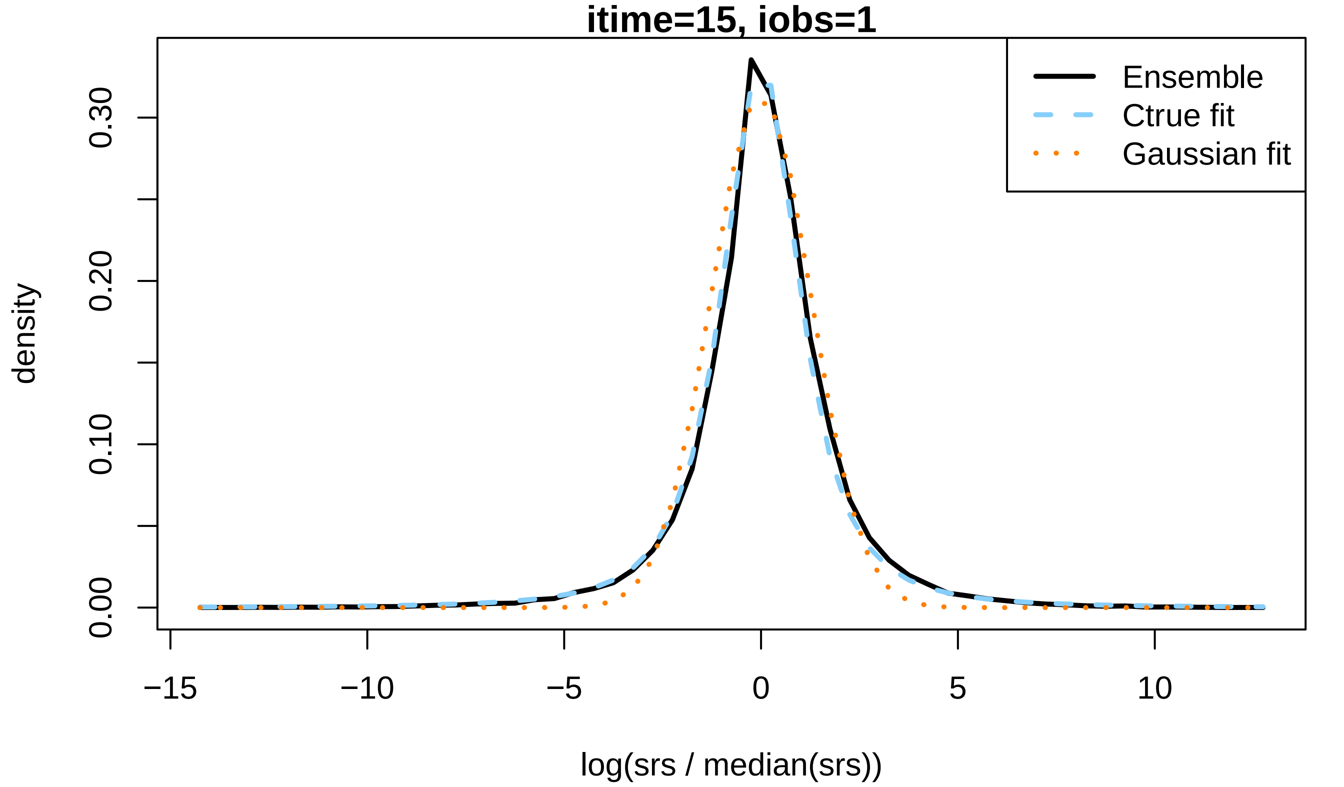
itime=14, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
15
log(srs / median(srs))

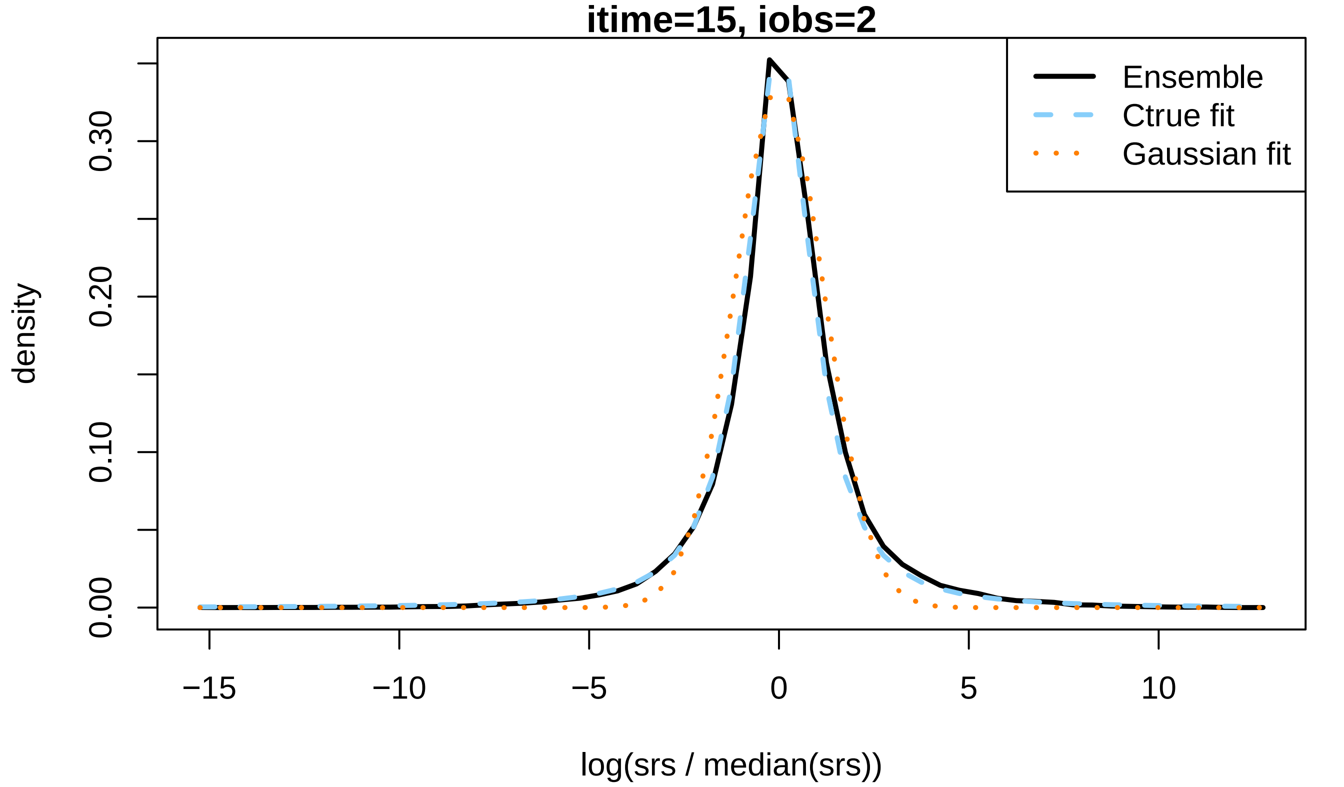

itime=15, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−10
−5
0
5
10
log(srs / median(srs))



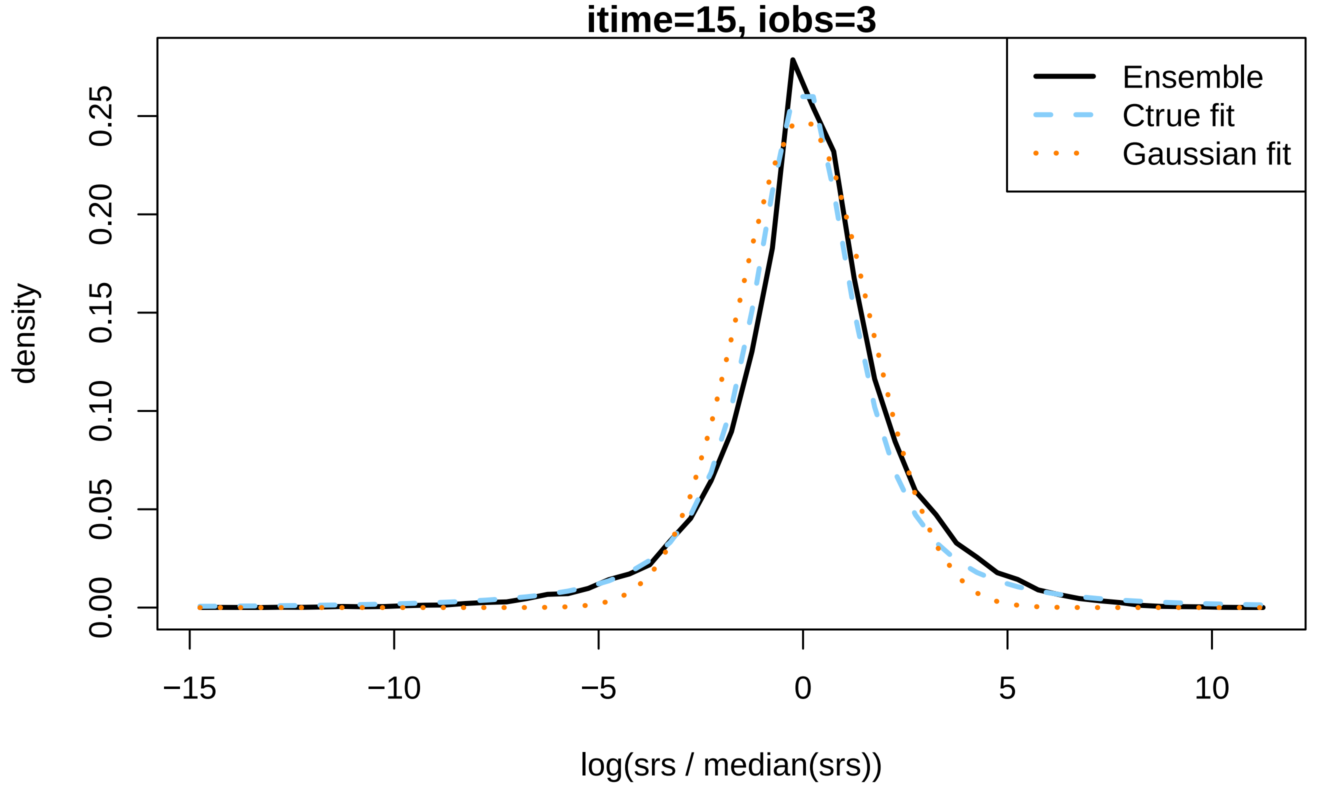
itime=15, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−15
−10
−5
0
5
10
log(srs / median(srs))

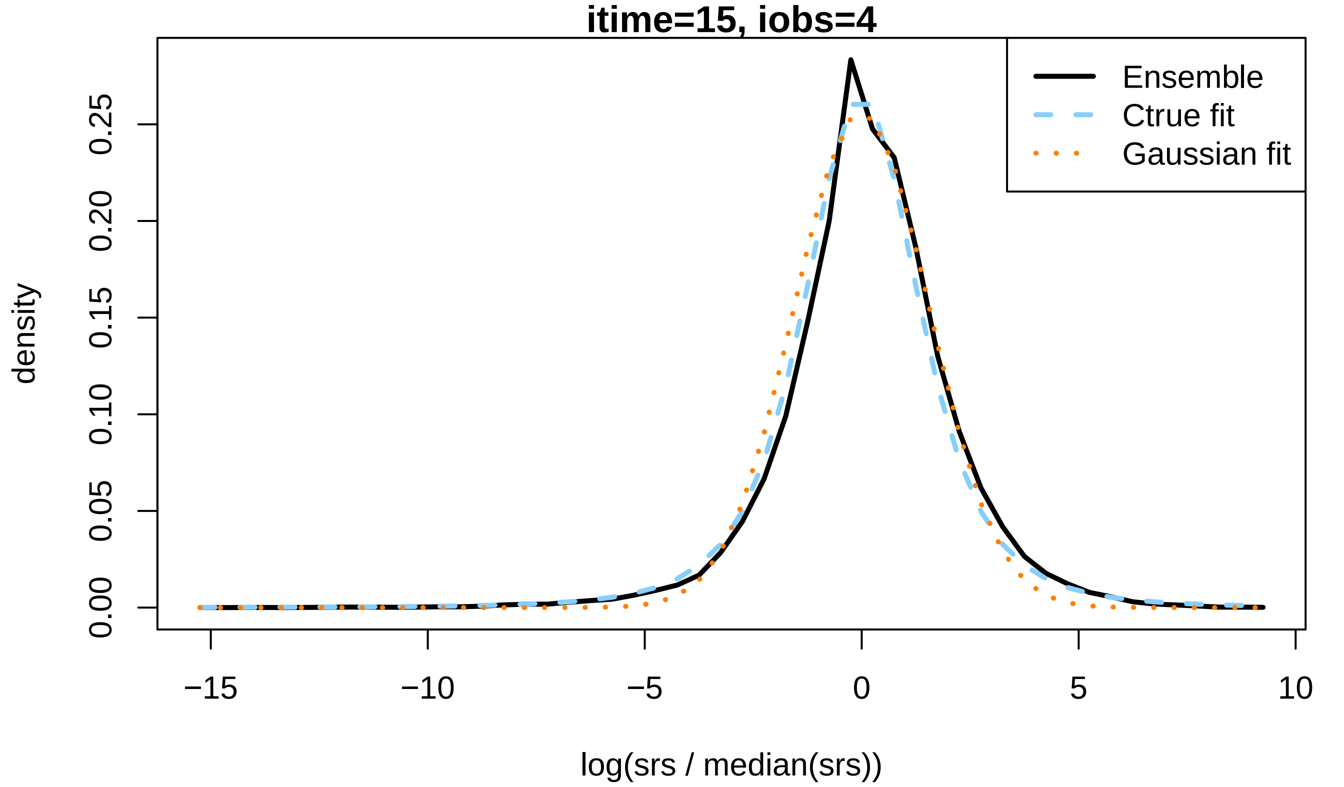

itime=15, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=15, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

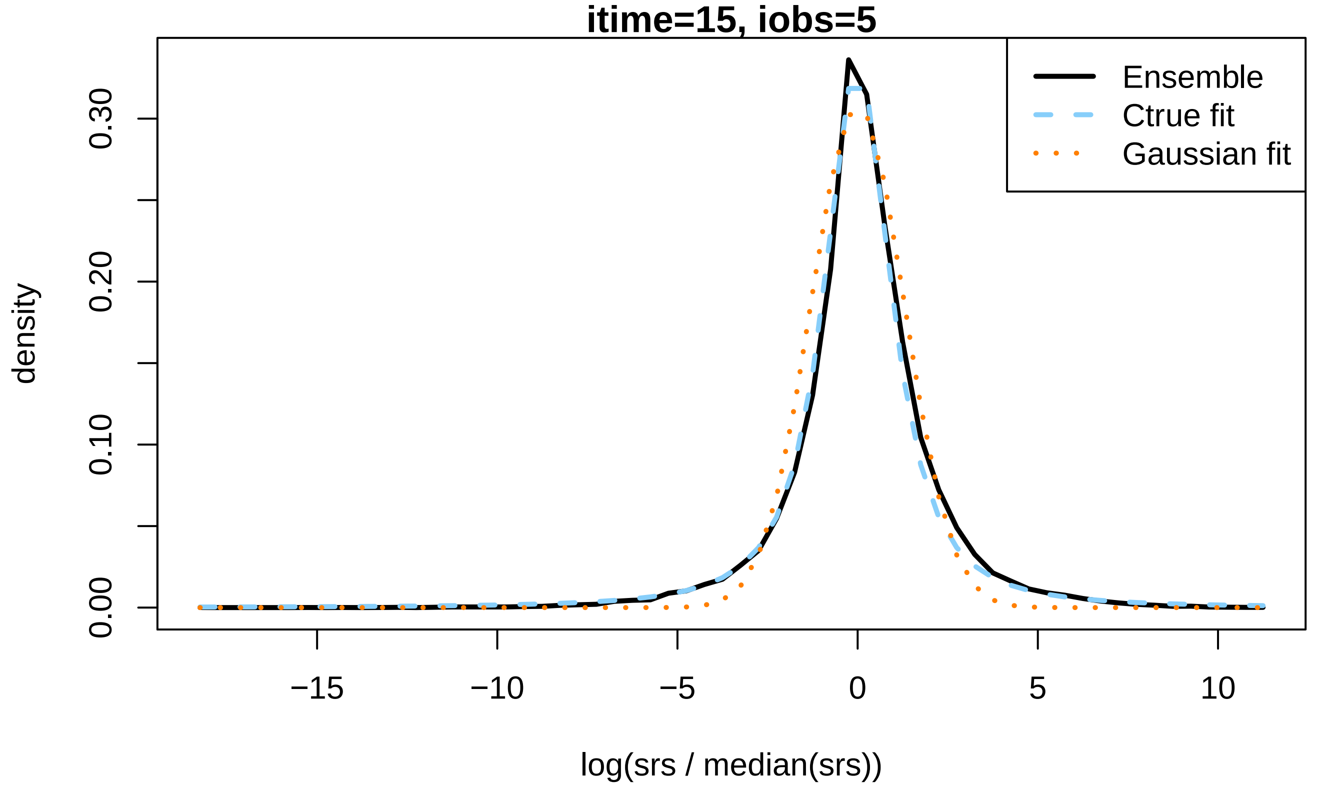


itime=15, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

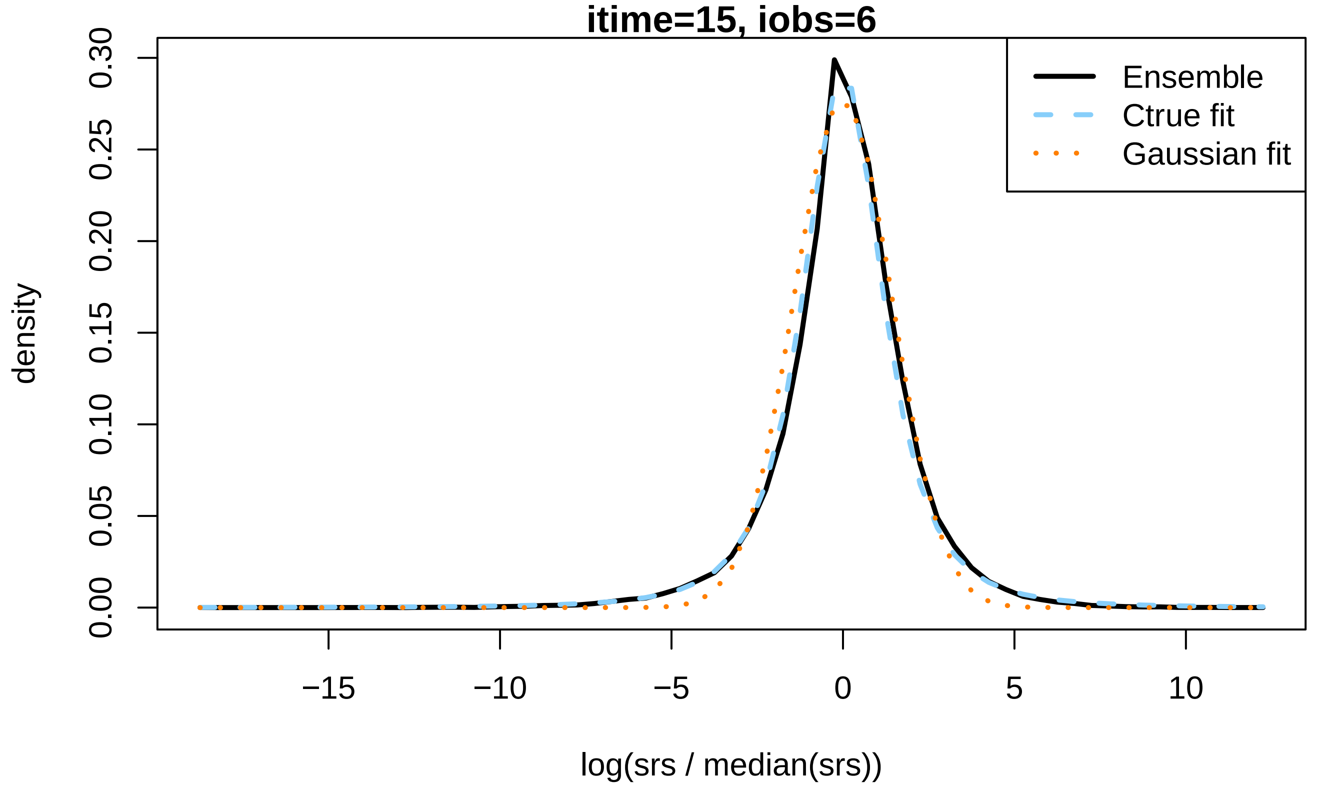

itime=15, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

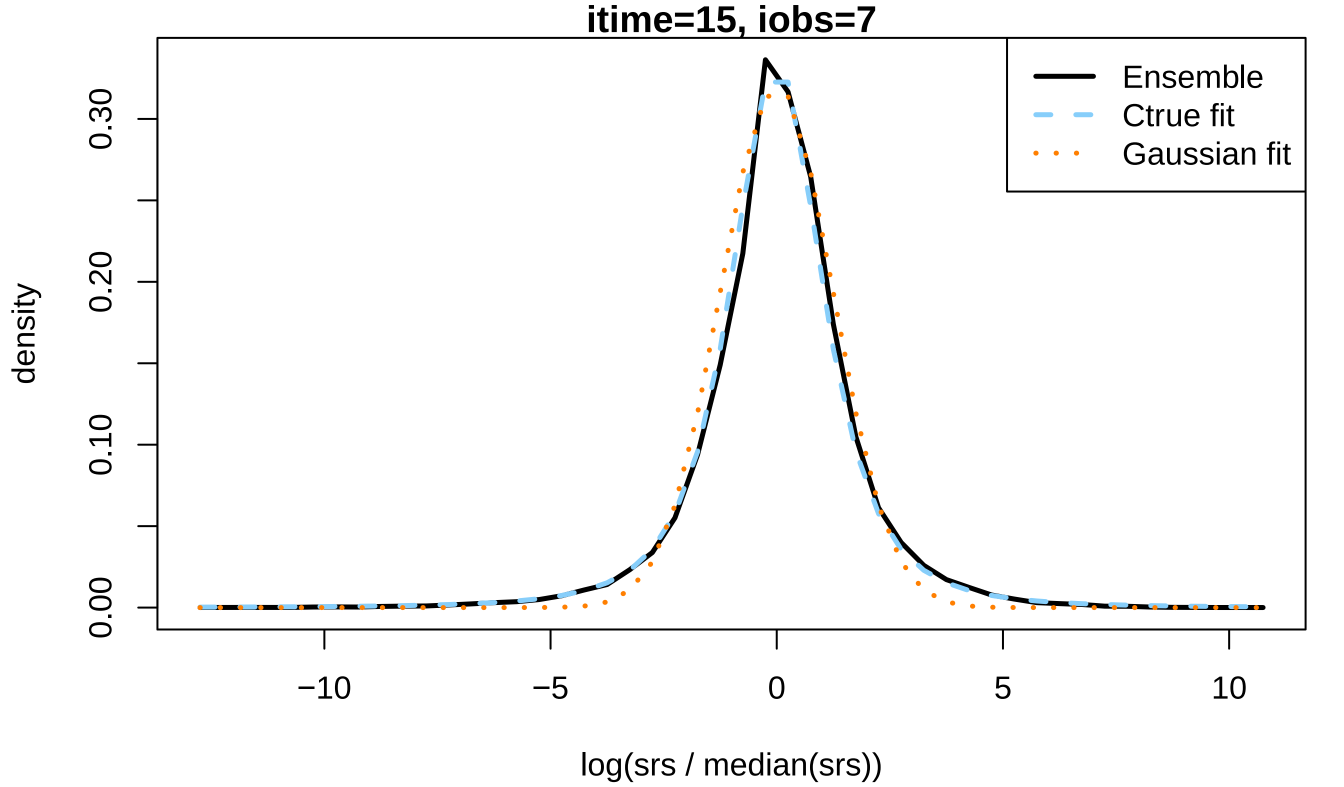


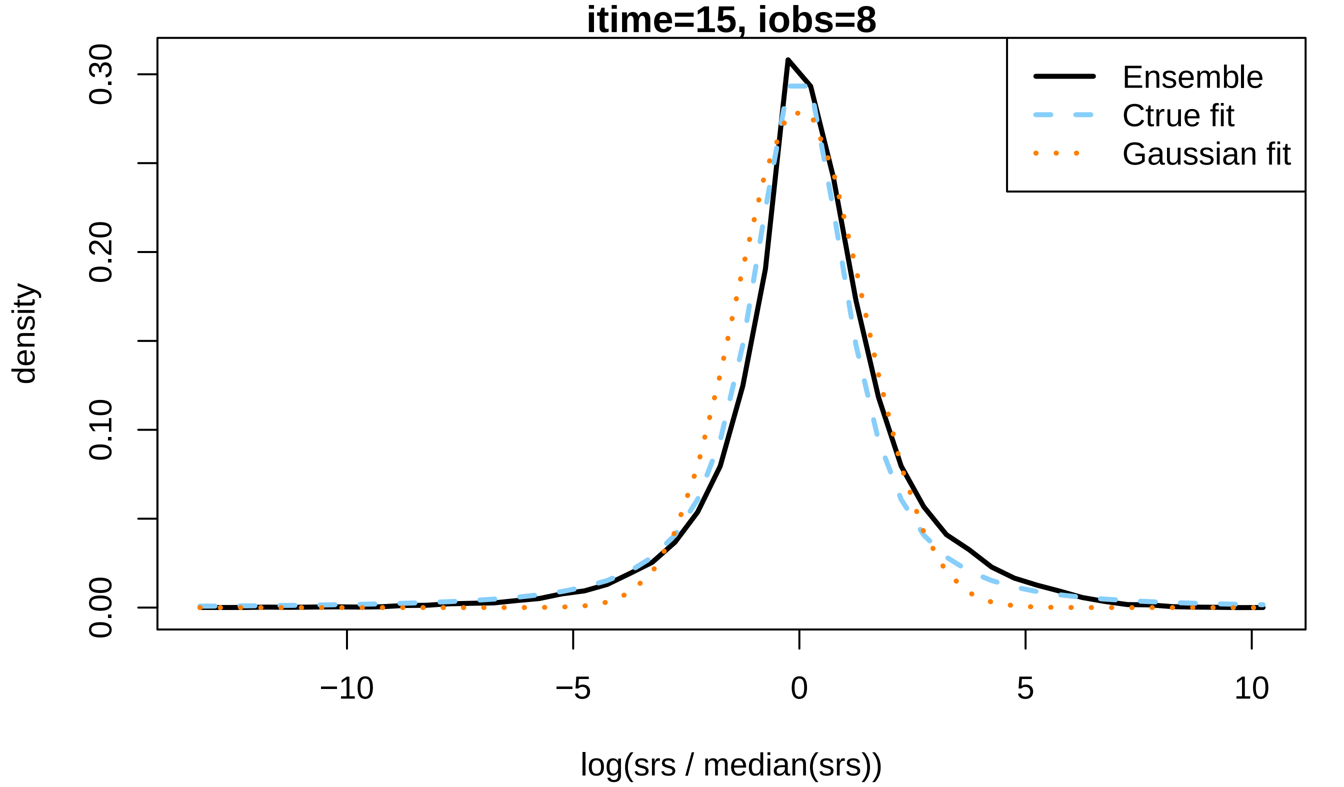
itime=15, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))



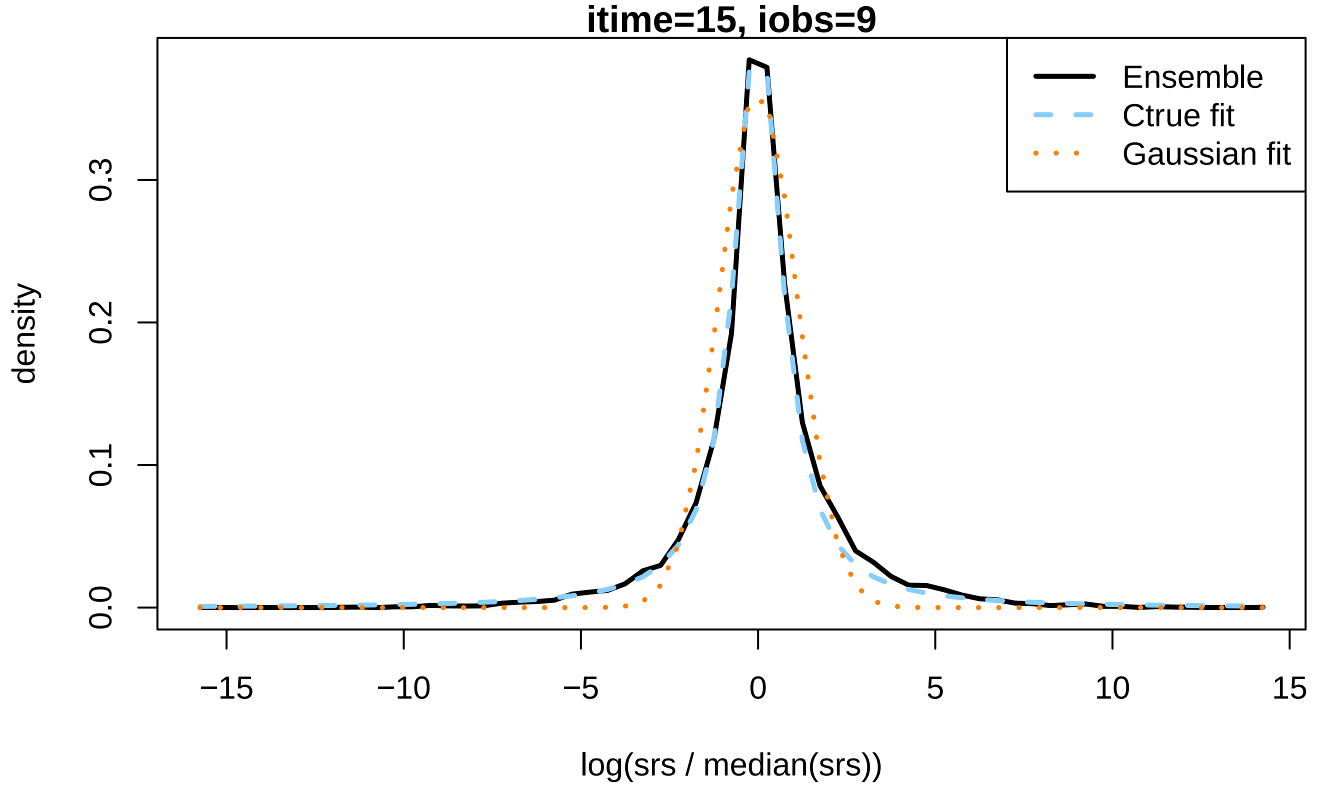
itime=15, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

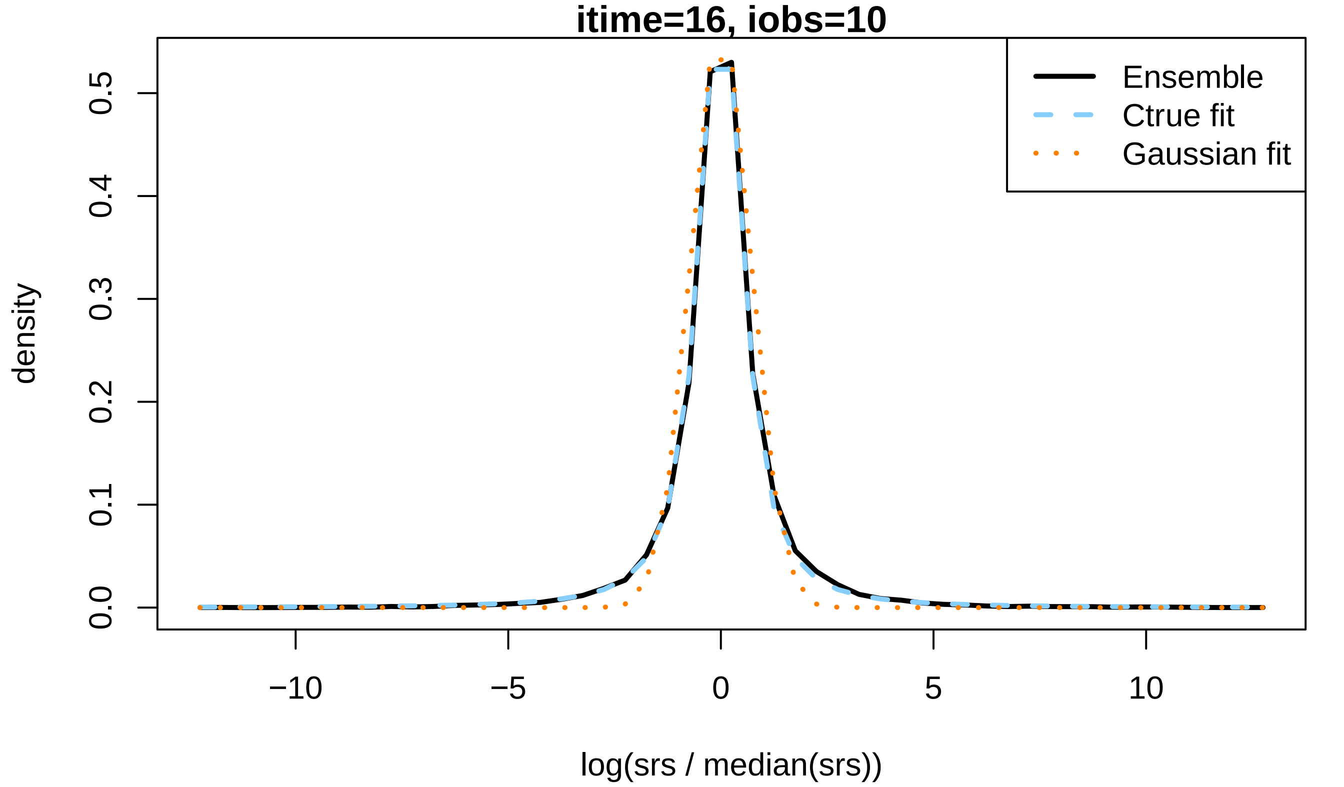

itime=15, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=15, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))

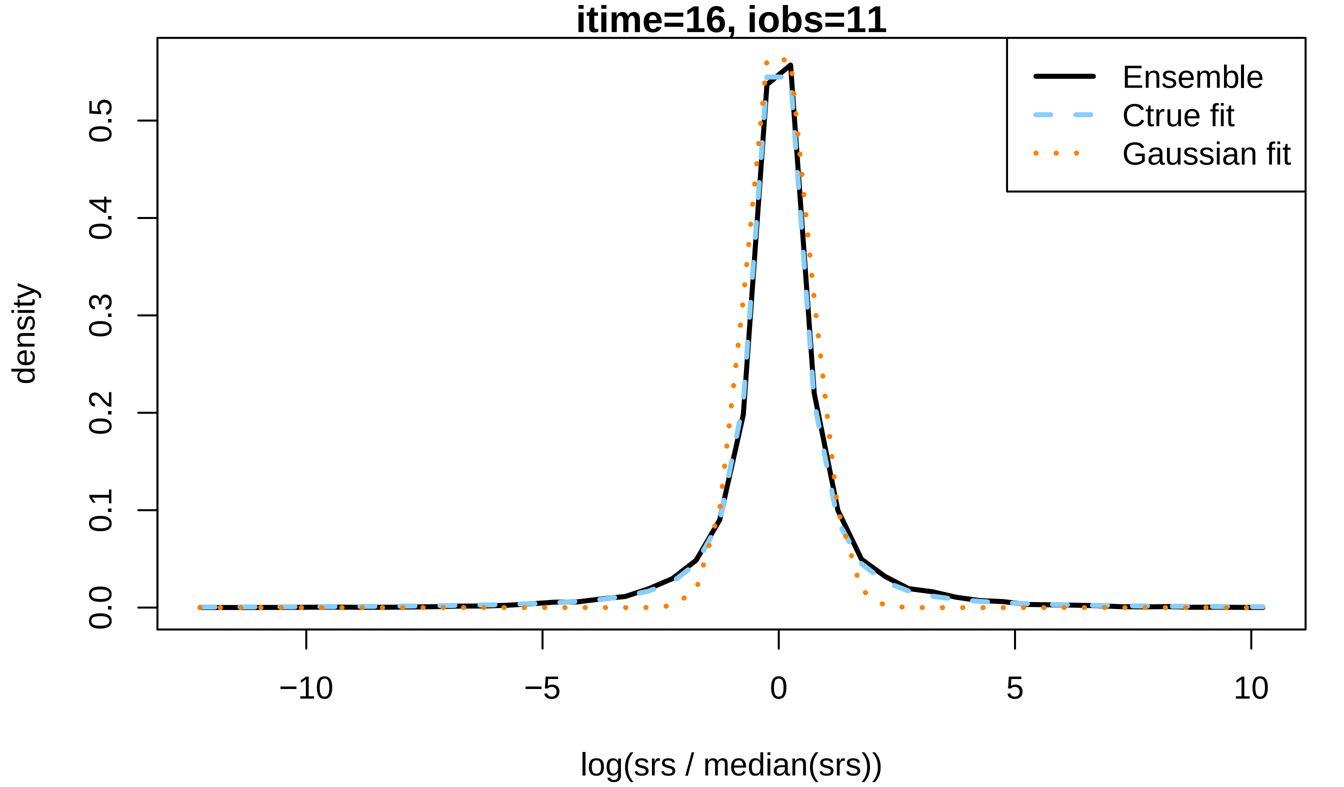


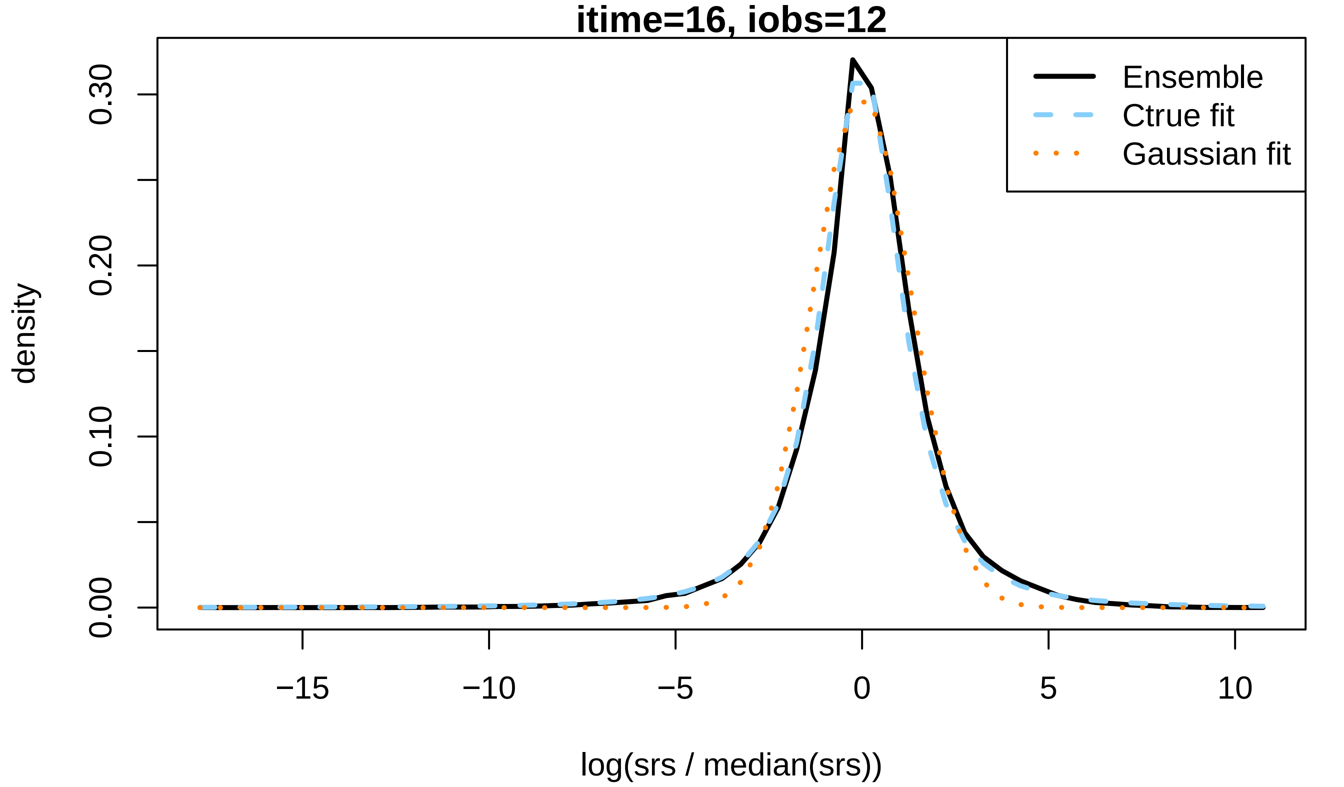
itime=15, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))


itime=15, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
15
log(srs / median(srs))

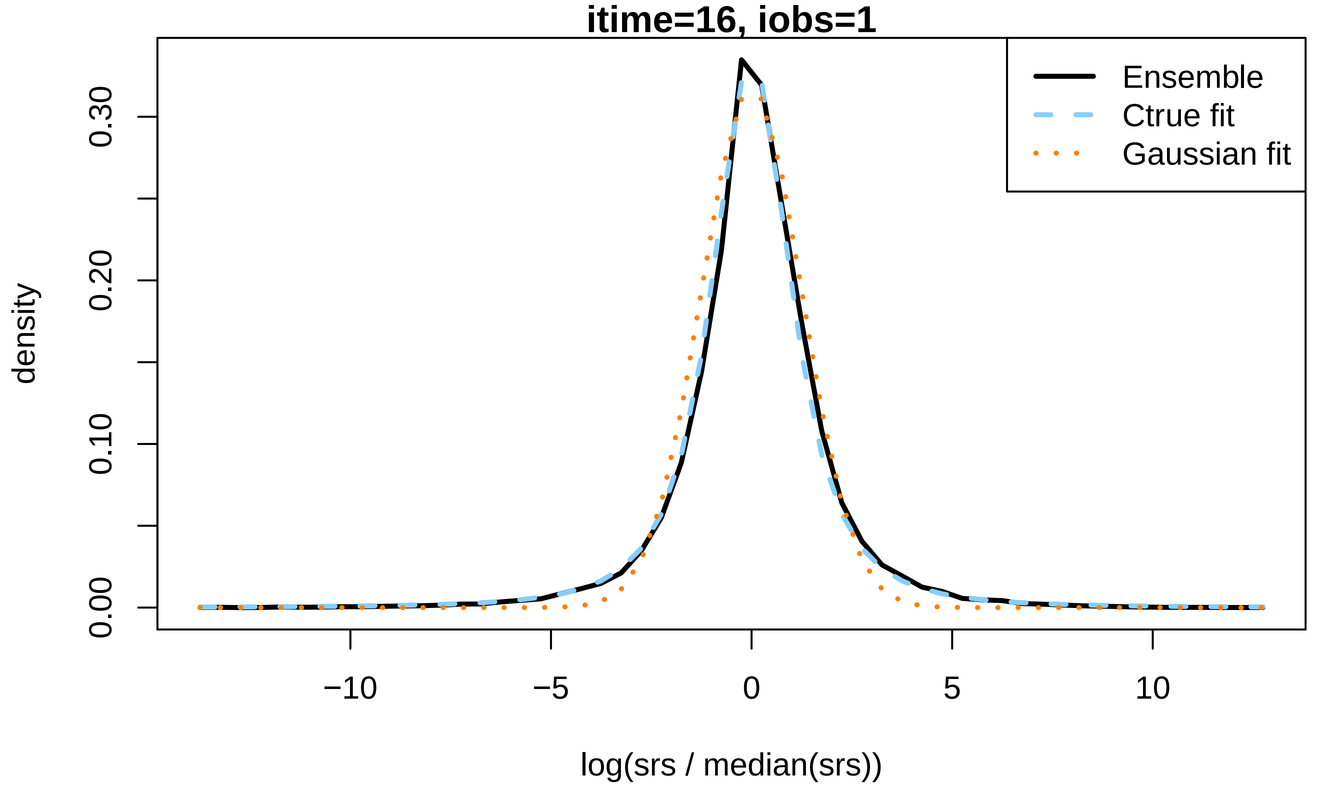


itime=16, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−10
−5
0
5
10
log(srs / median(srs))

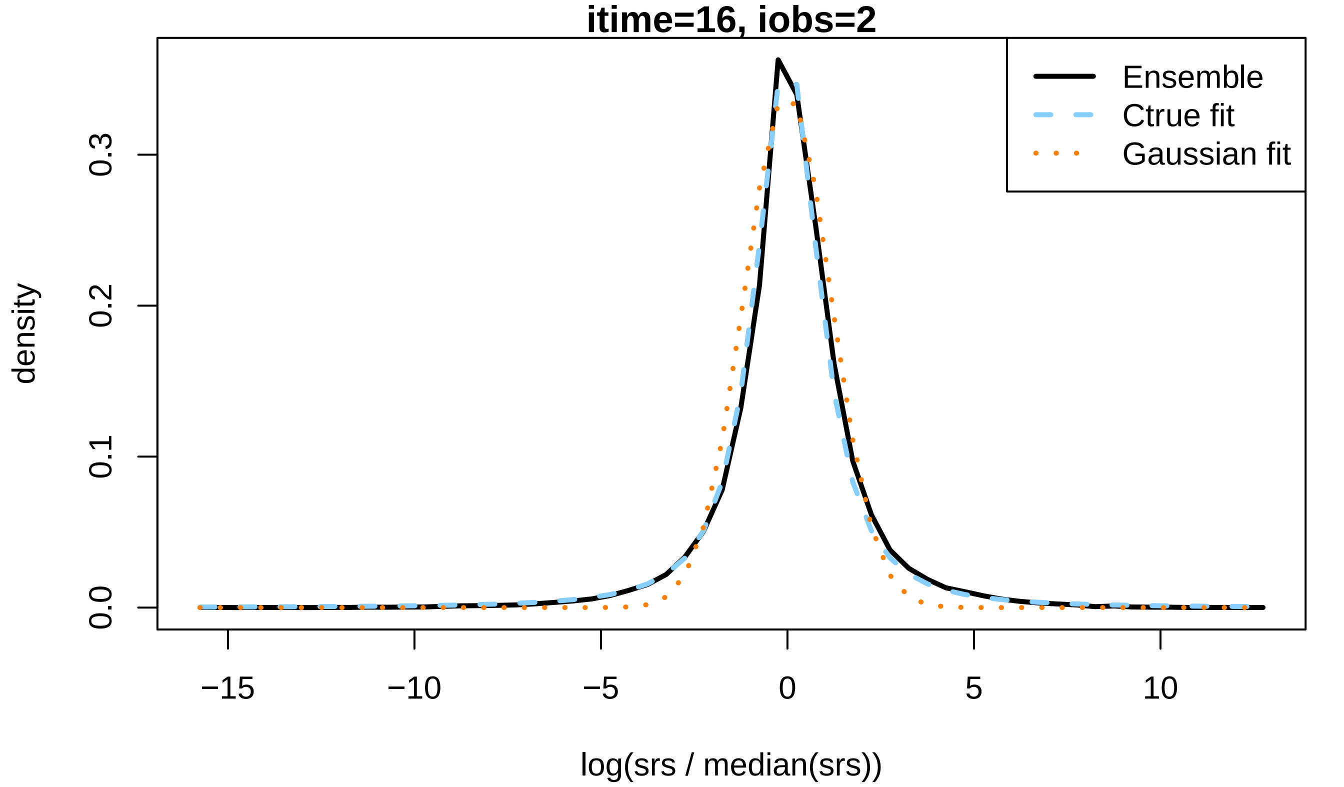

itime=16, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−10
−5
0
5
10
log(srs / median(srs))

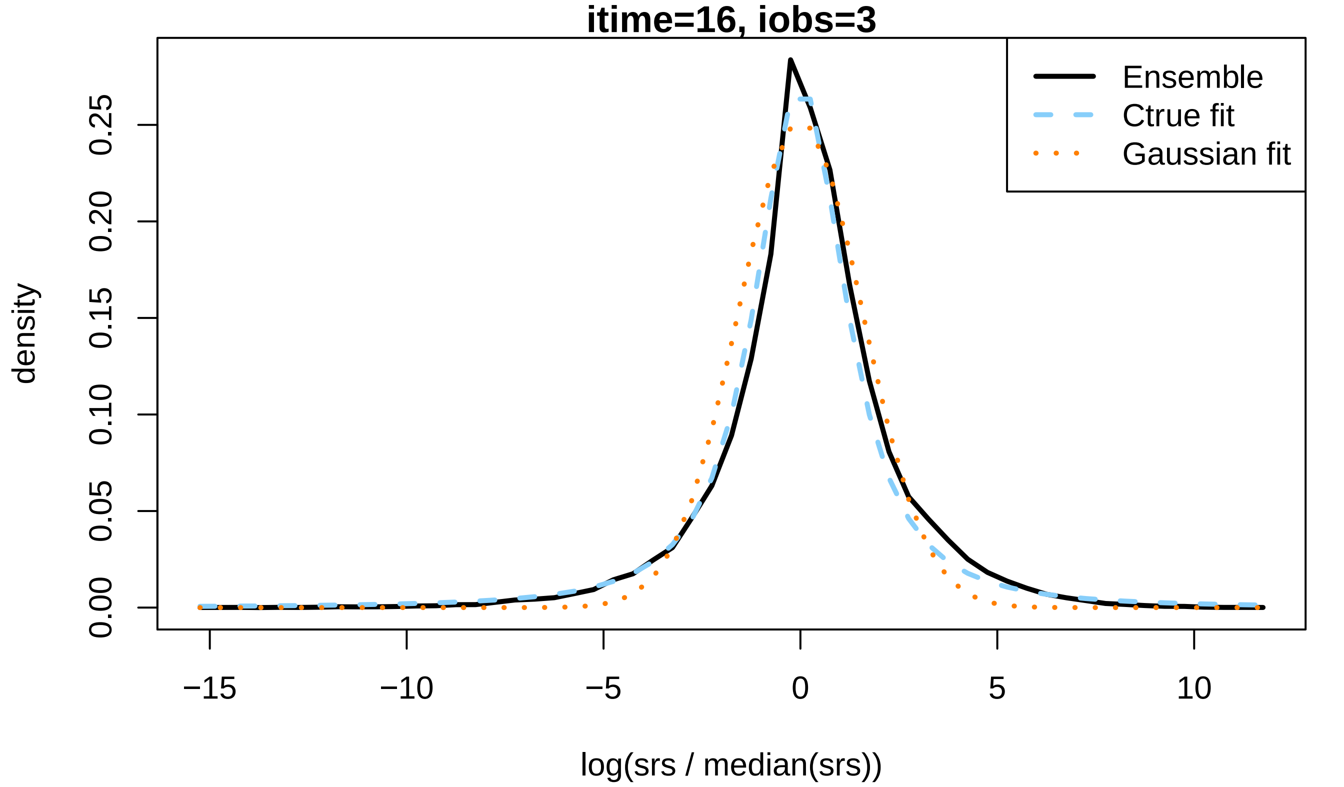


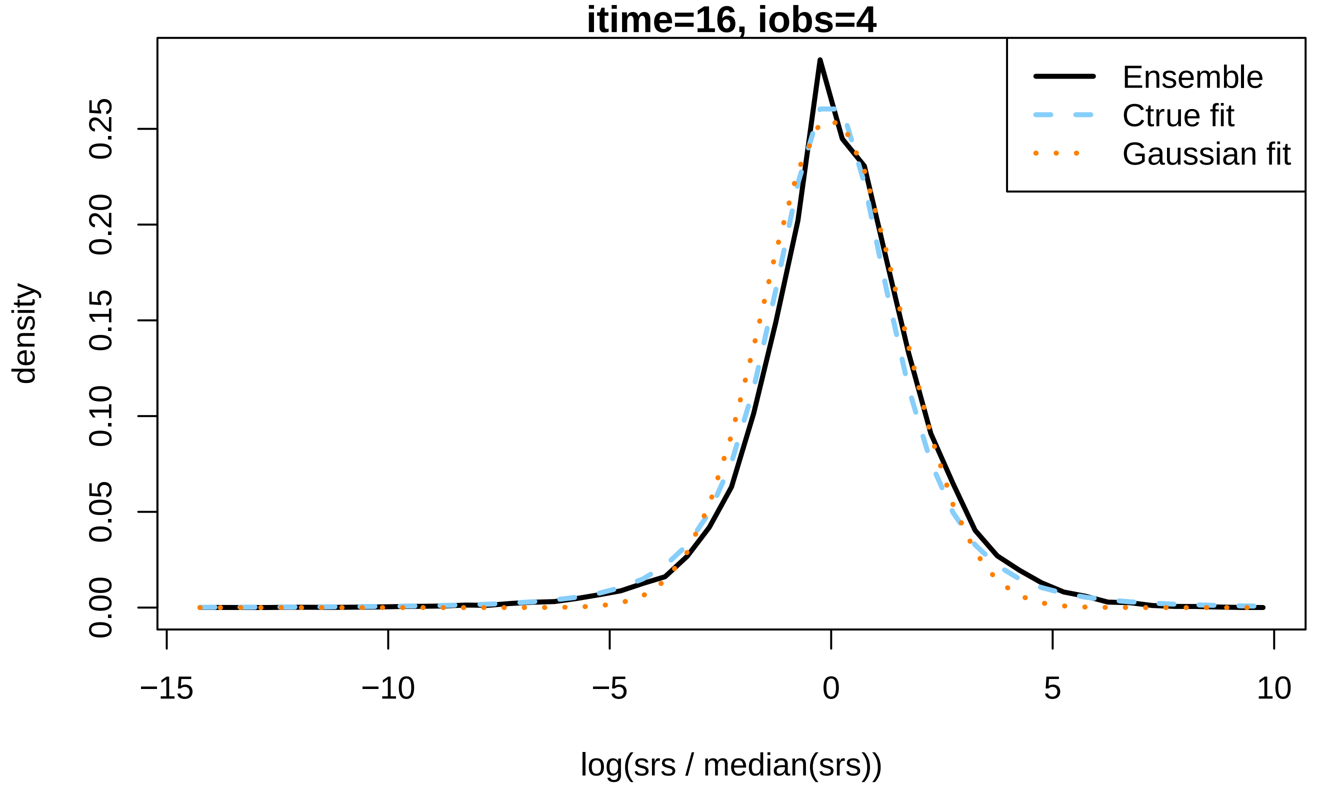
itime=16, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



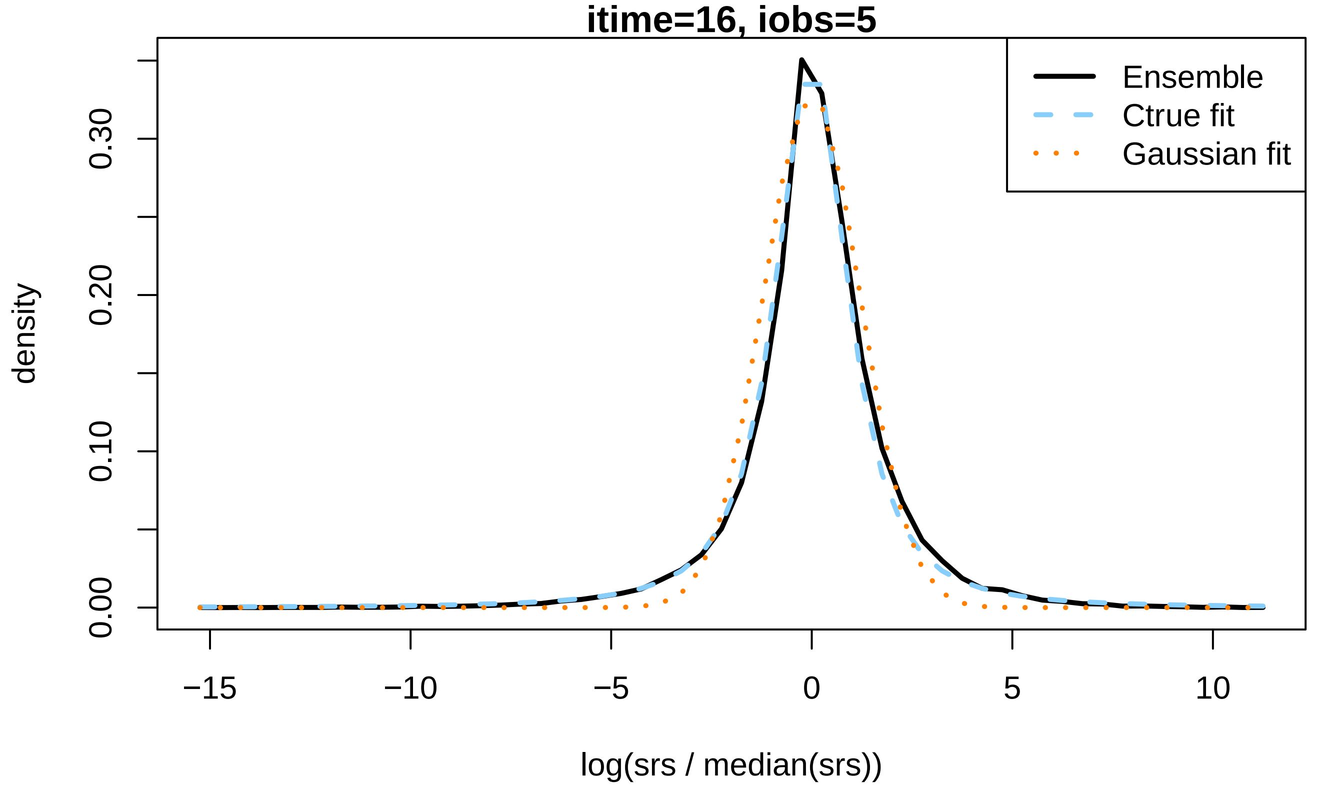
itime=16, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))


itime=16, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

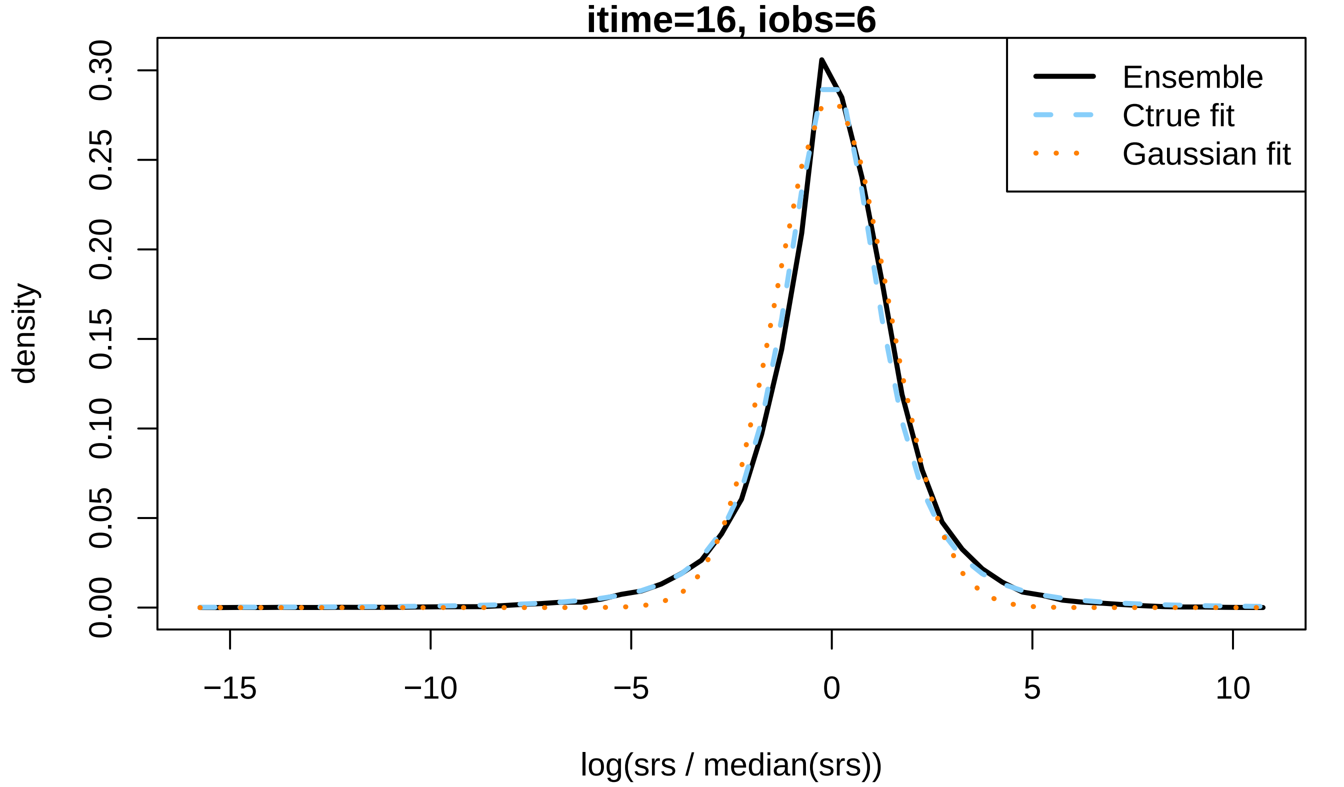

itime=16, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

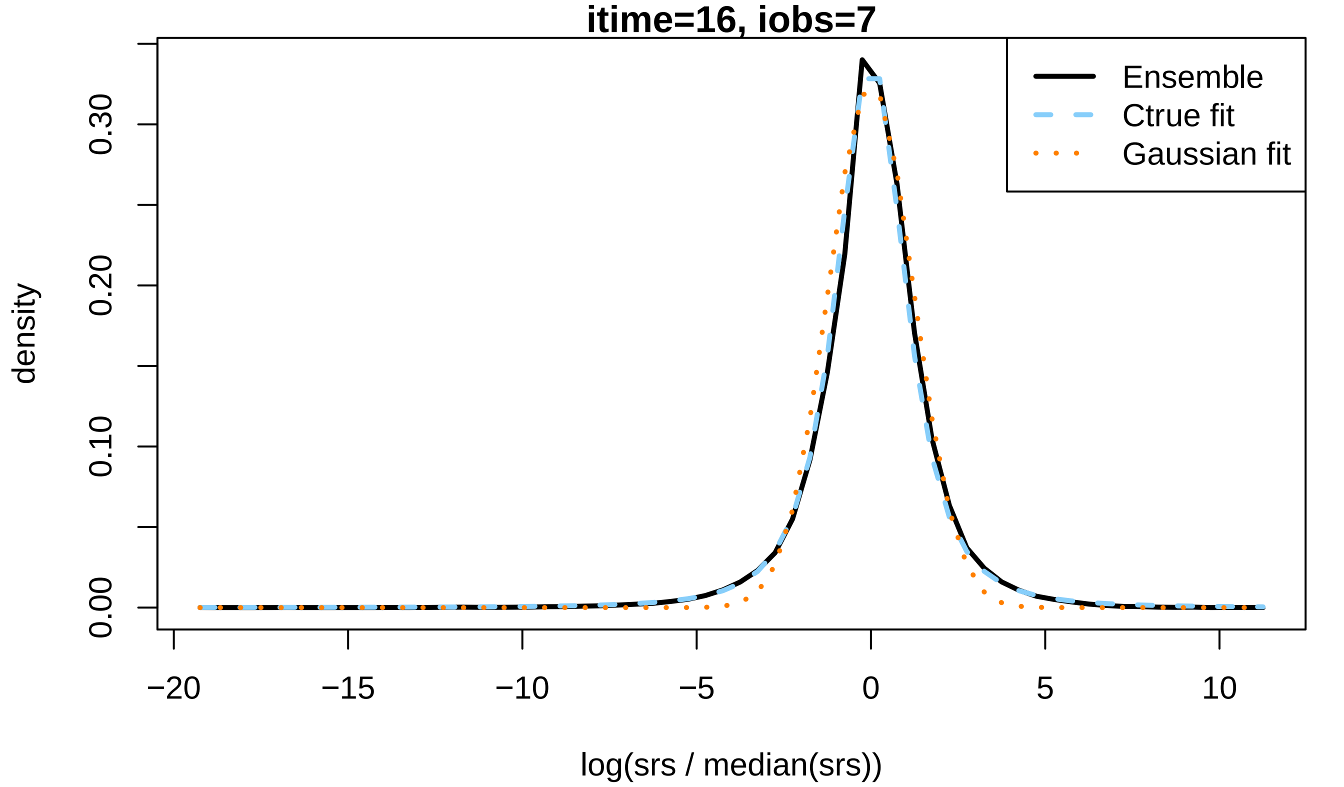


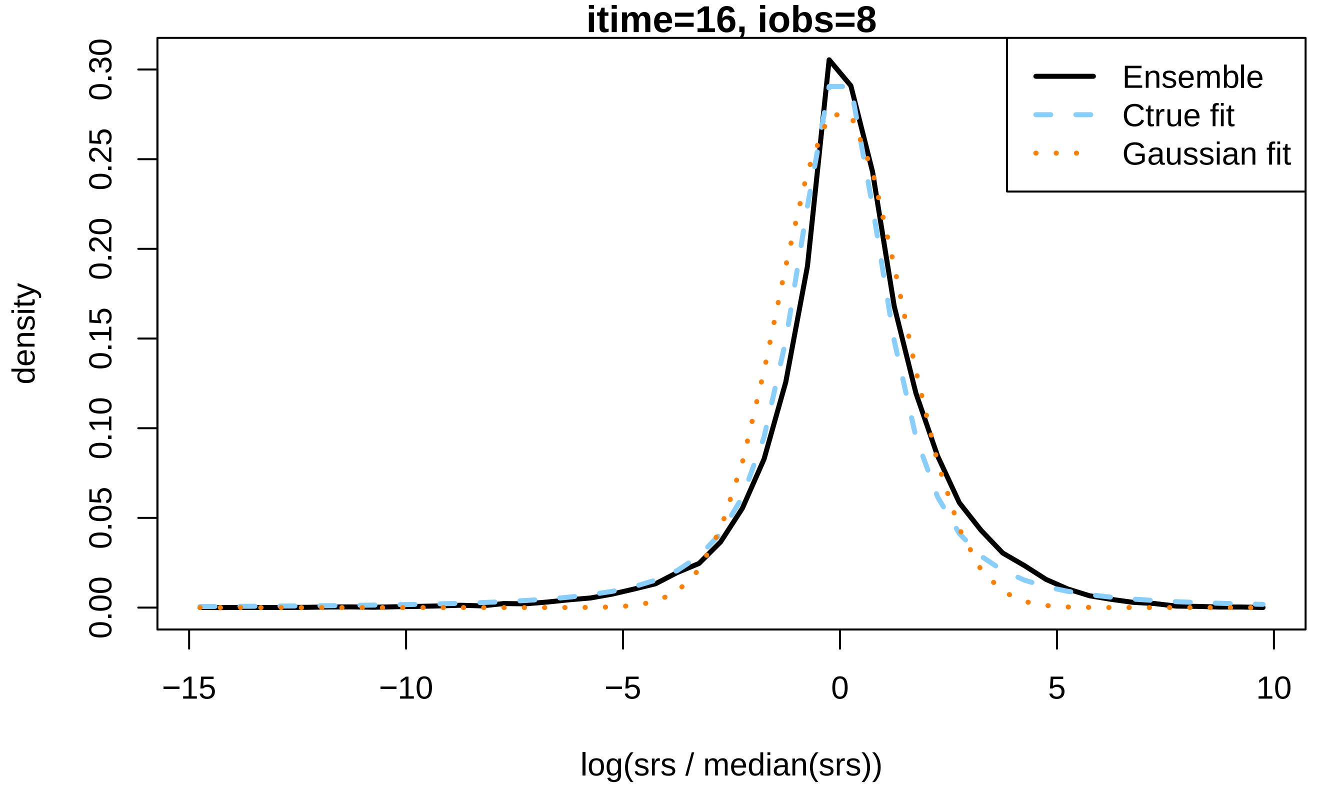
itime=16, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))



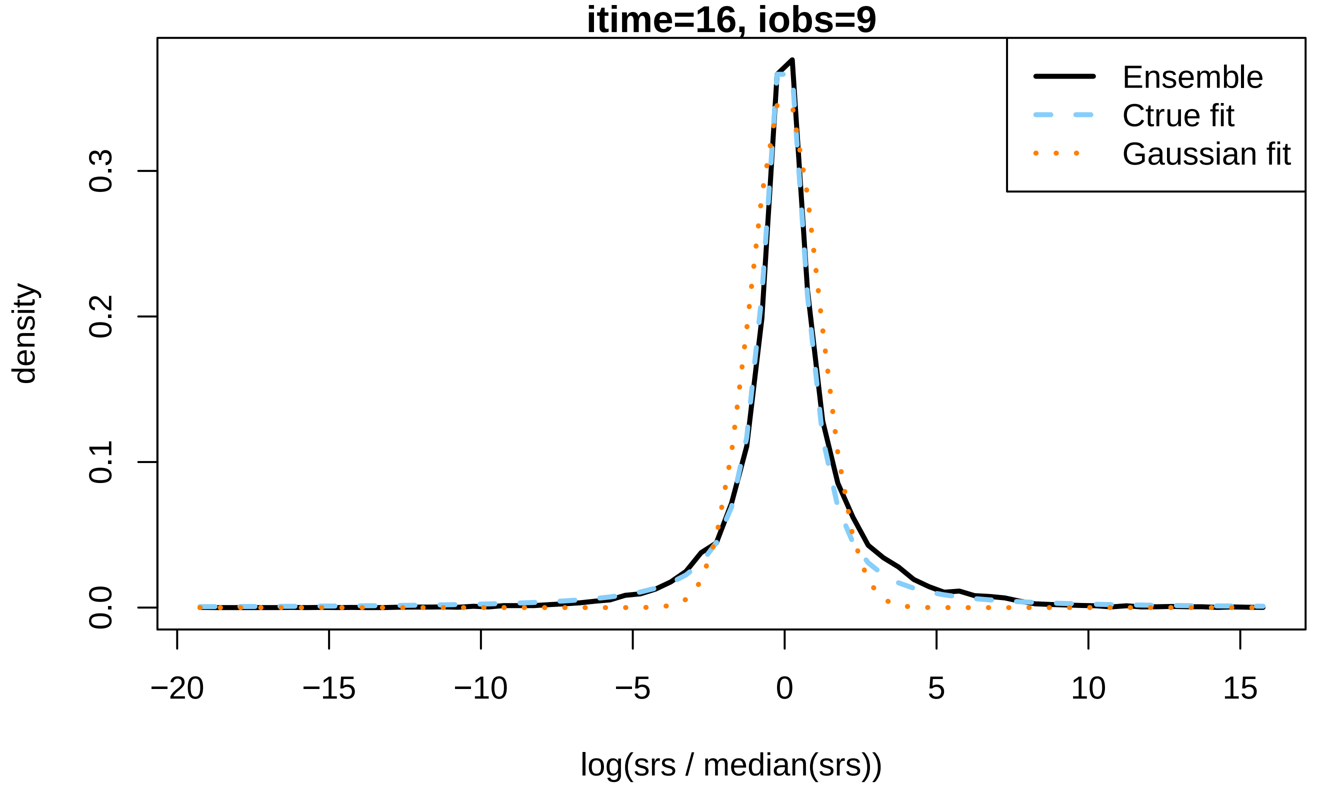
itime=16, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=16, iobs=6
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

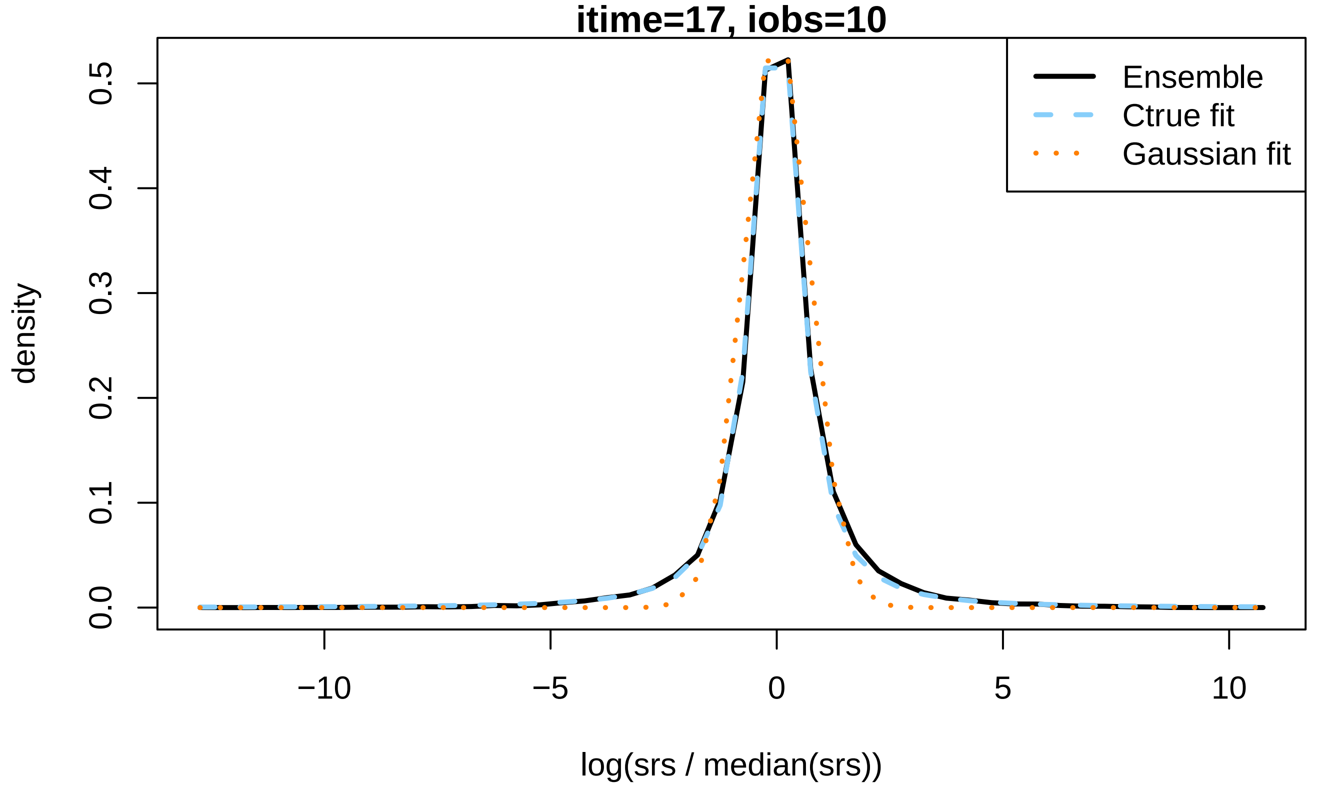

itime=16, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−20
−15
−10
−5
0
5
10
log(srs / median(srs))



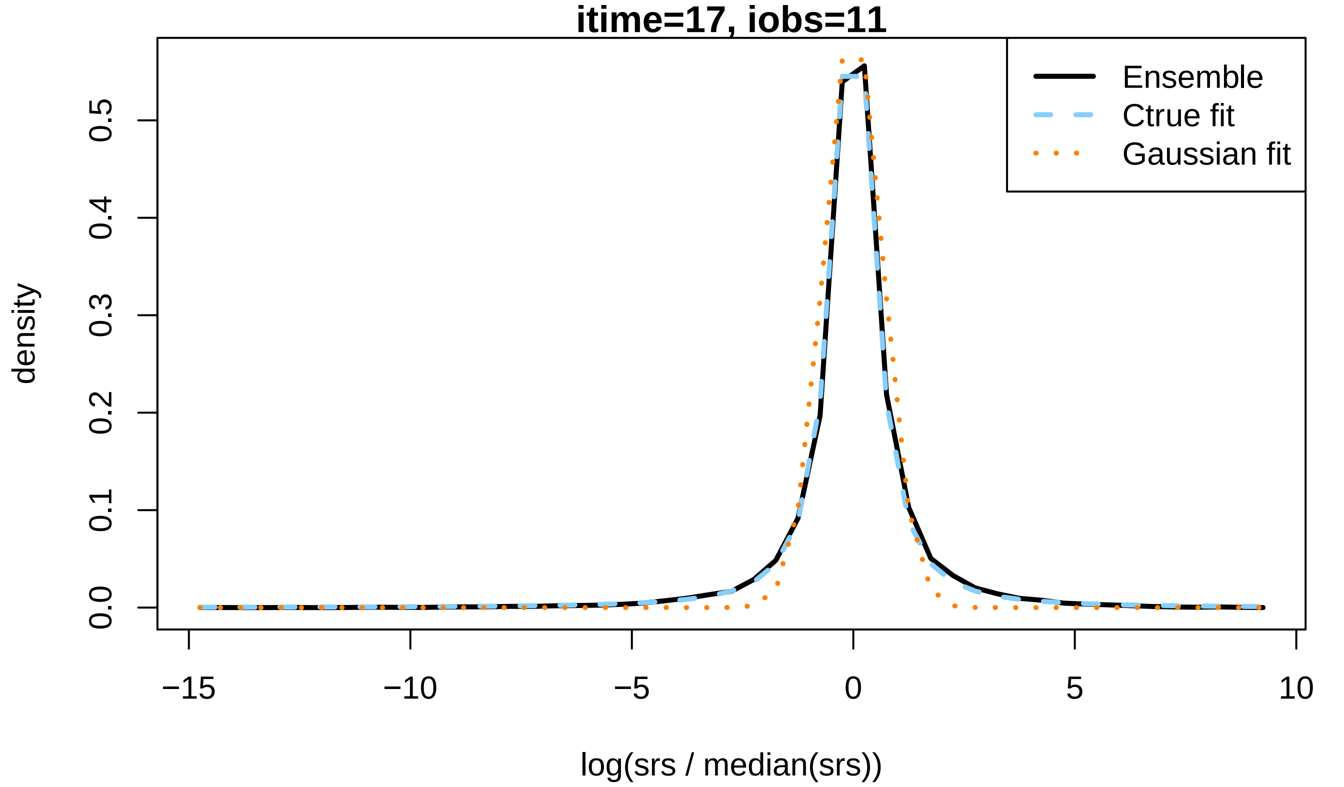
itime=16, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



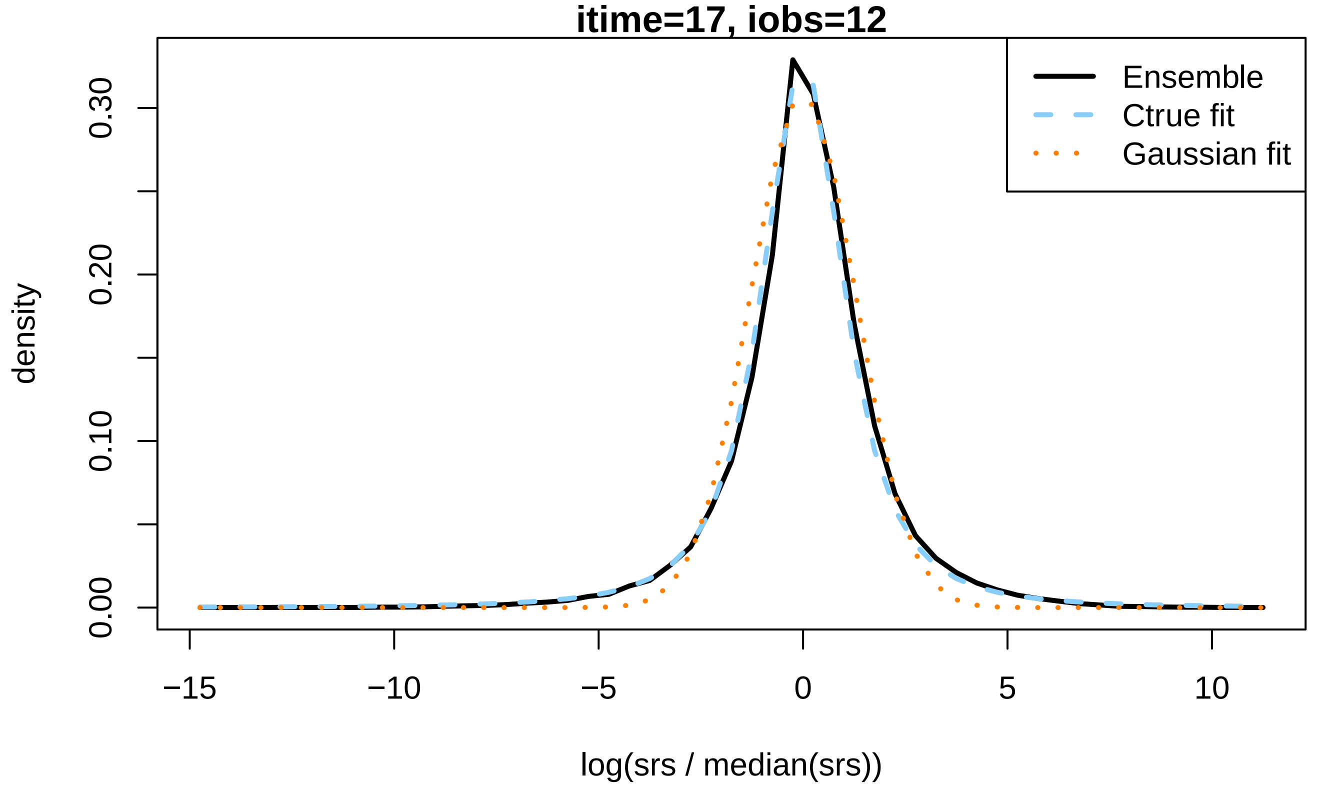
itime=16, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−20
−15
−10
−5
0
5
10
15
log(srs / median(srs))

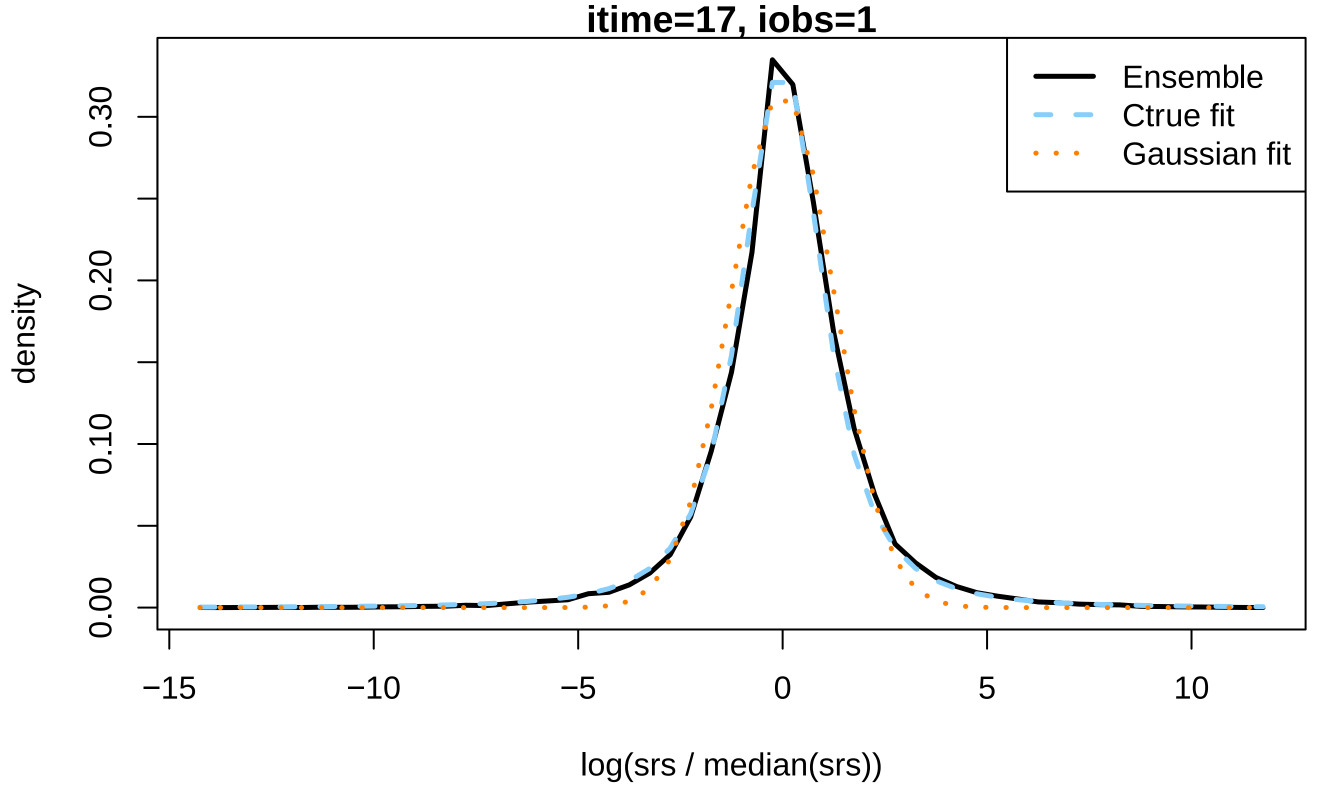


itime=17, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−10
−5
0
5
10
log(srs / median(srs))

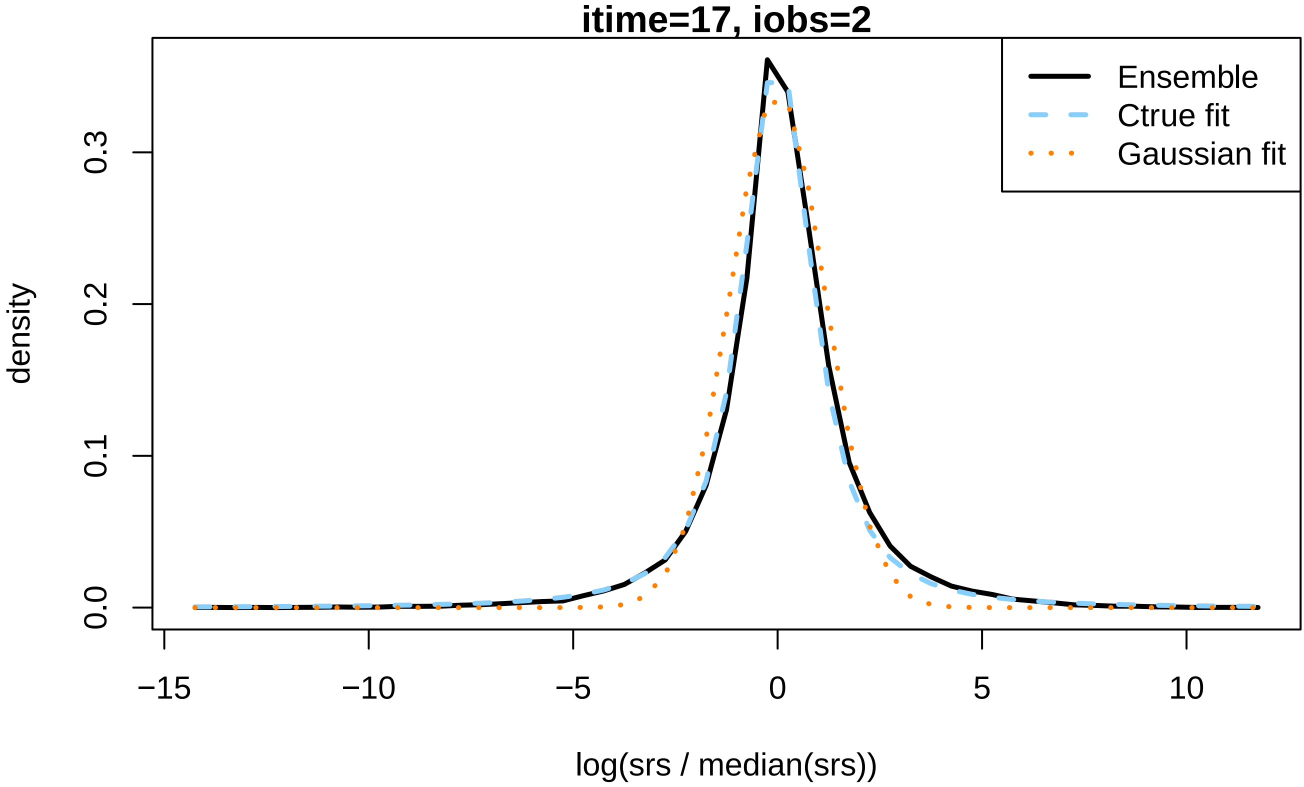


itime=17, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−15
−10
−5
0
5
10
log(srs / median(srs))

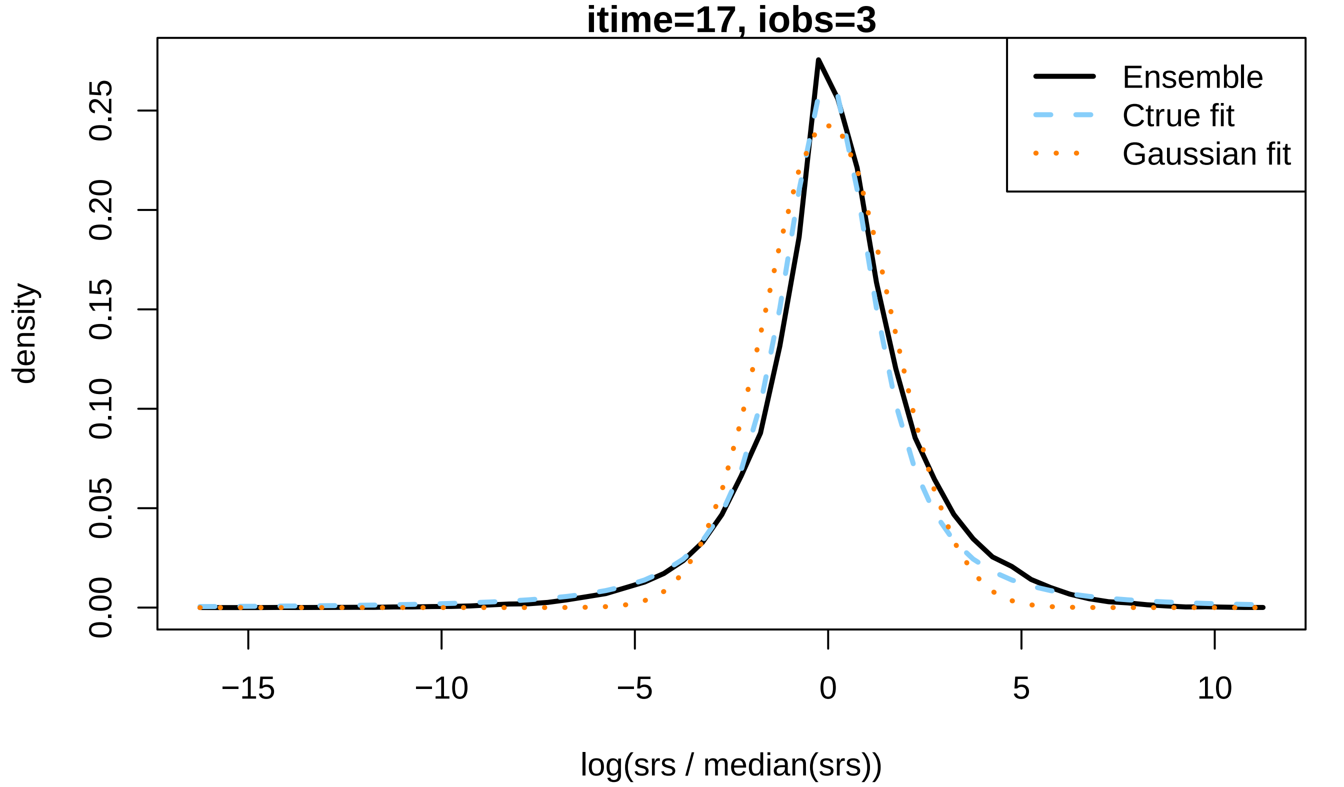

itime=17, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



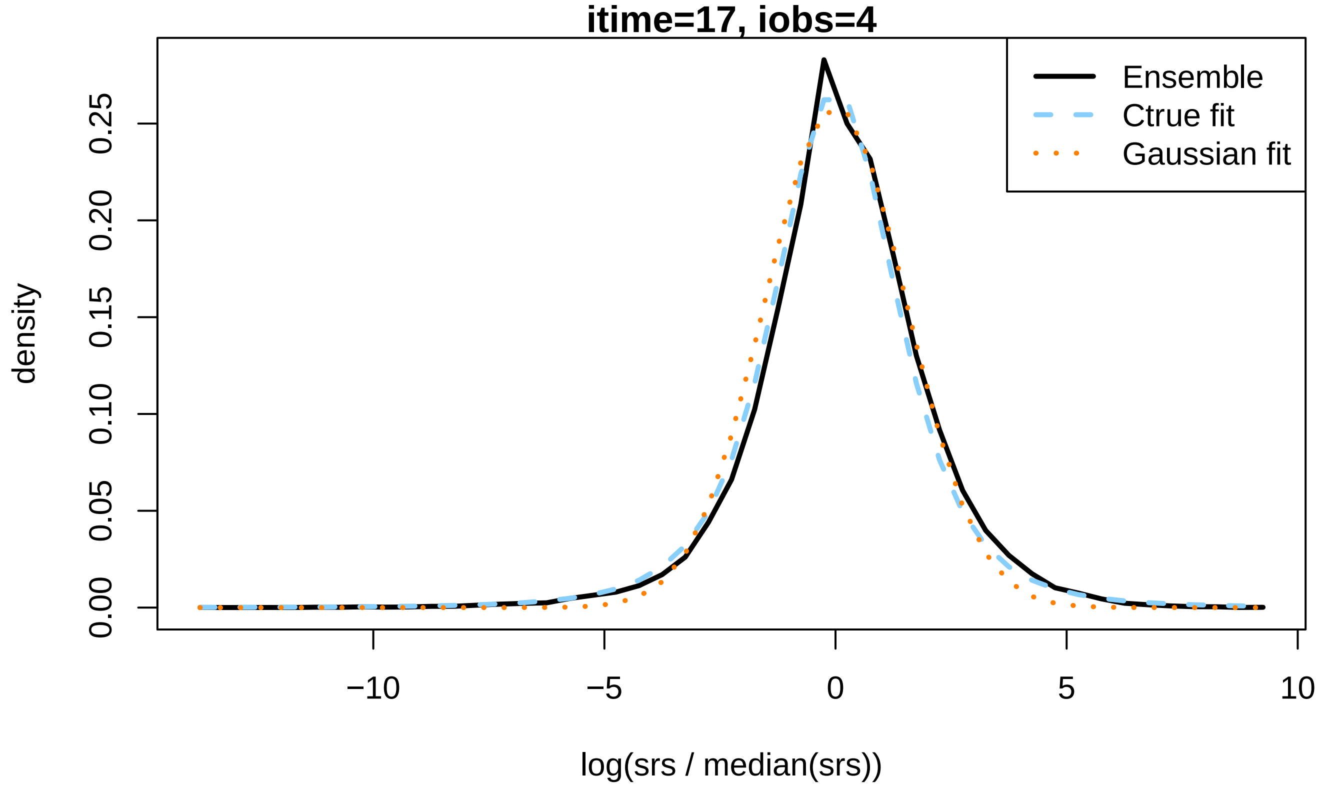
itime=17, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

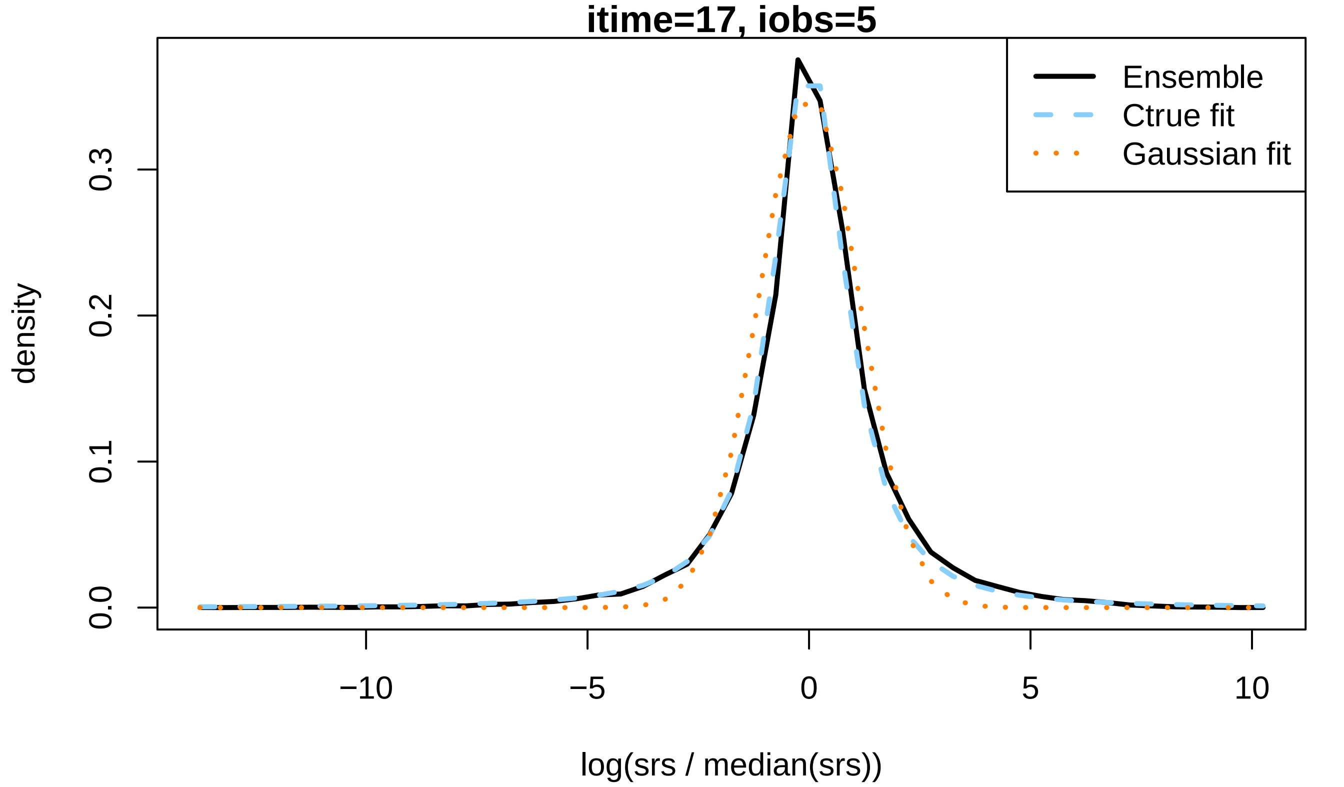


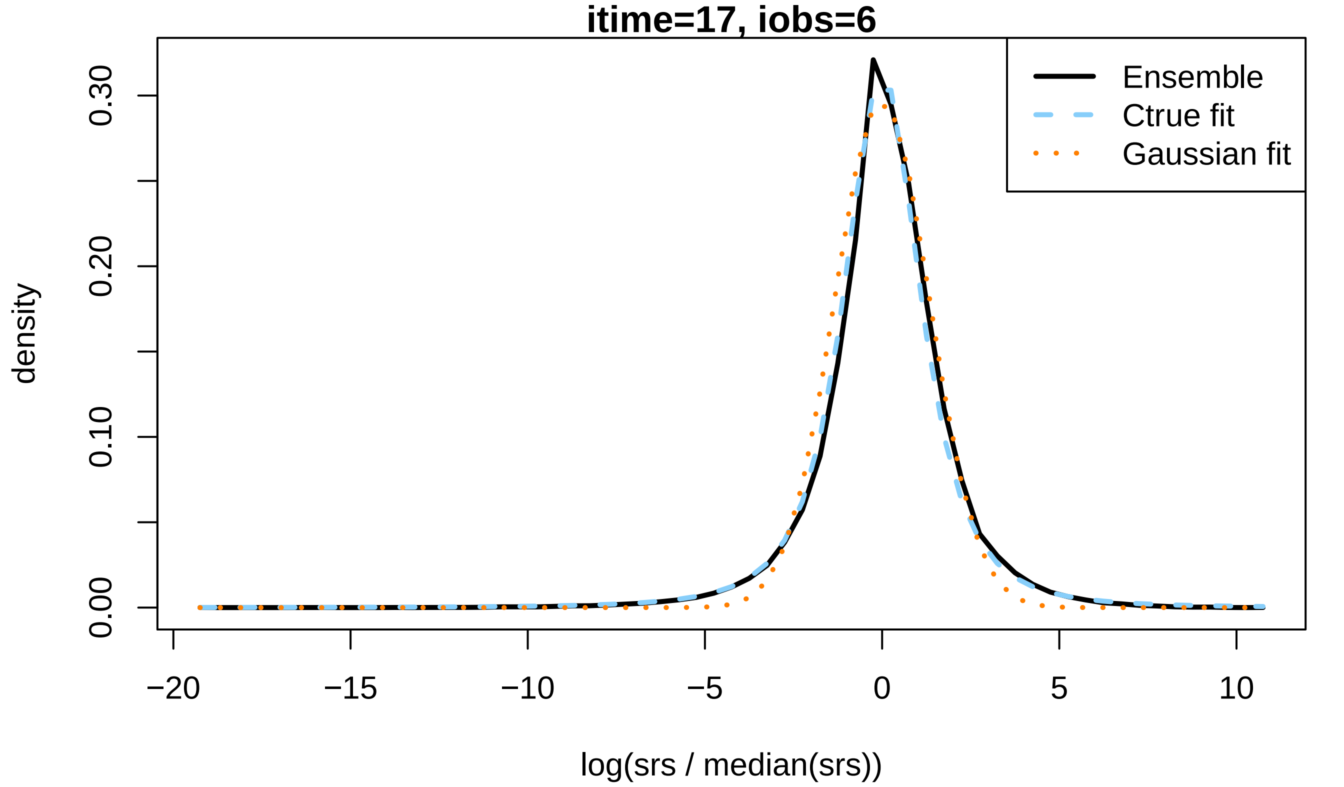
itime=17, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))



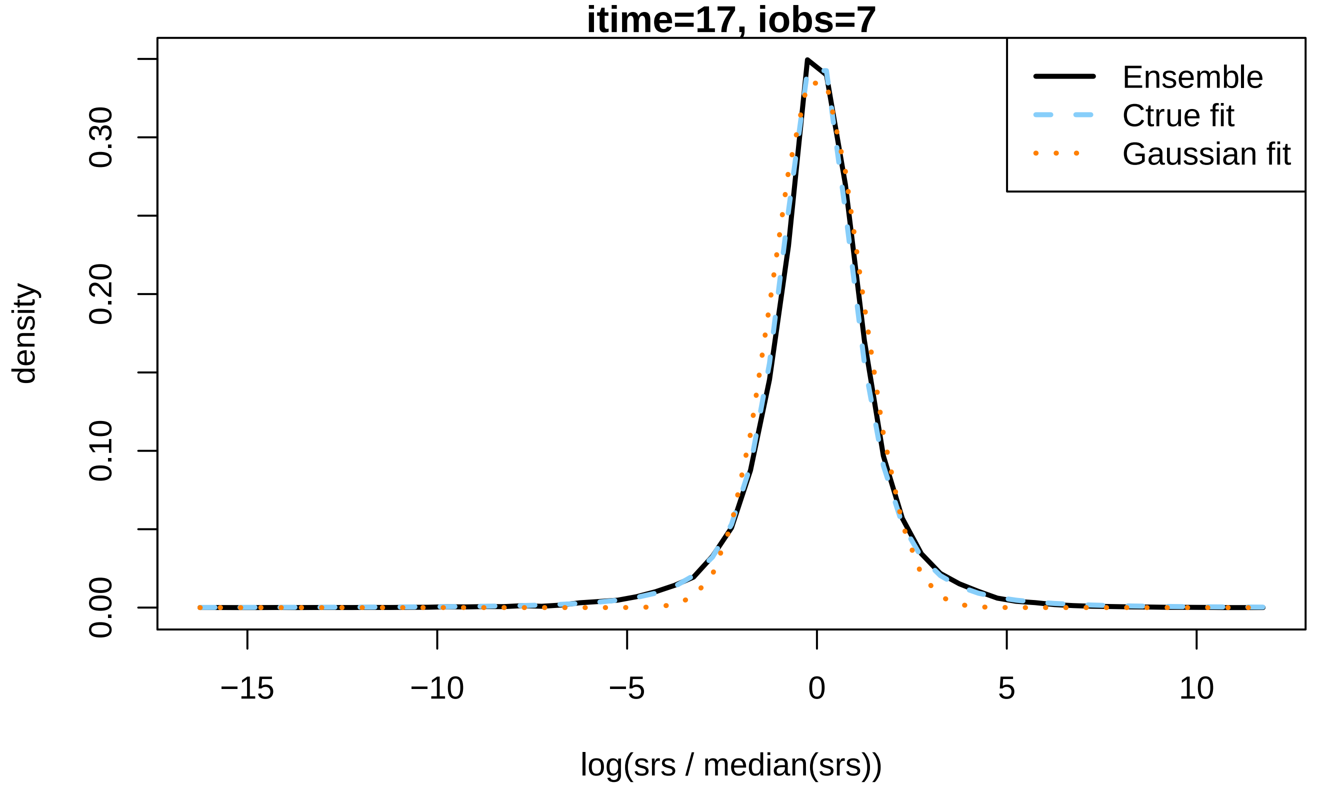
itime=17, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

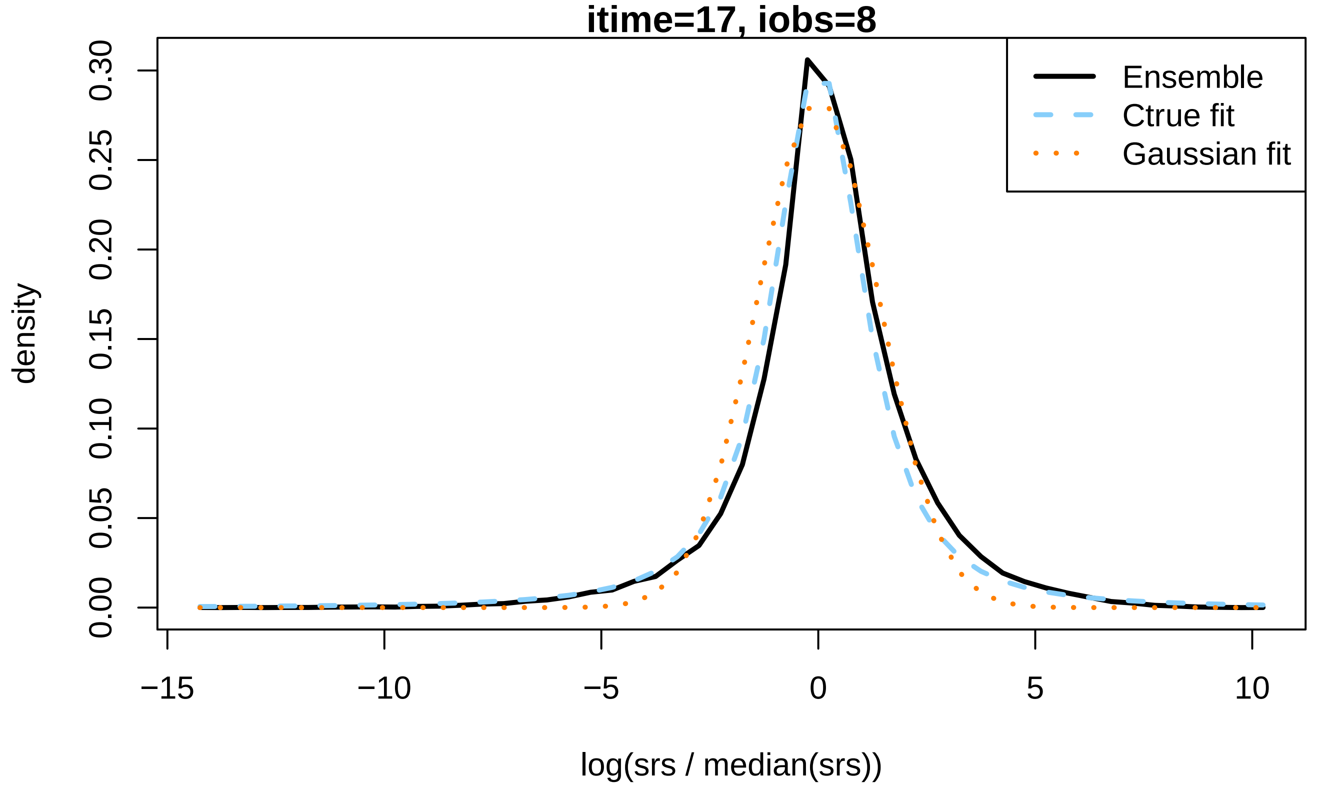

itime=17, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−10
−5
0
5
10
log(srs / median(srs))

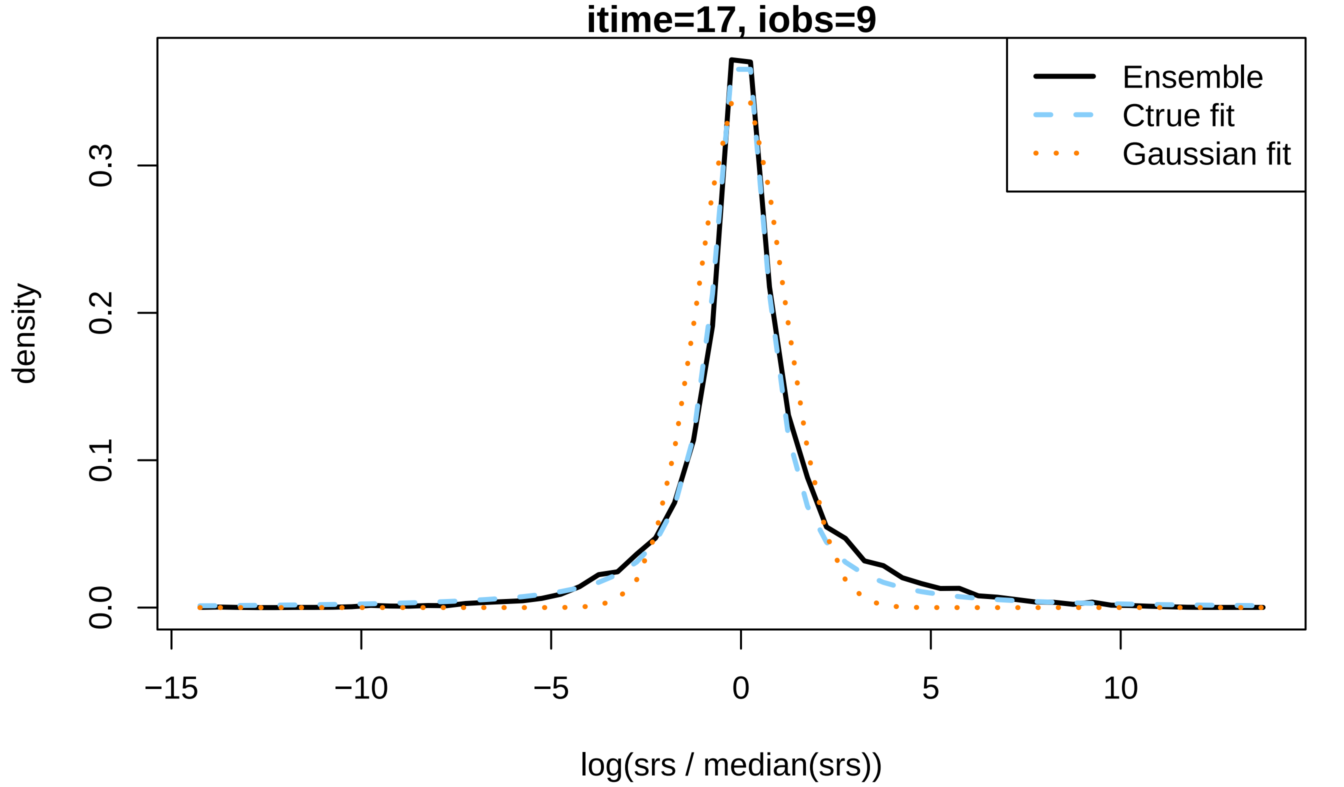

itime=17, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−10
−5
0
5
10
log(srs / median(srs))



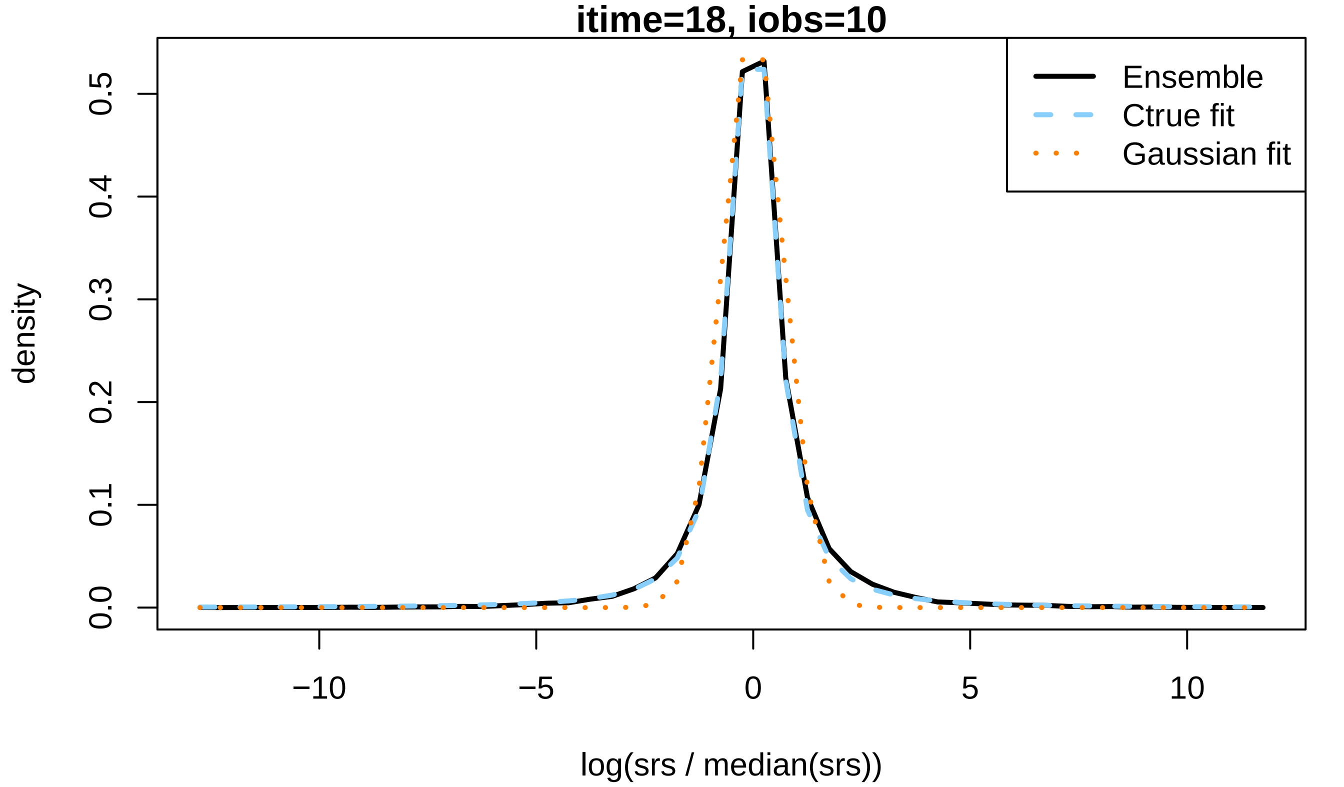
itime=17, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−20
−15
−10
−5
0
5
10
log(srs / median(srs))

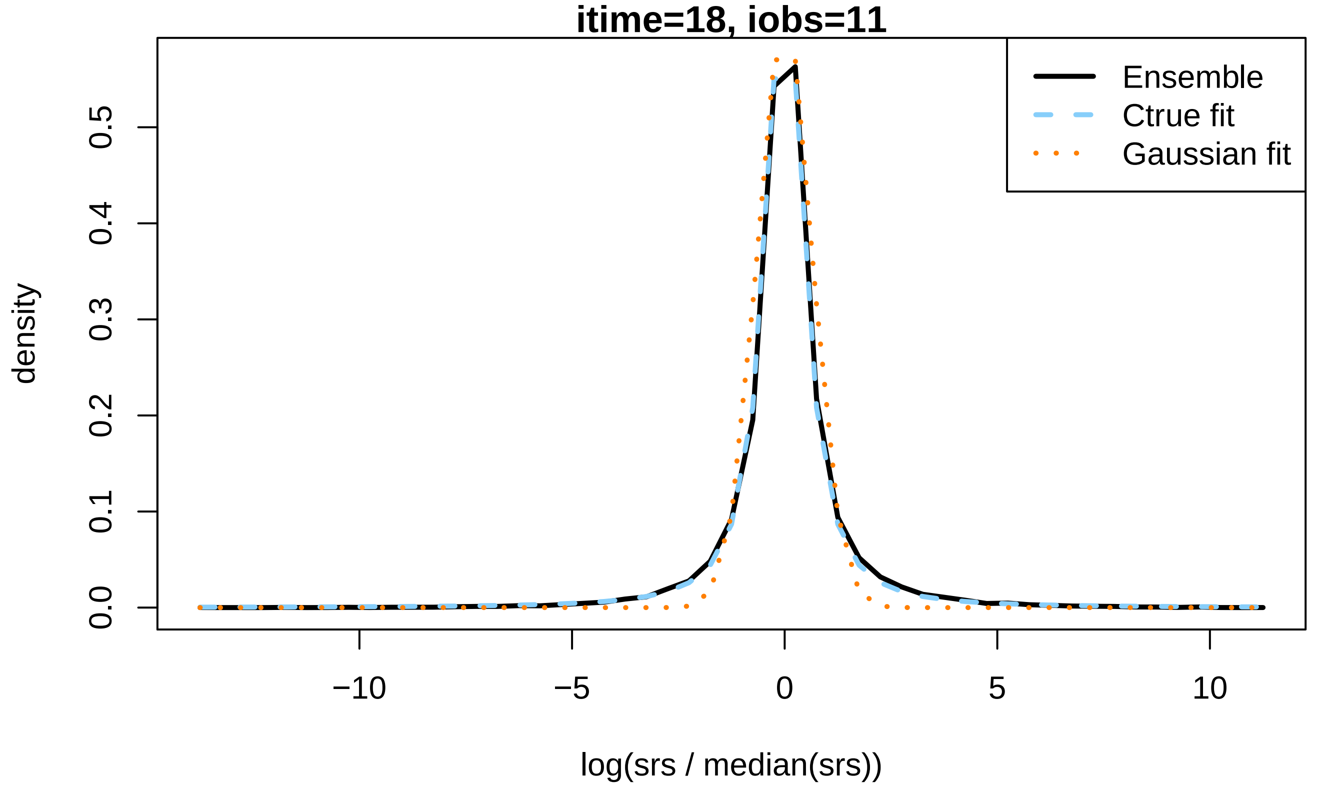

itime=17, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

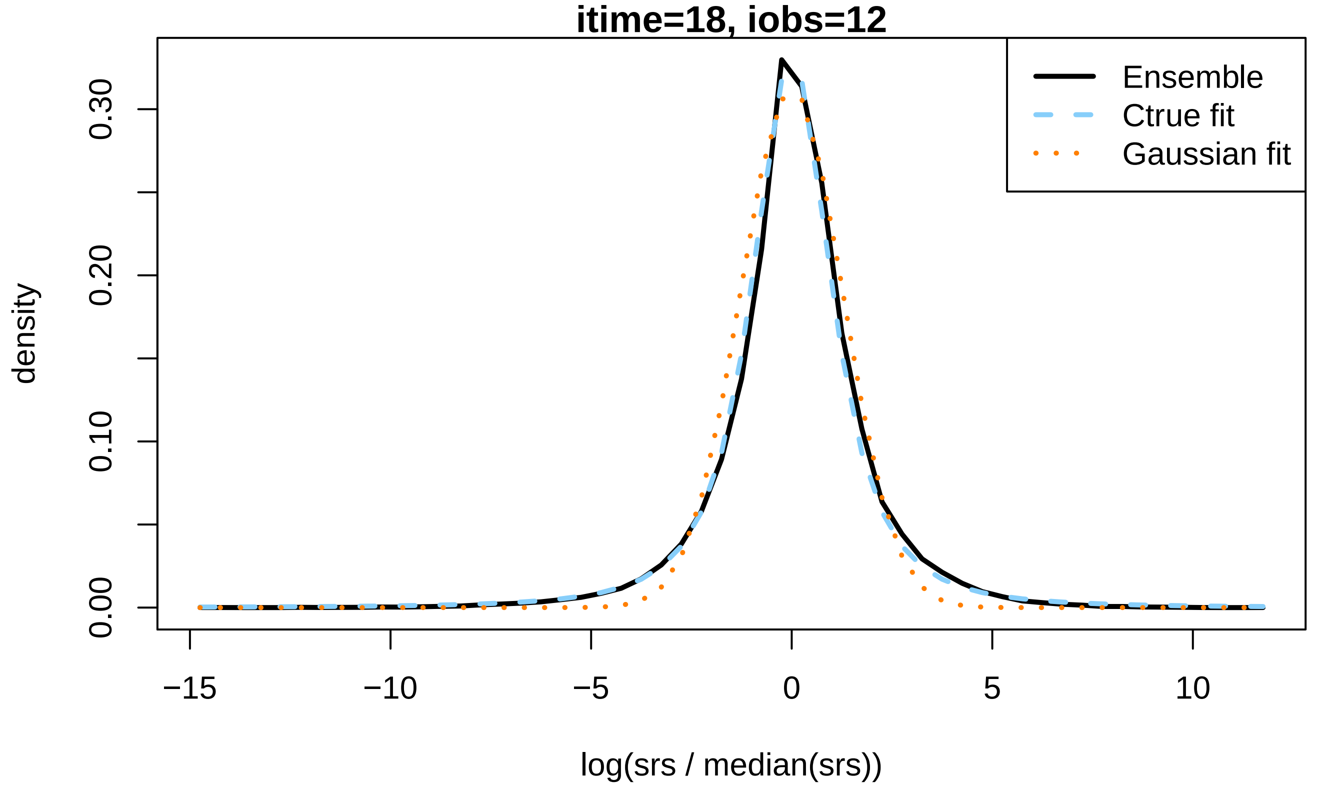

itime=17, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

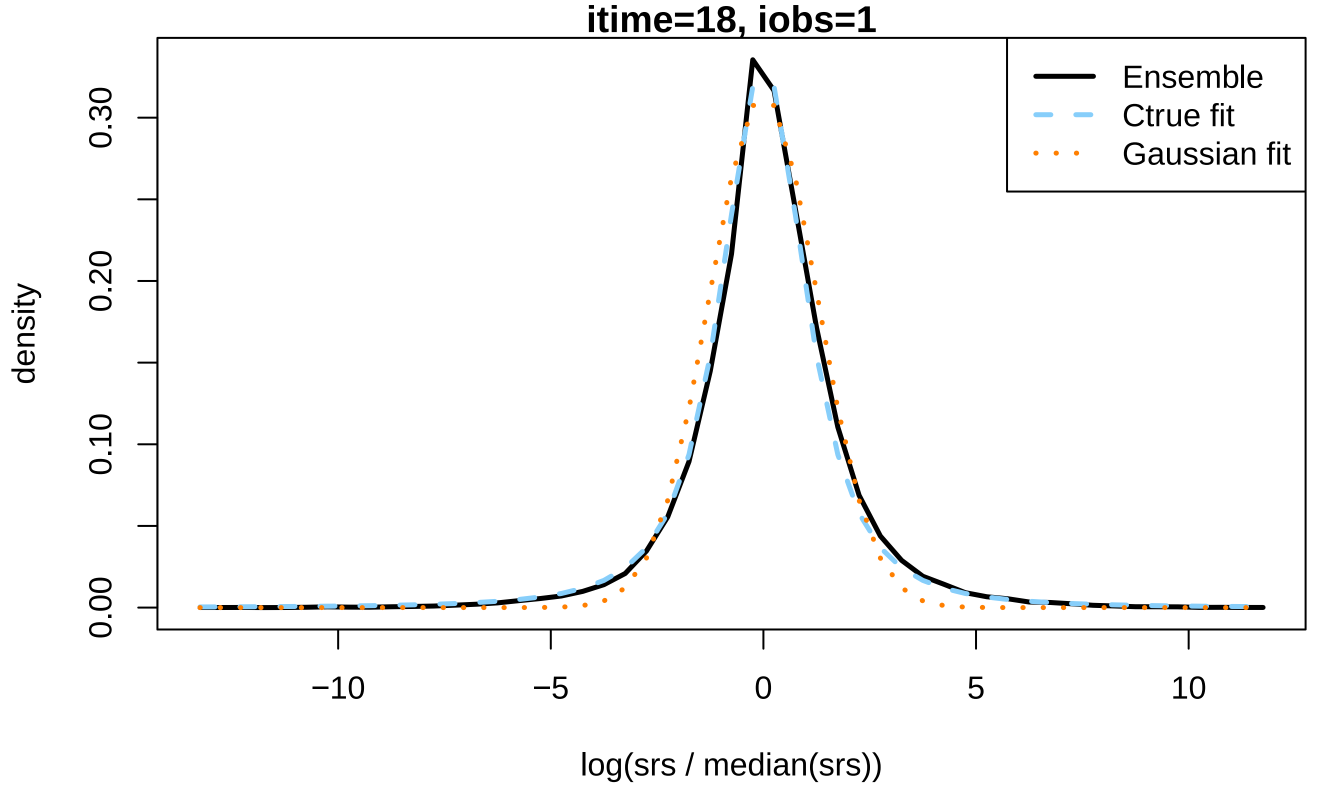


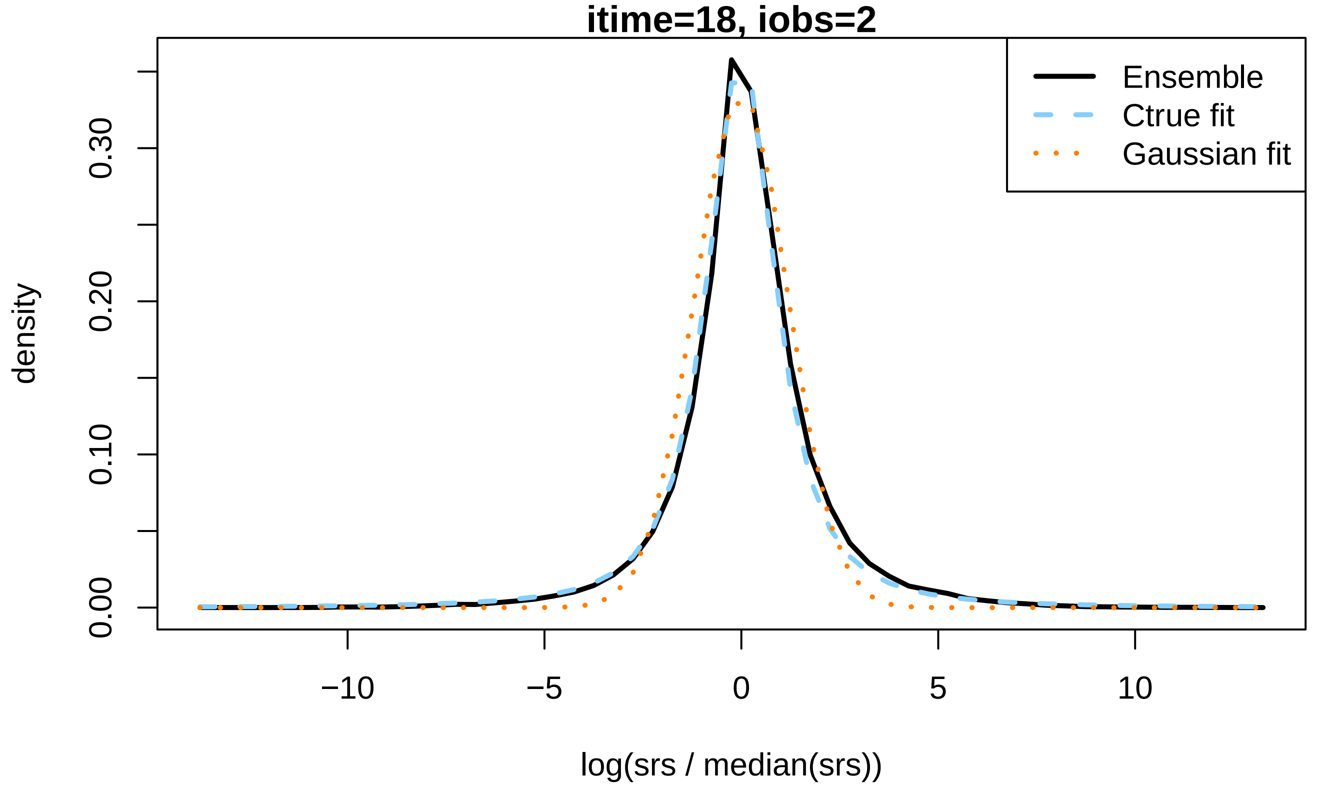
itime=17, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=18, iobs=10
density
0.0
0.1
0.2
0.3
0.4
0.5
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

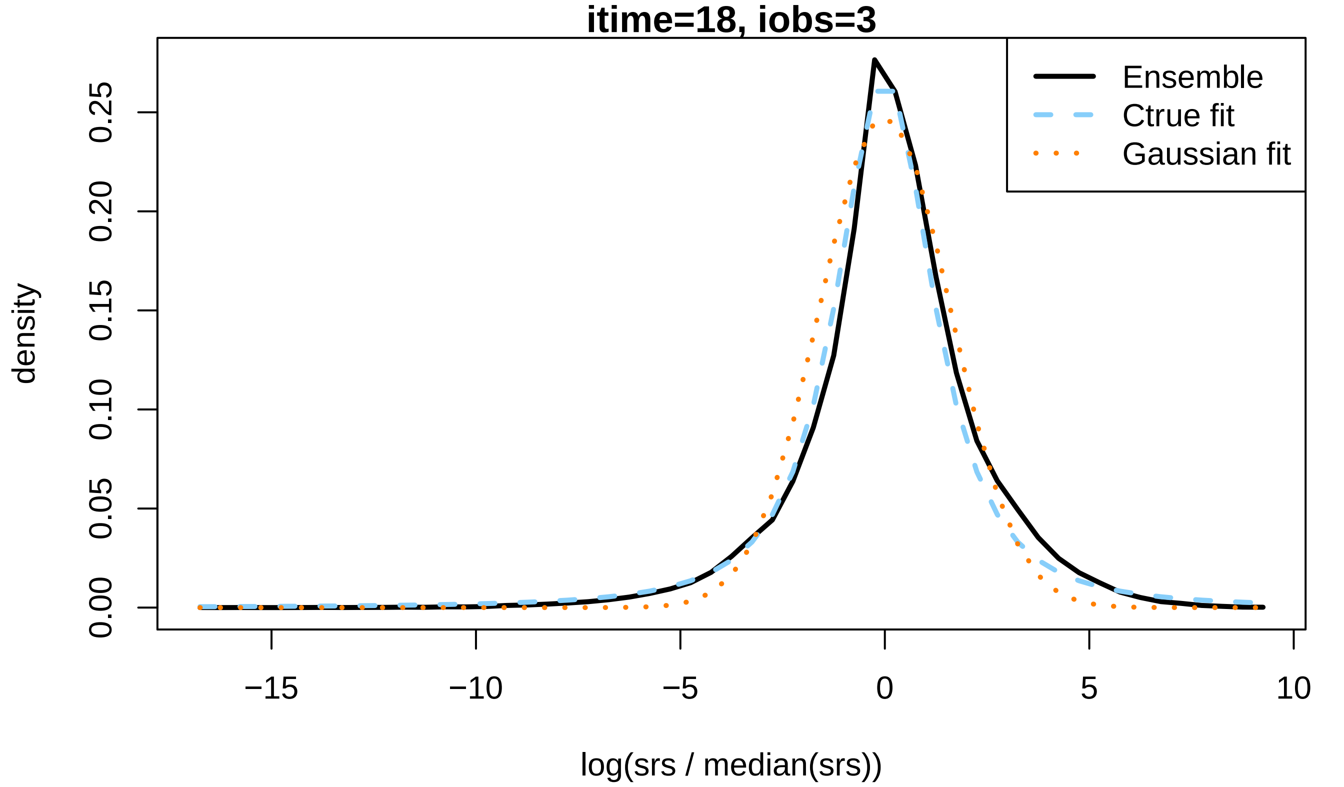

itime=18, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−10
−5
0
5
10
log(srs / median(srs))


itime=18, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

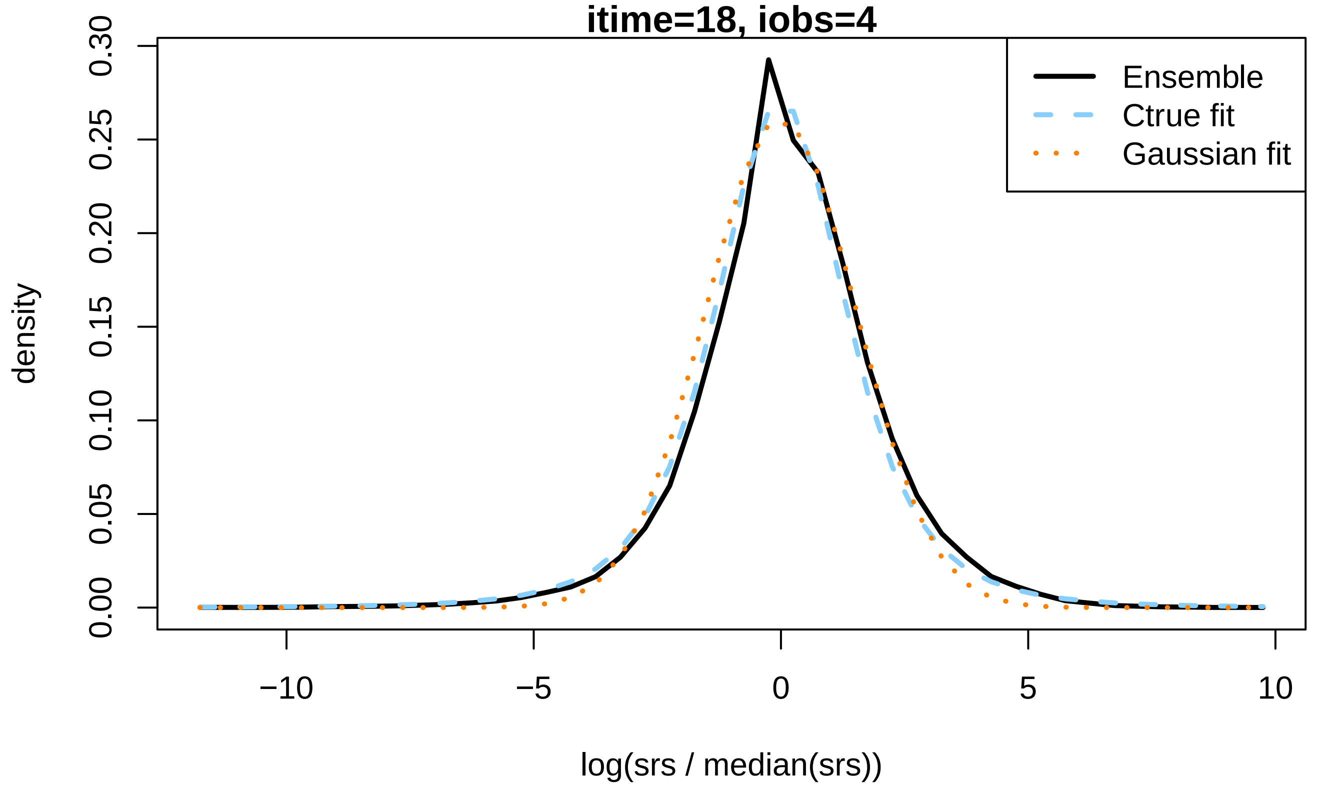

itime=18, iobs=1
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit



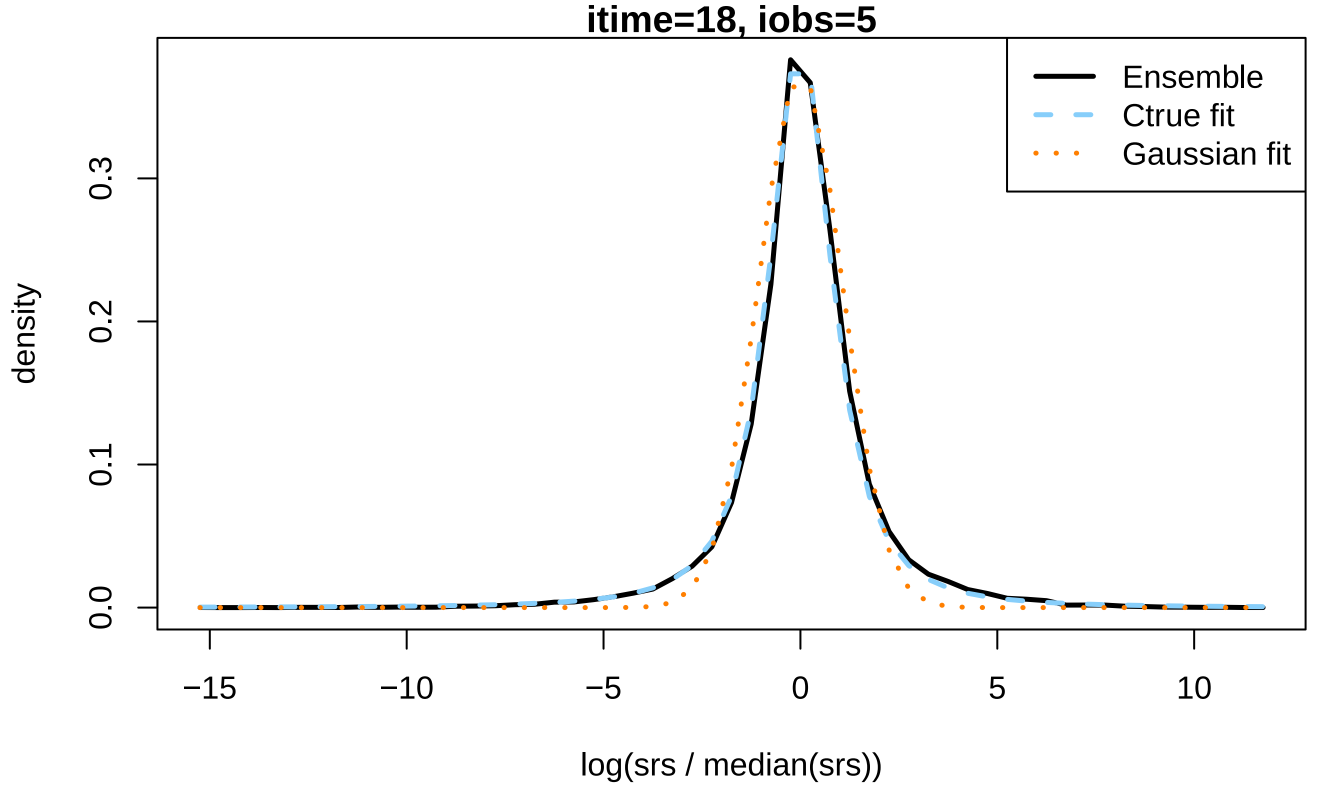
itime=18, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))

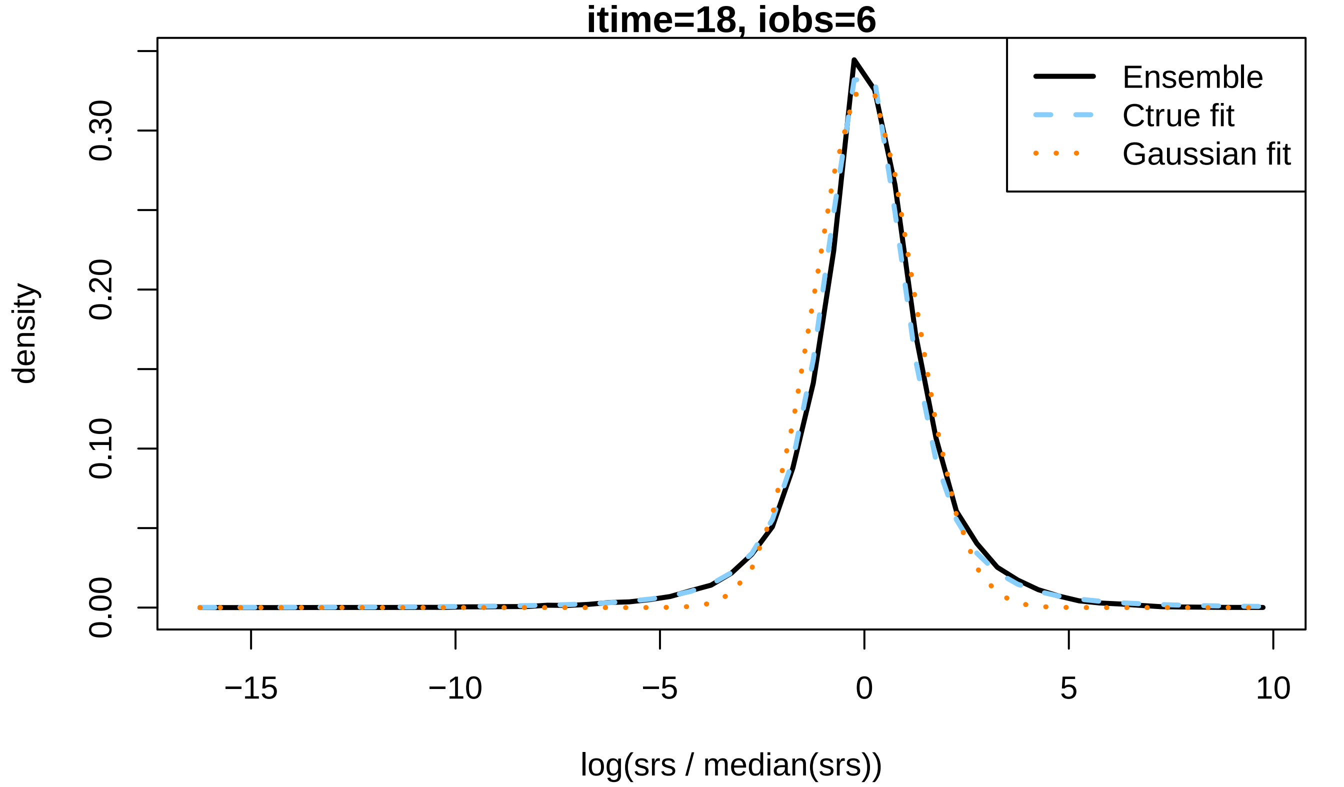


itime=18, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

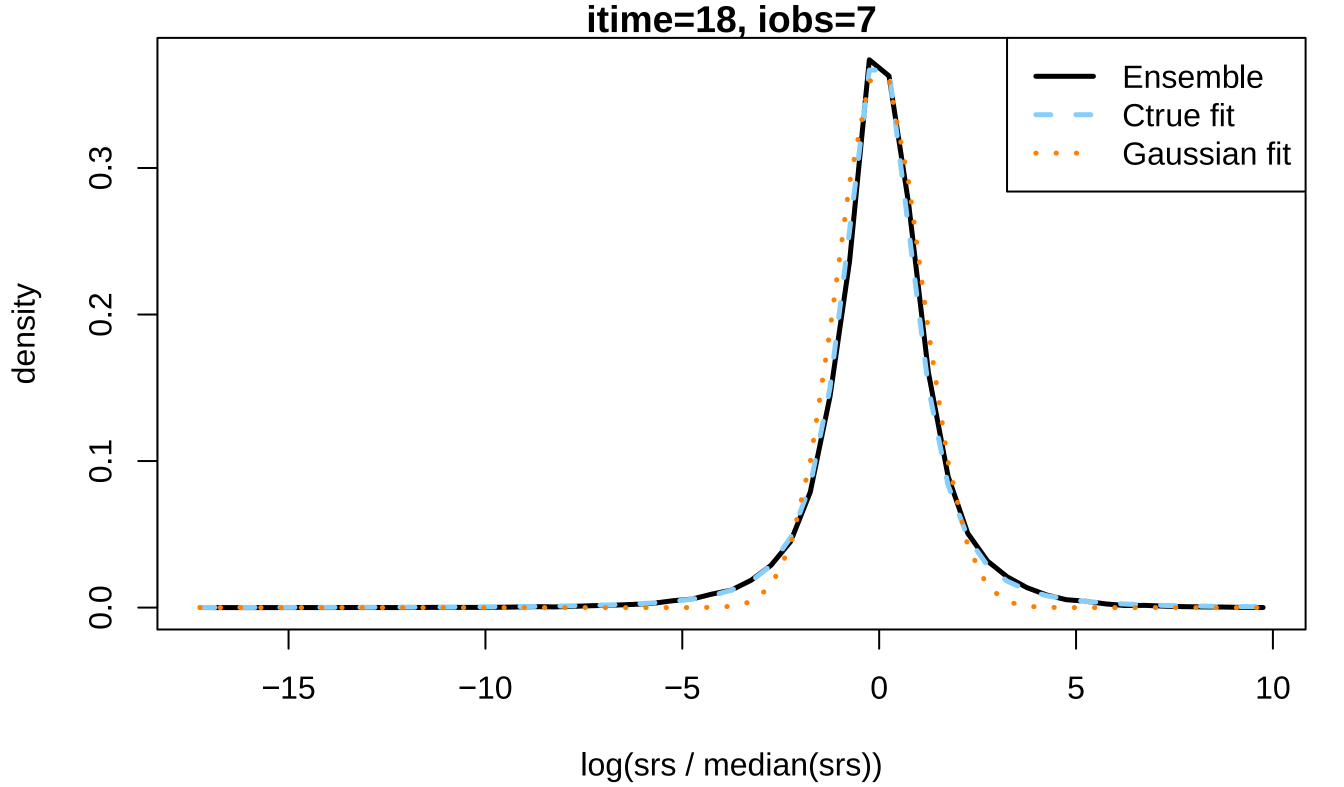


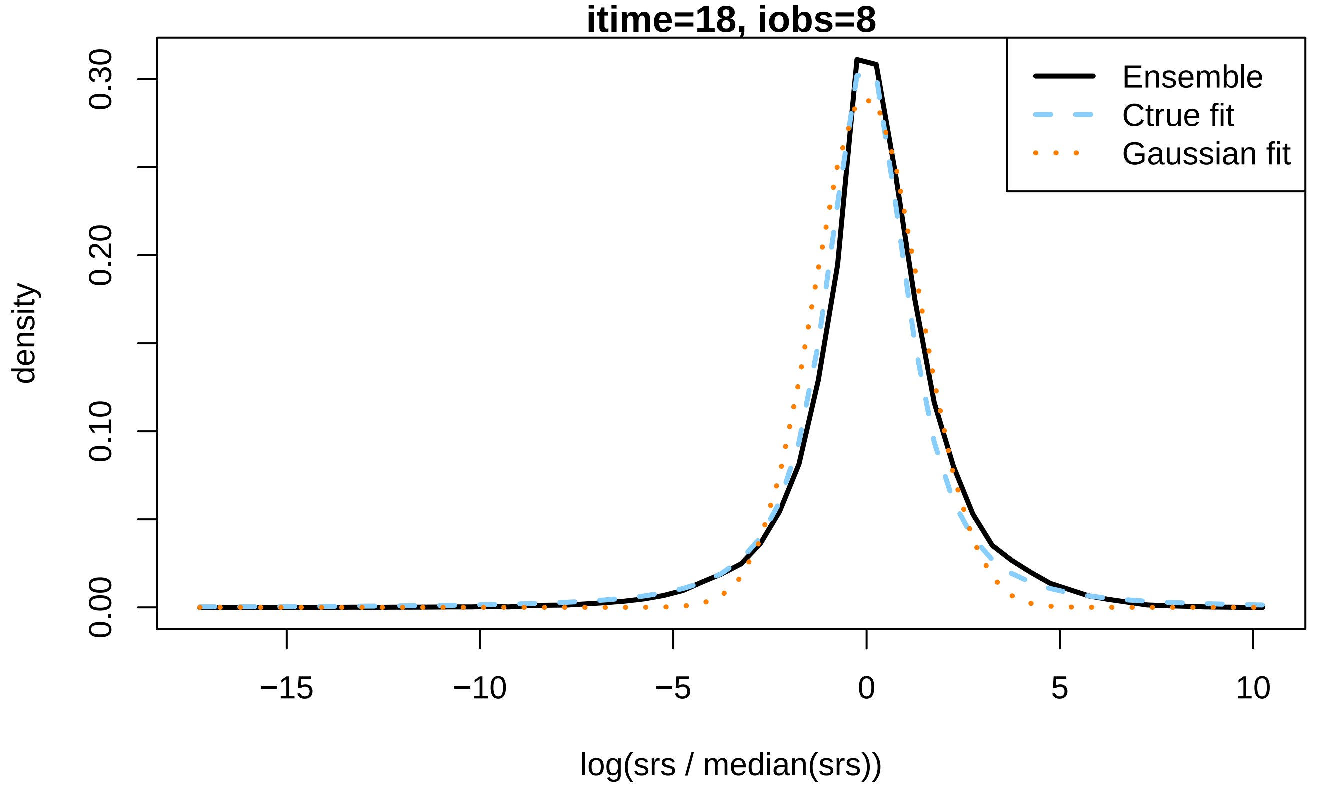
itime=18, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−10
−5
0
5
10
log(srs / median(srs))


itime=18, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

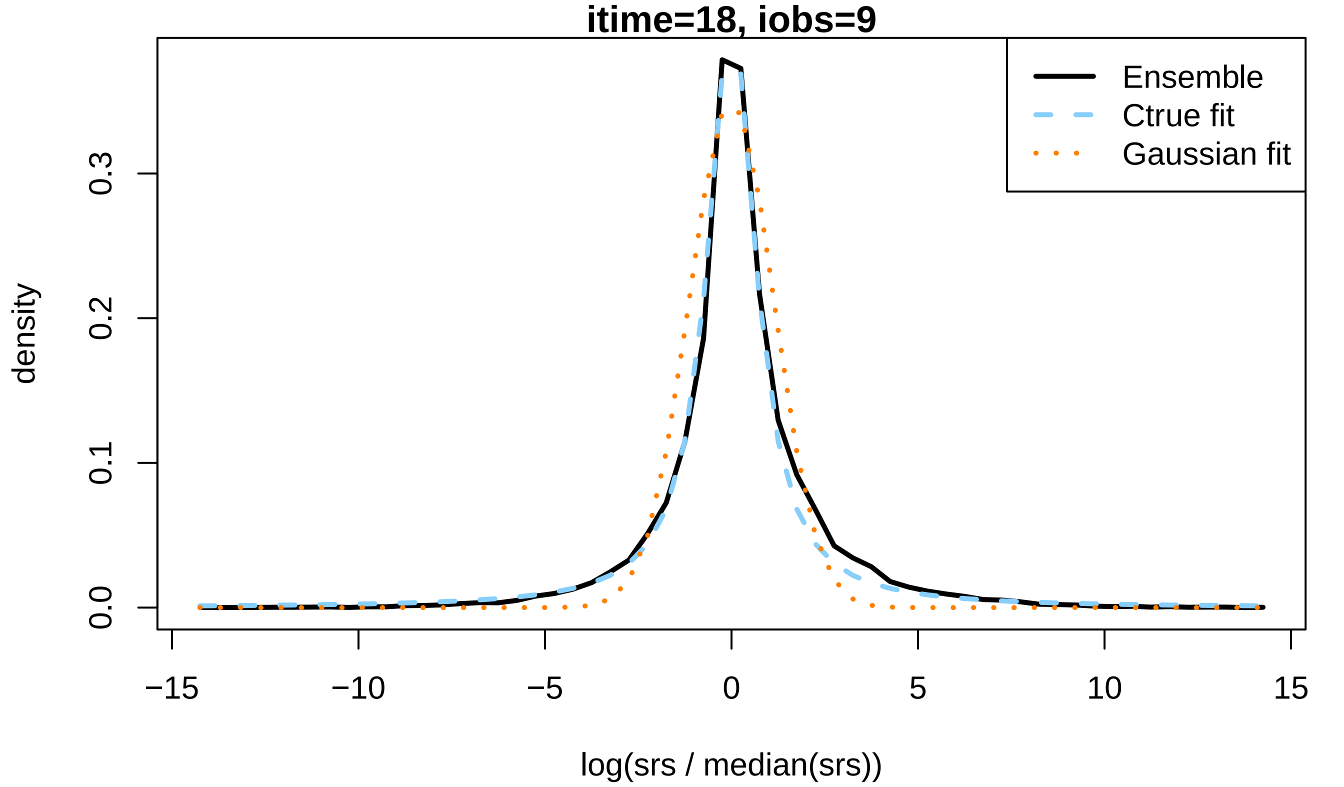


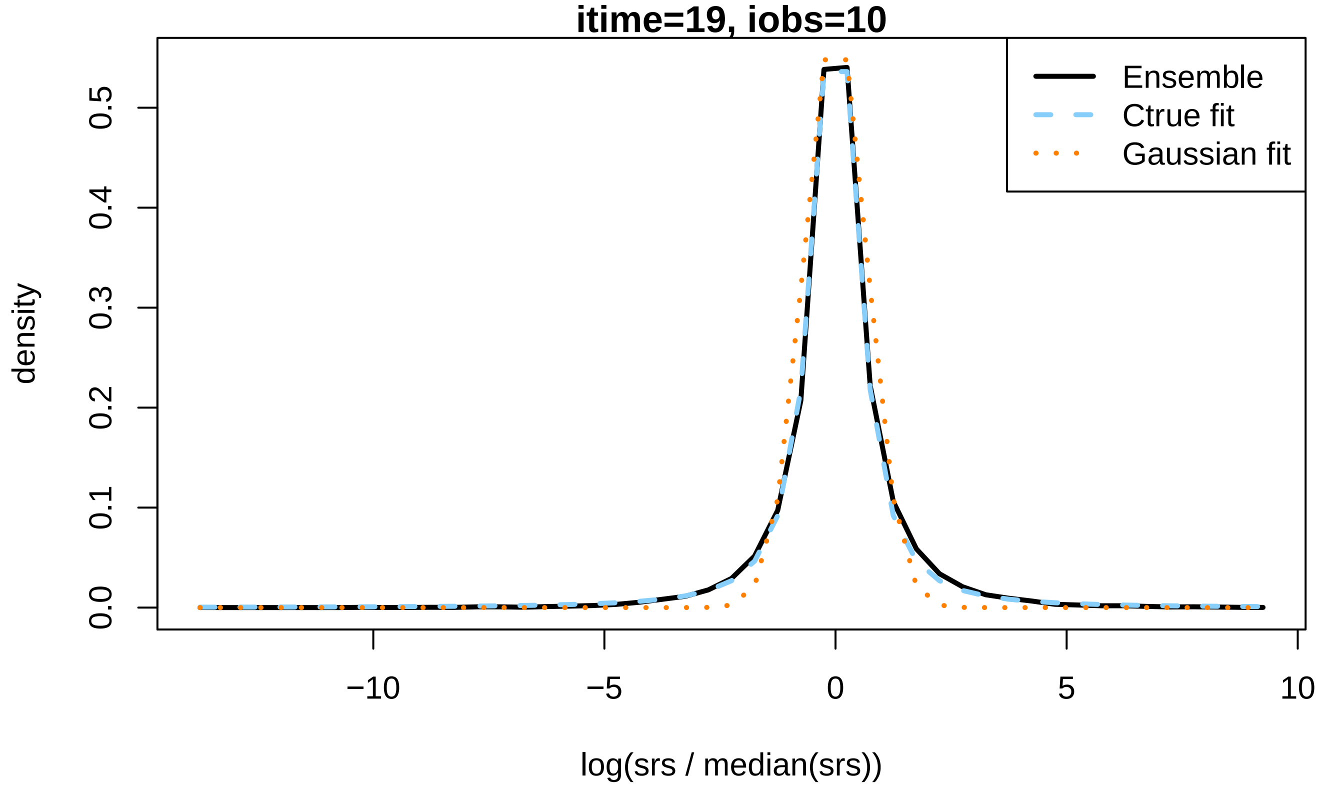
itime=18, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

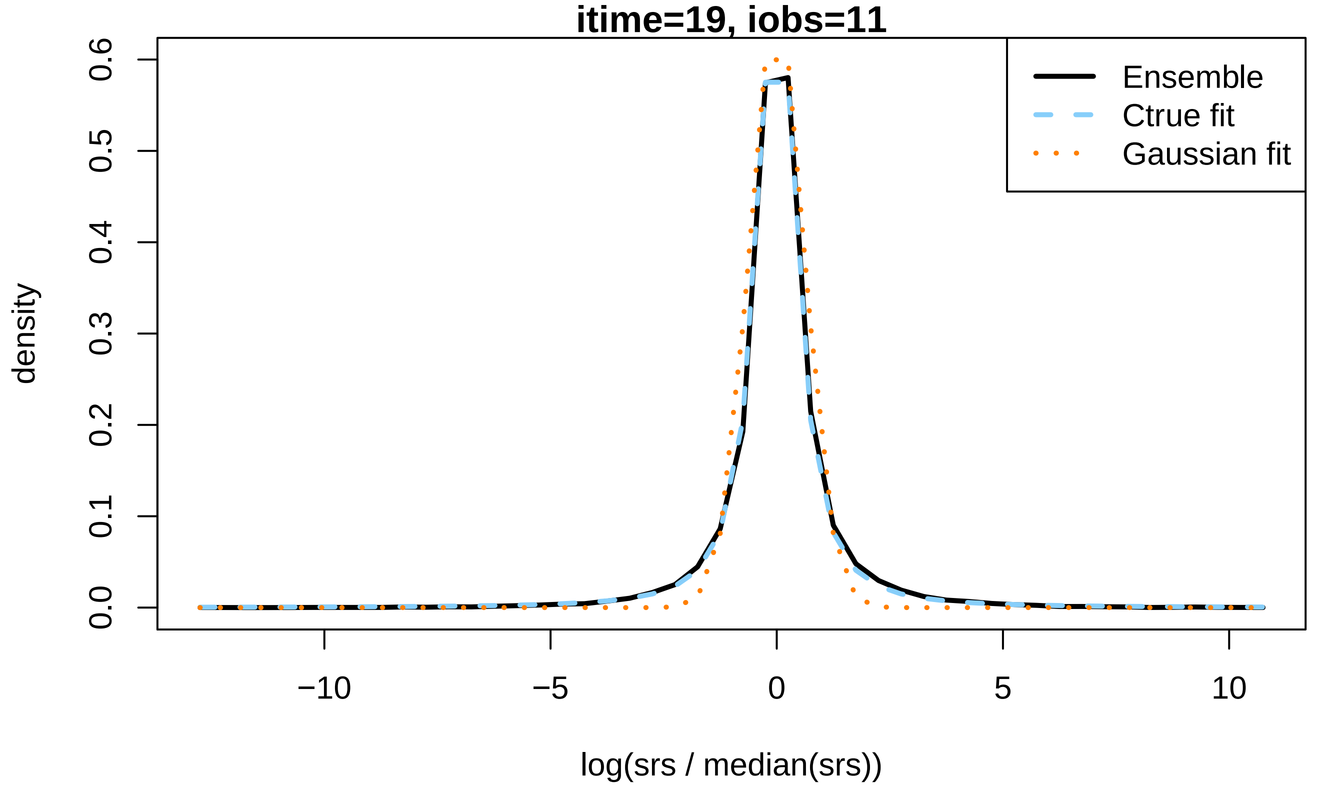

itime=18, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))



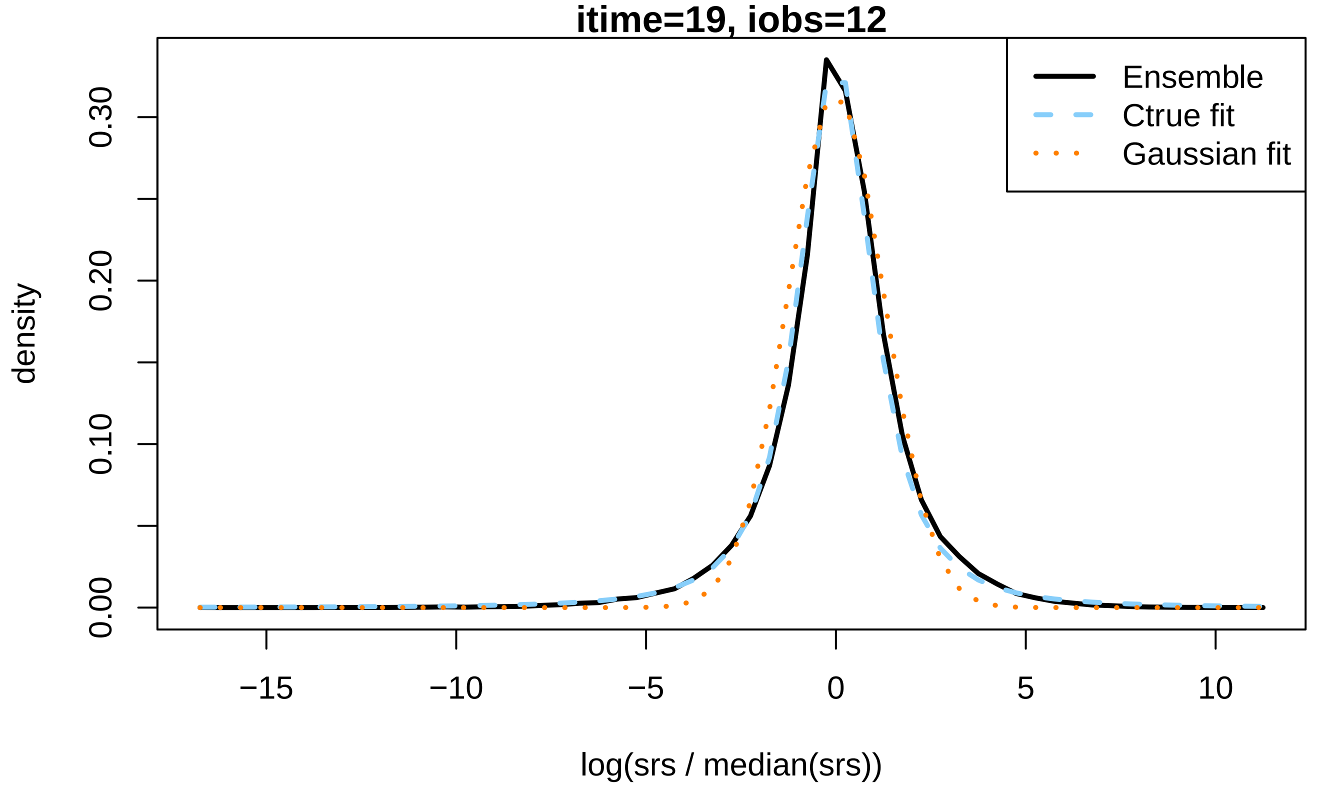
itime=18, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

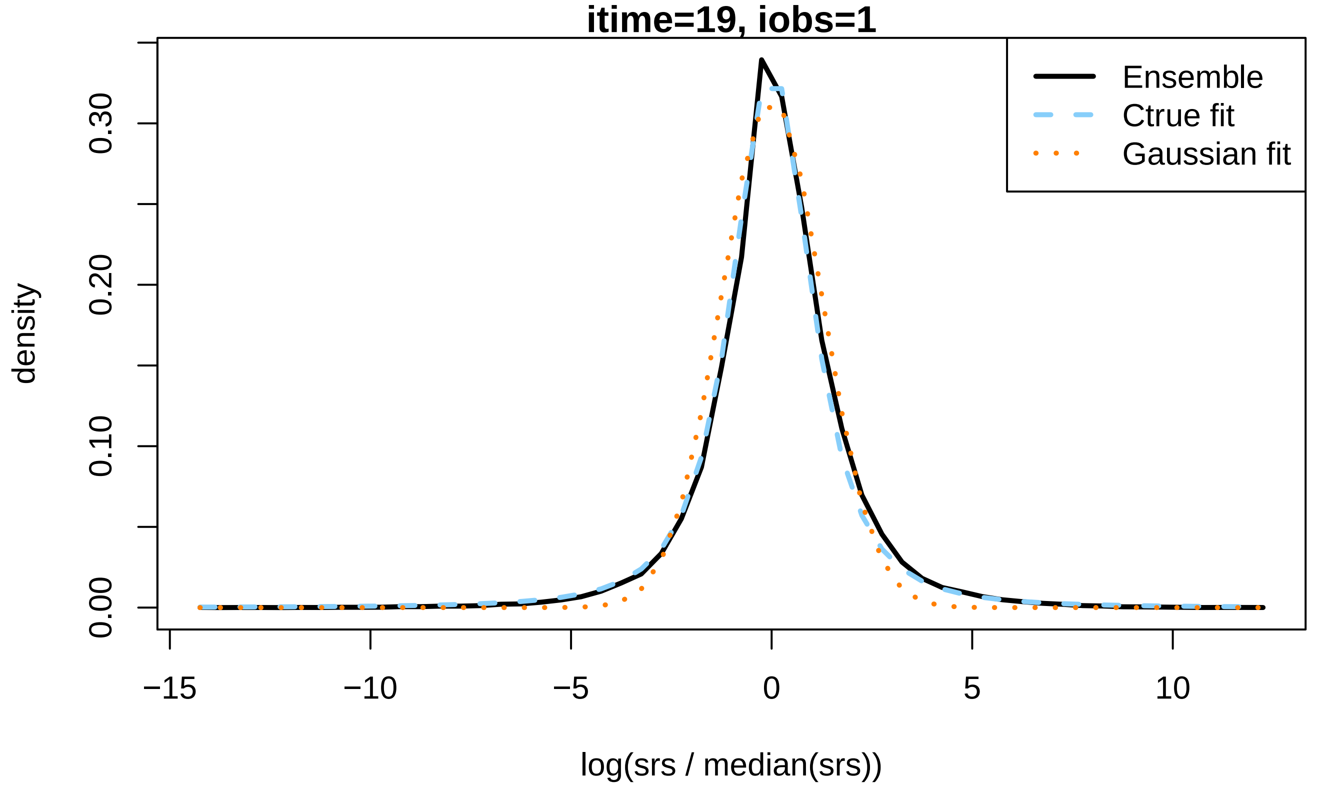

itime=18, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
15
log(srs / median(srs))

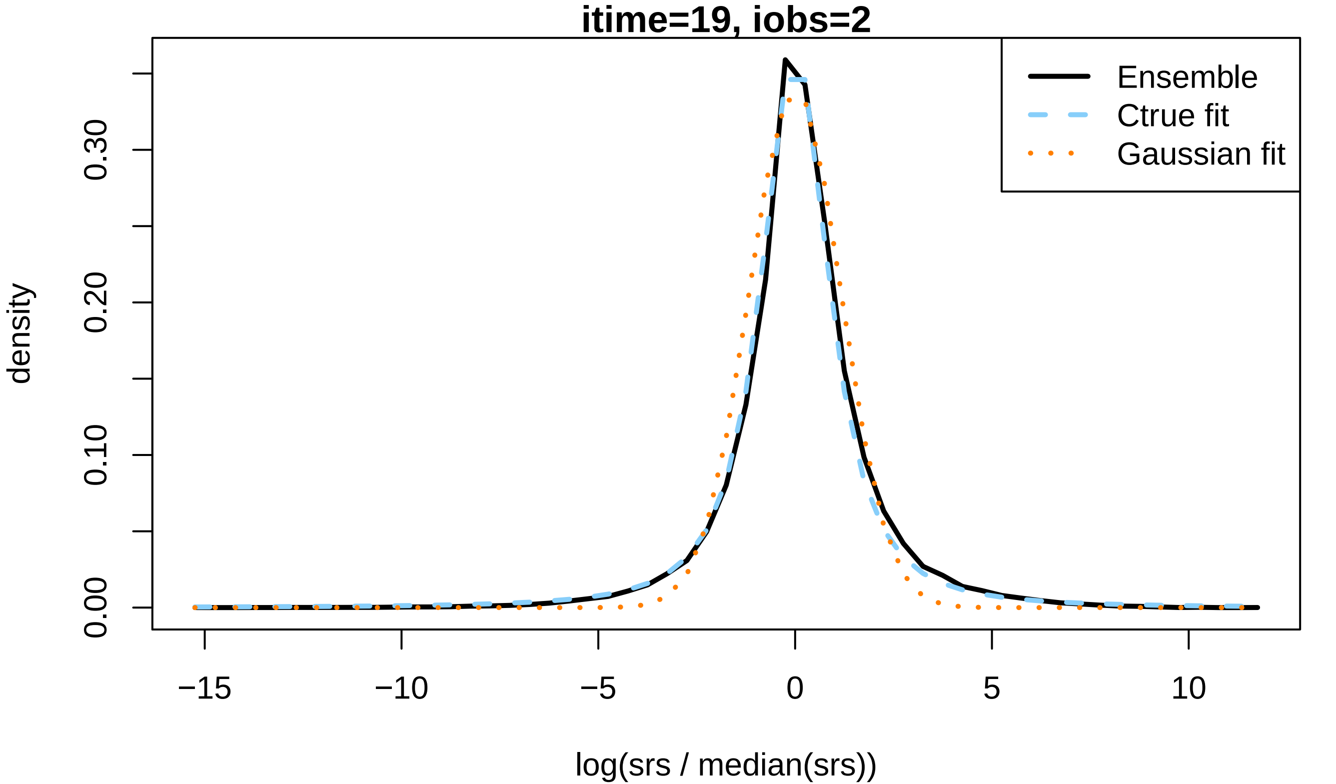

itime=19, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−10
−5
0
5
10
log(srs / median(srs))

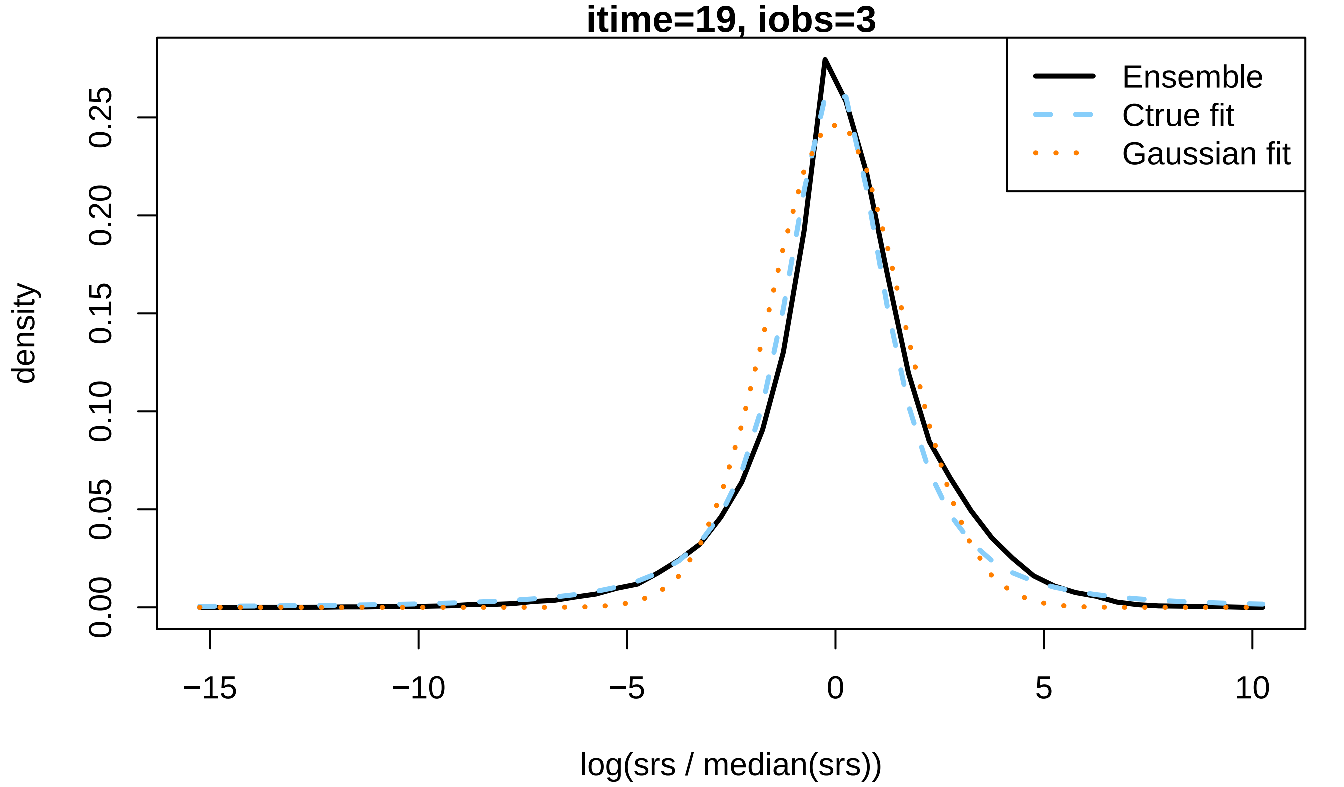


itime=19, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−10
−5
0
5
10
log(srs / median(srs))

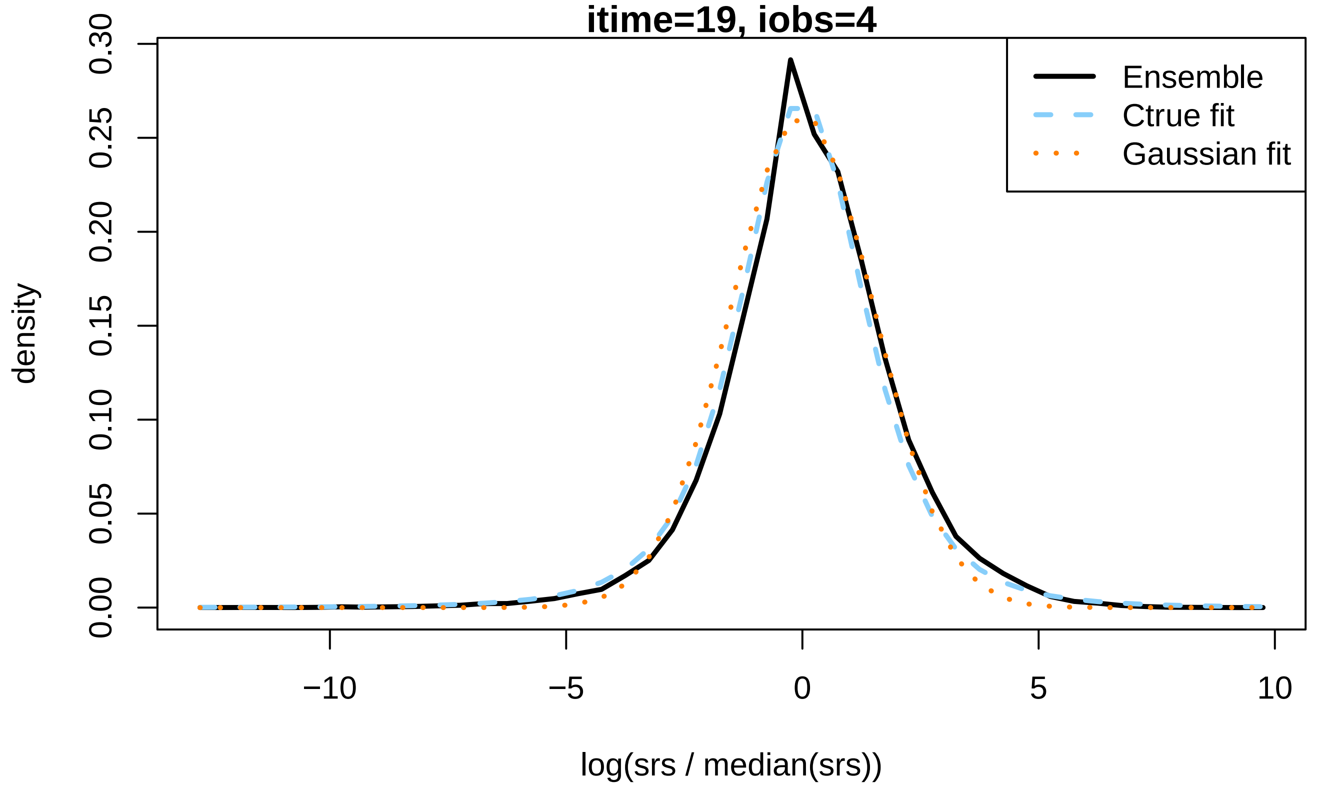

itime=19, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=19, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

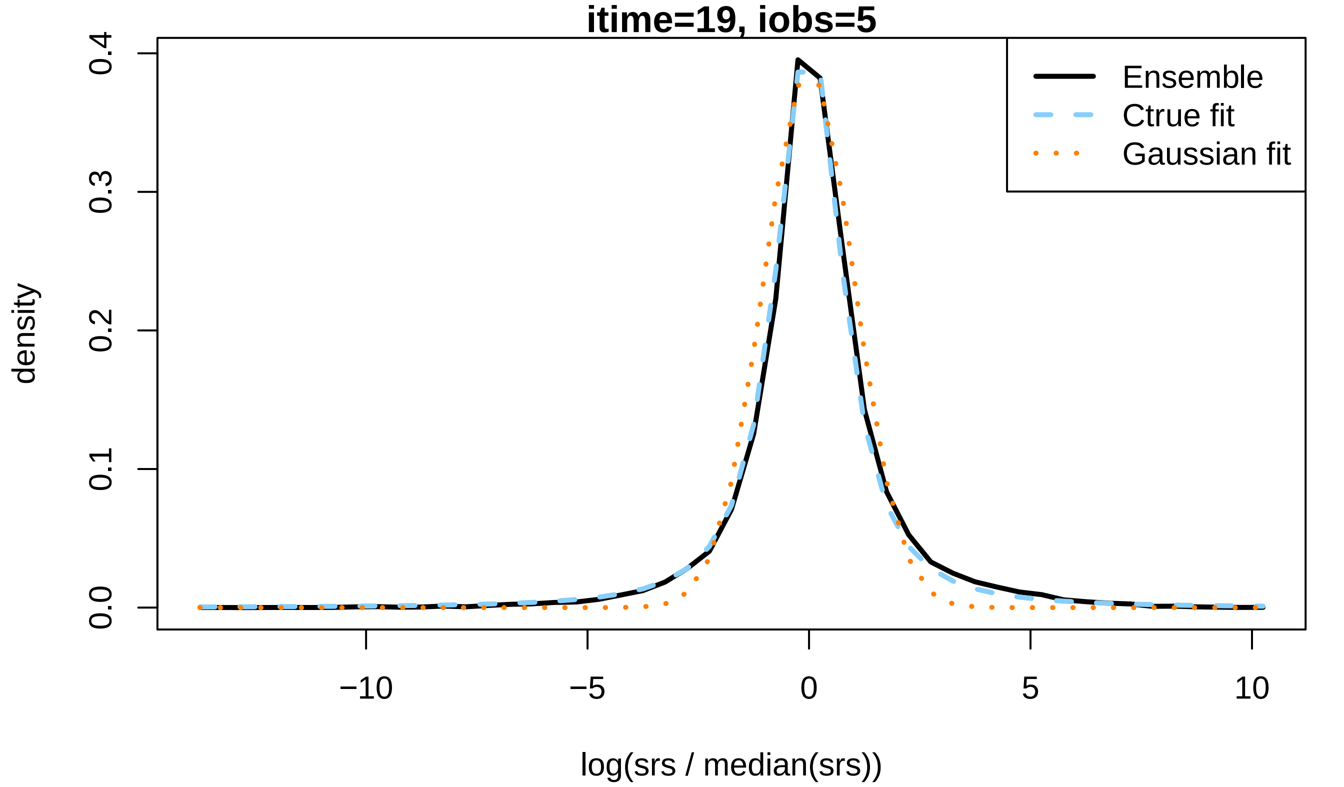

itime=19, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

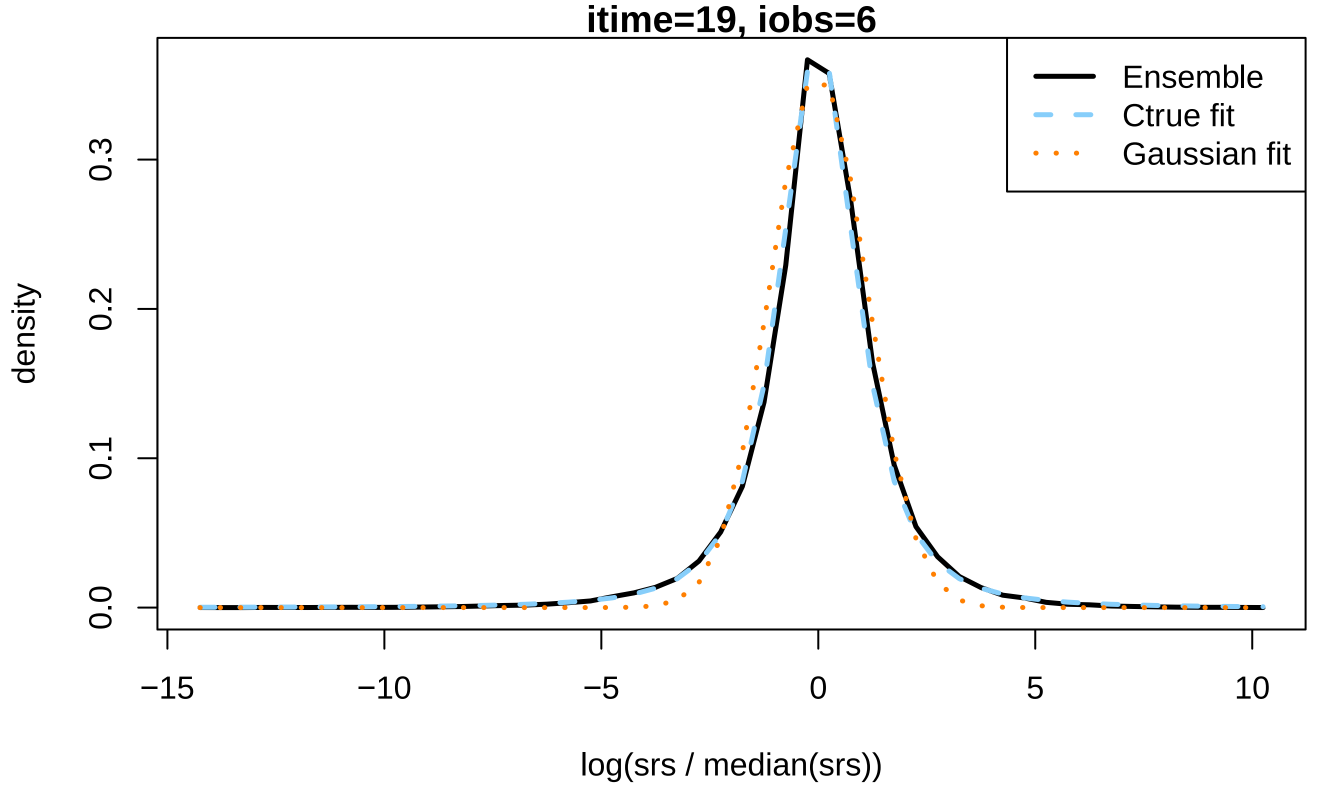


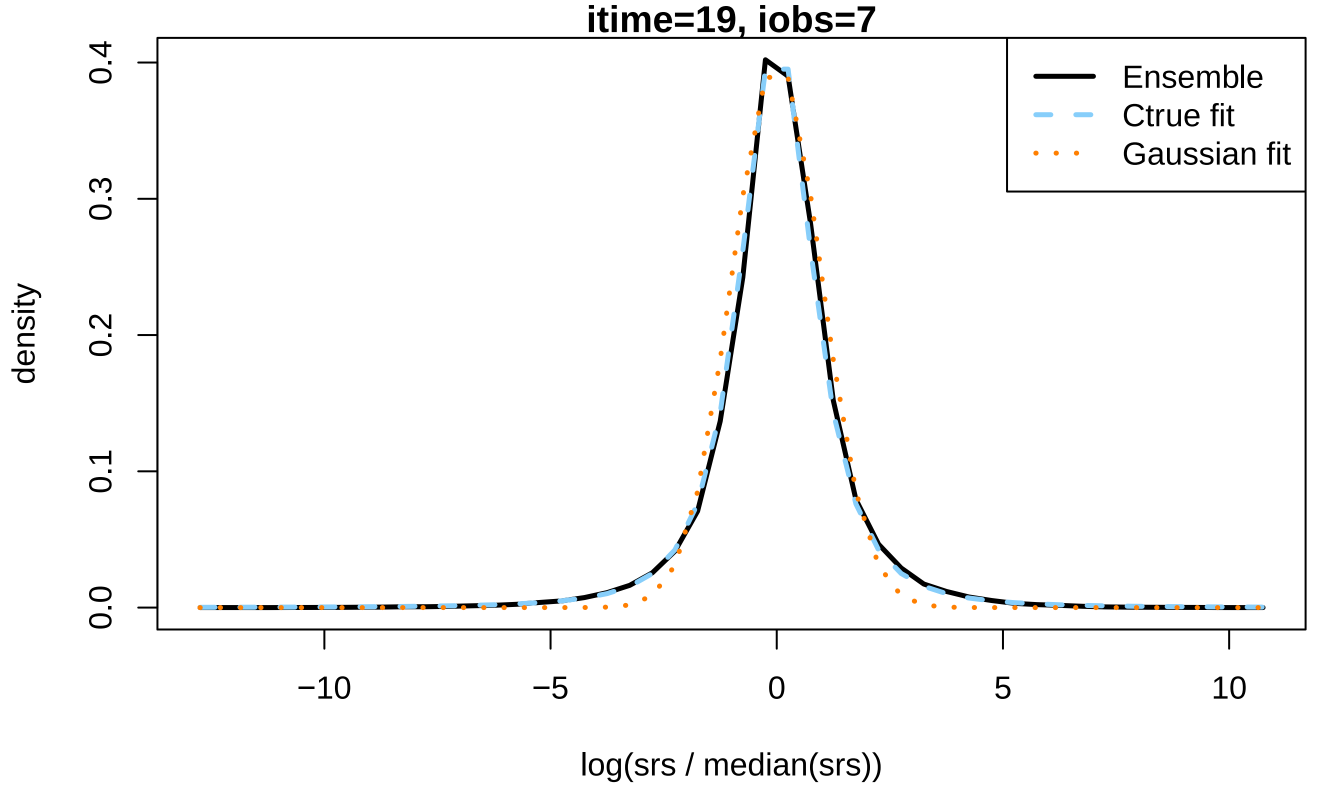
itime=19, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

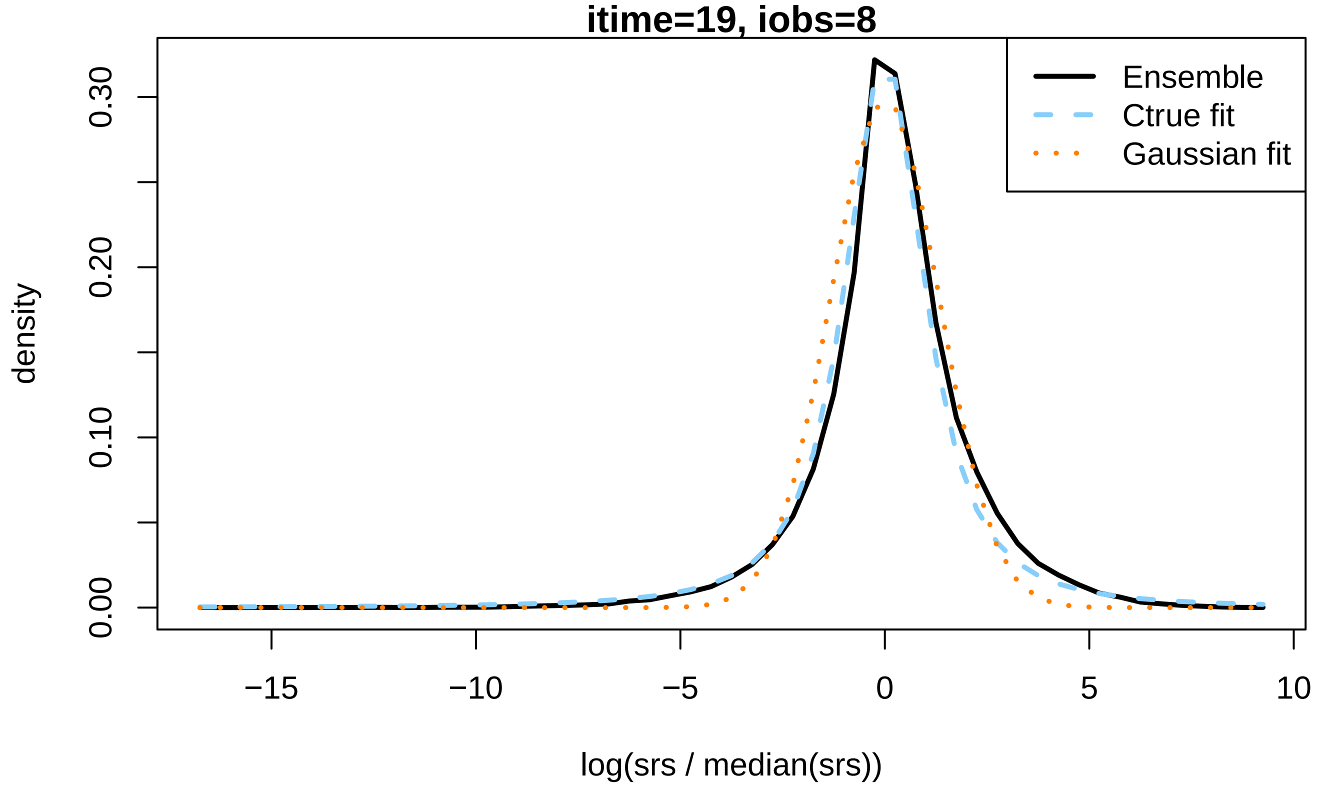


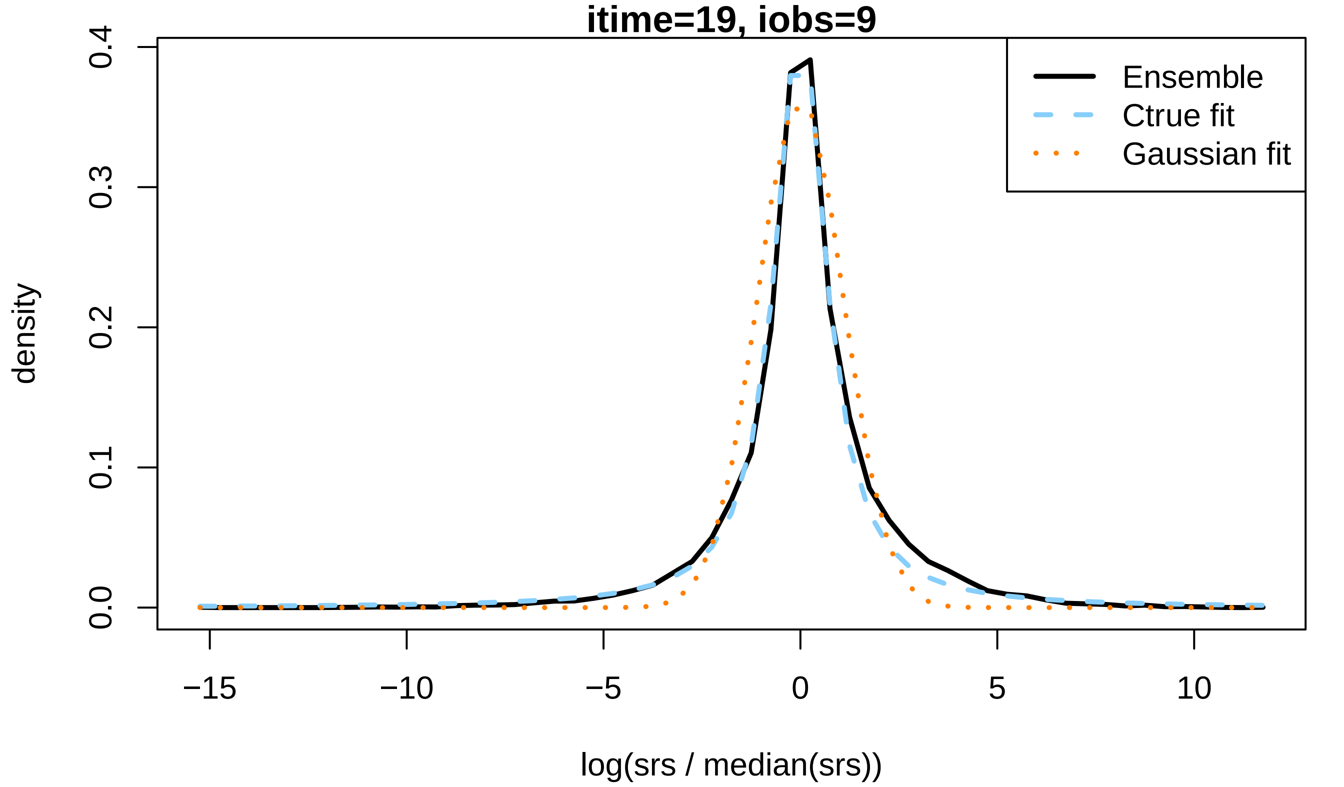
itime=19, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−10
−5
0
5
10
log(srs / median(srs))

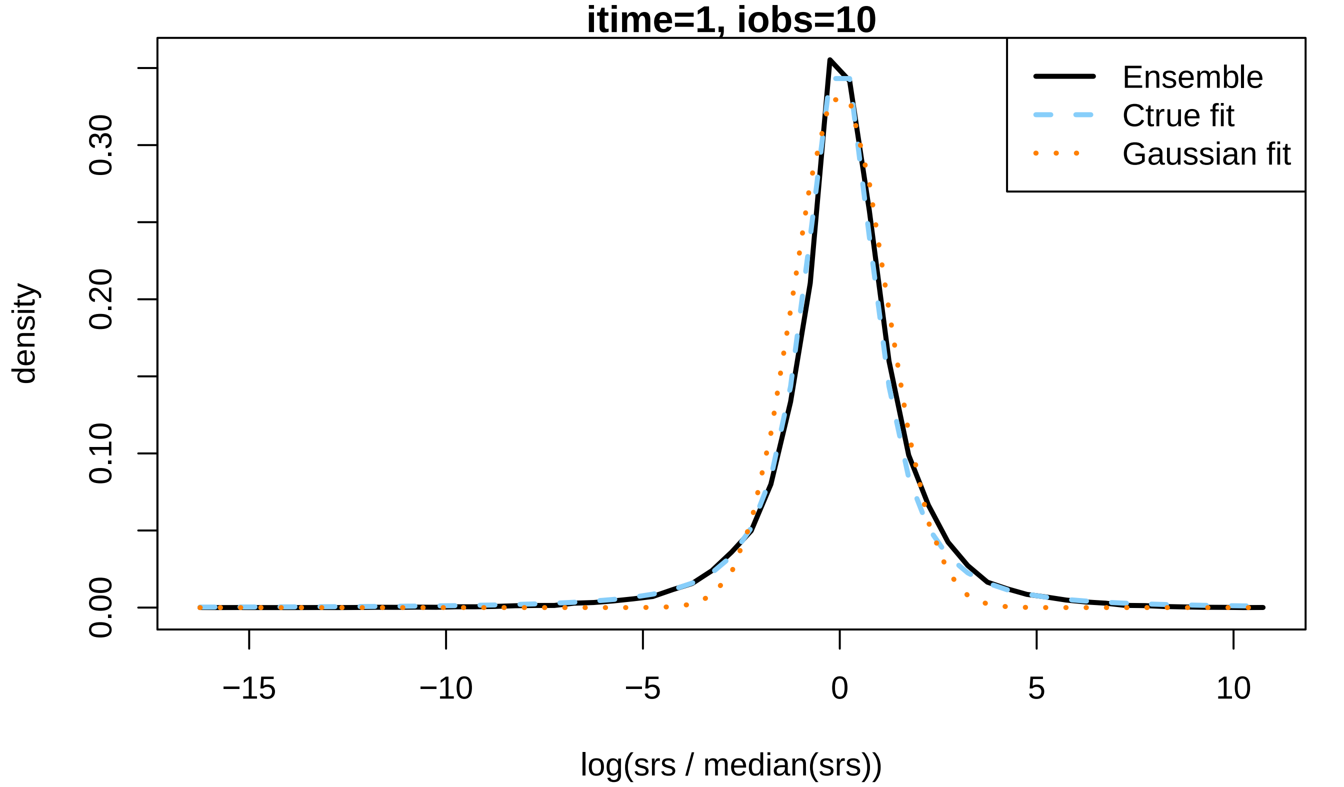

itime=19, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
log(srs / median(srs))


itime=19, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

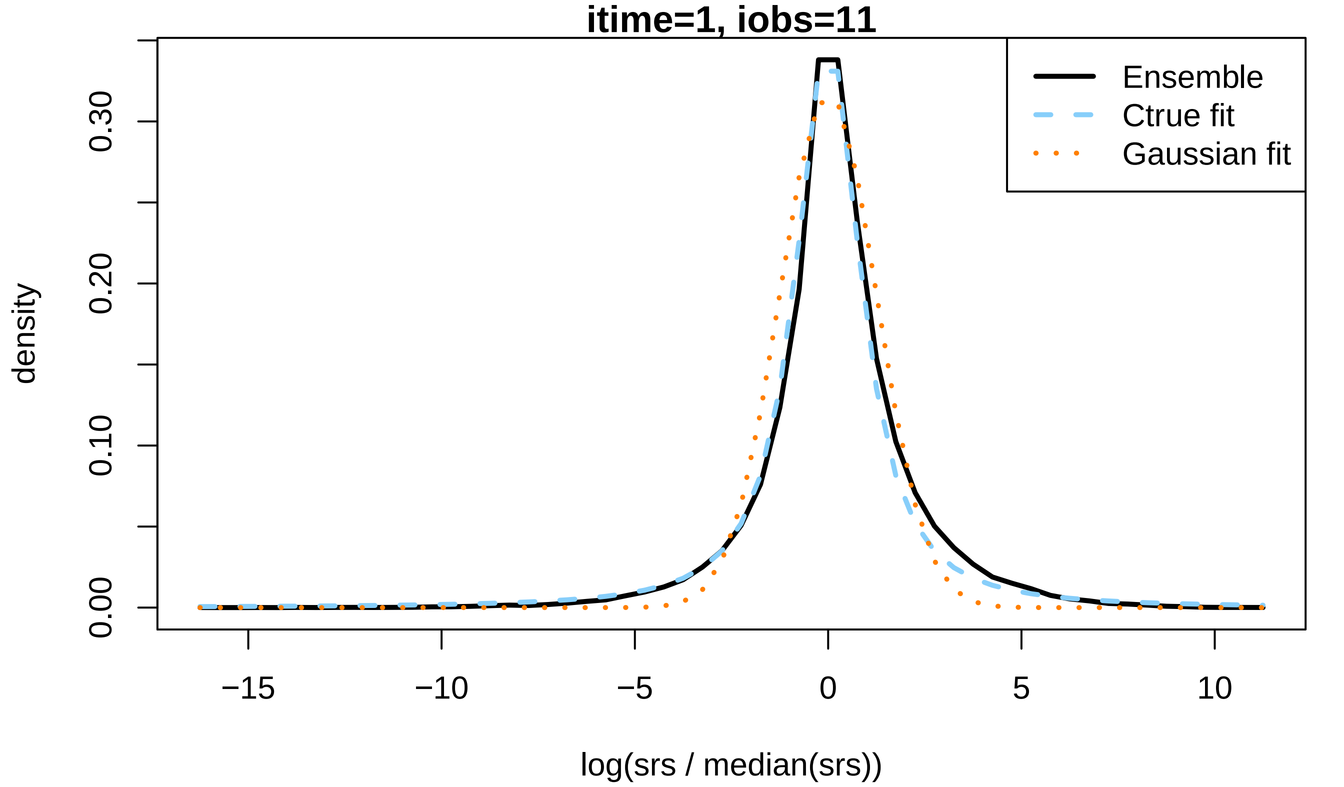

itime=19, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
log(srs / median(srs))


itime=19, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

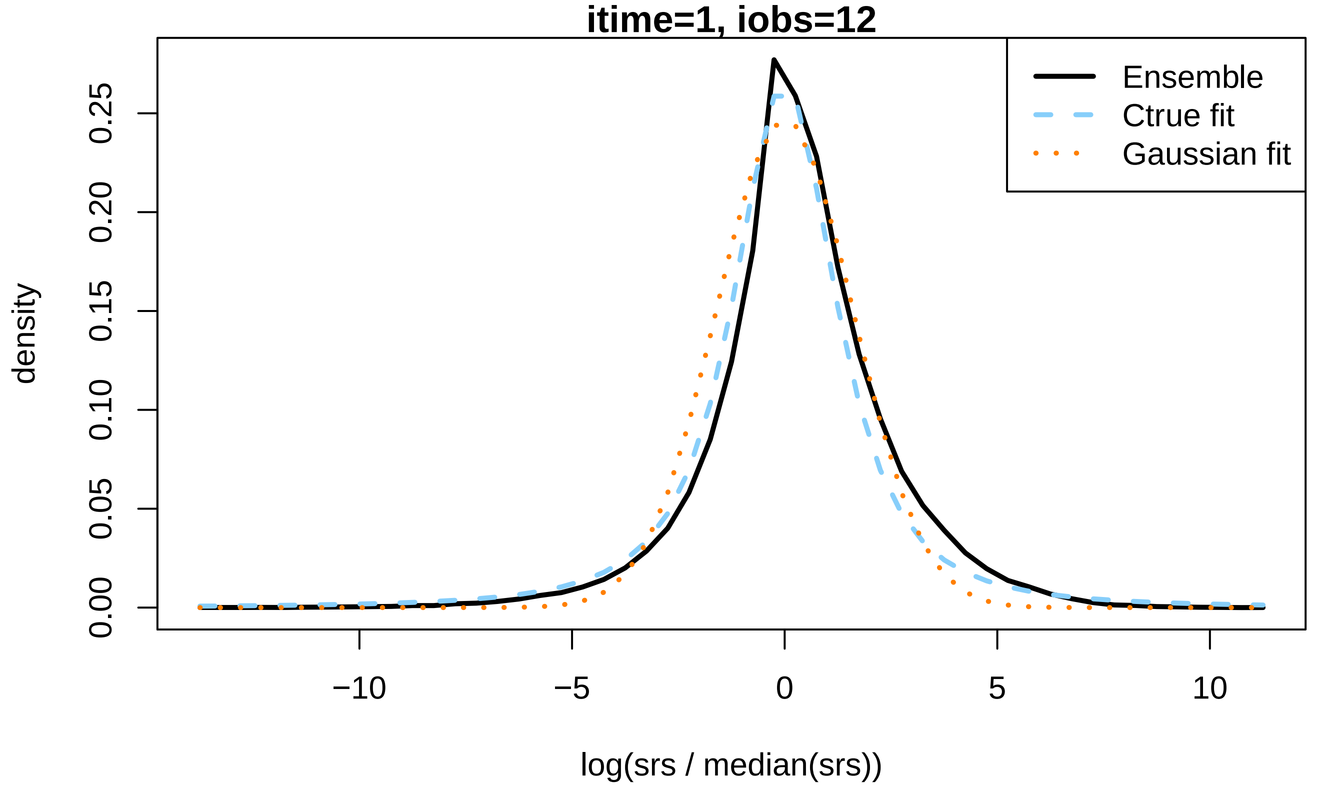

itime=19, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))



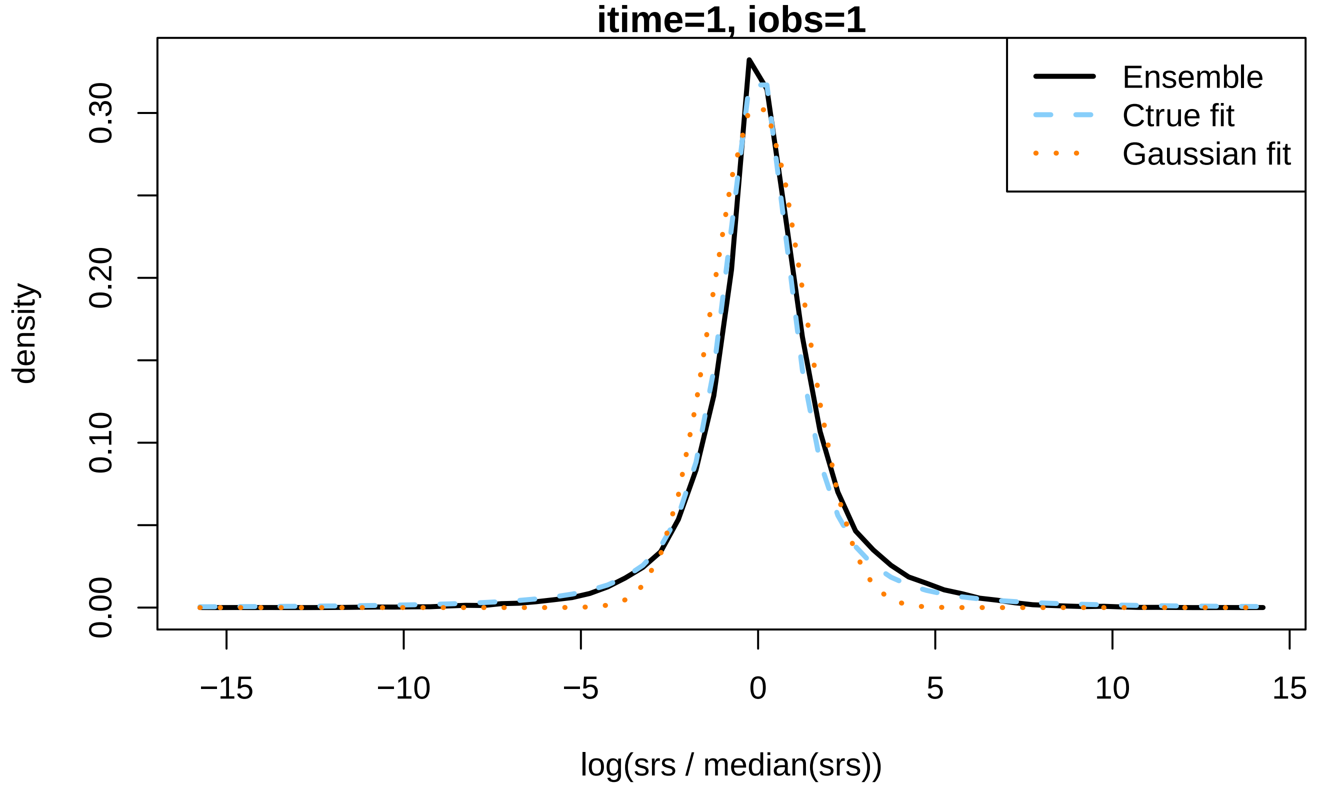
itime=1, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

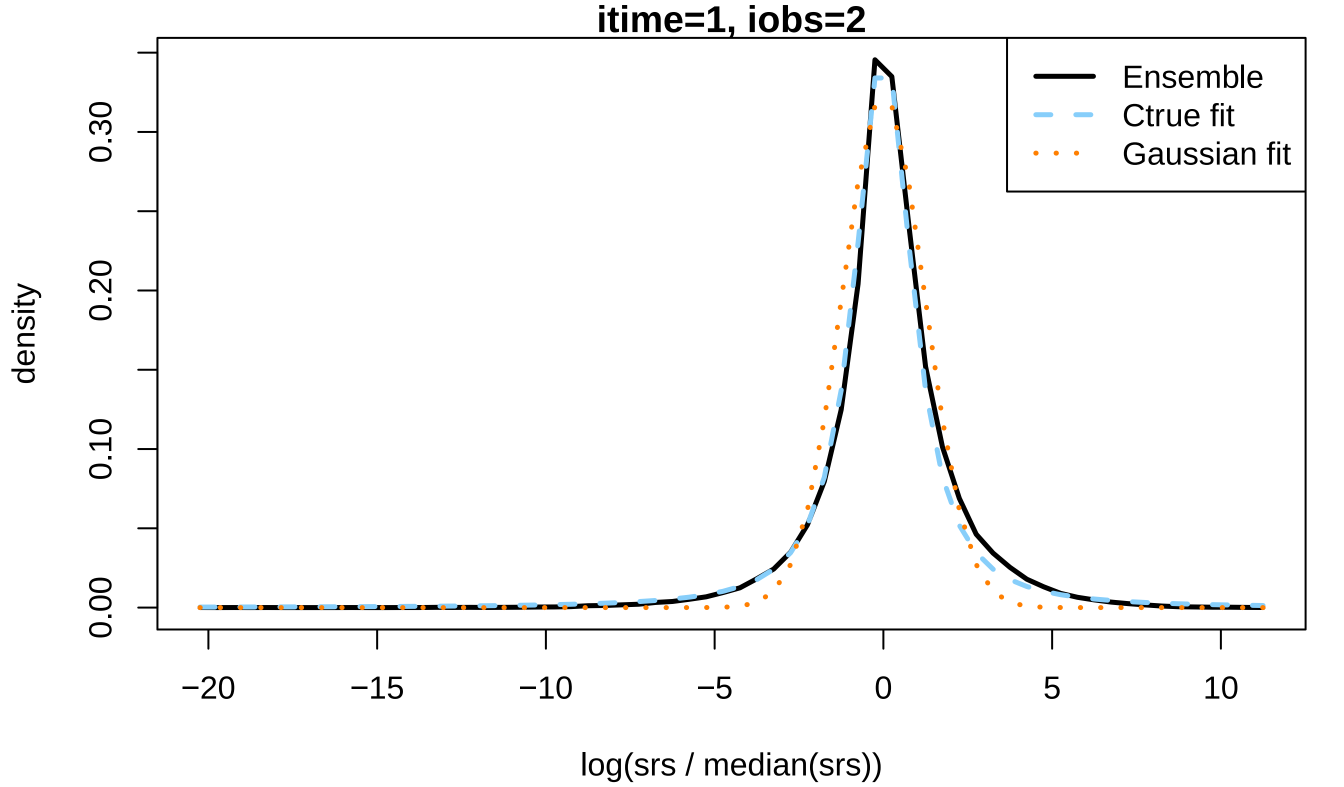


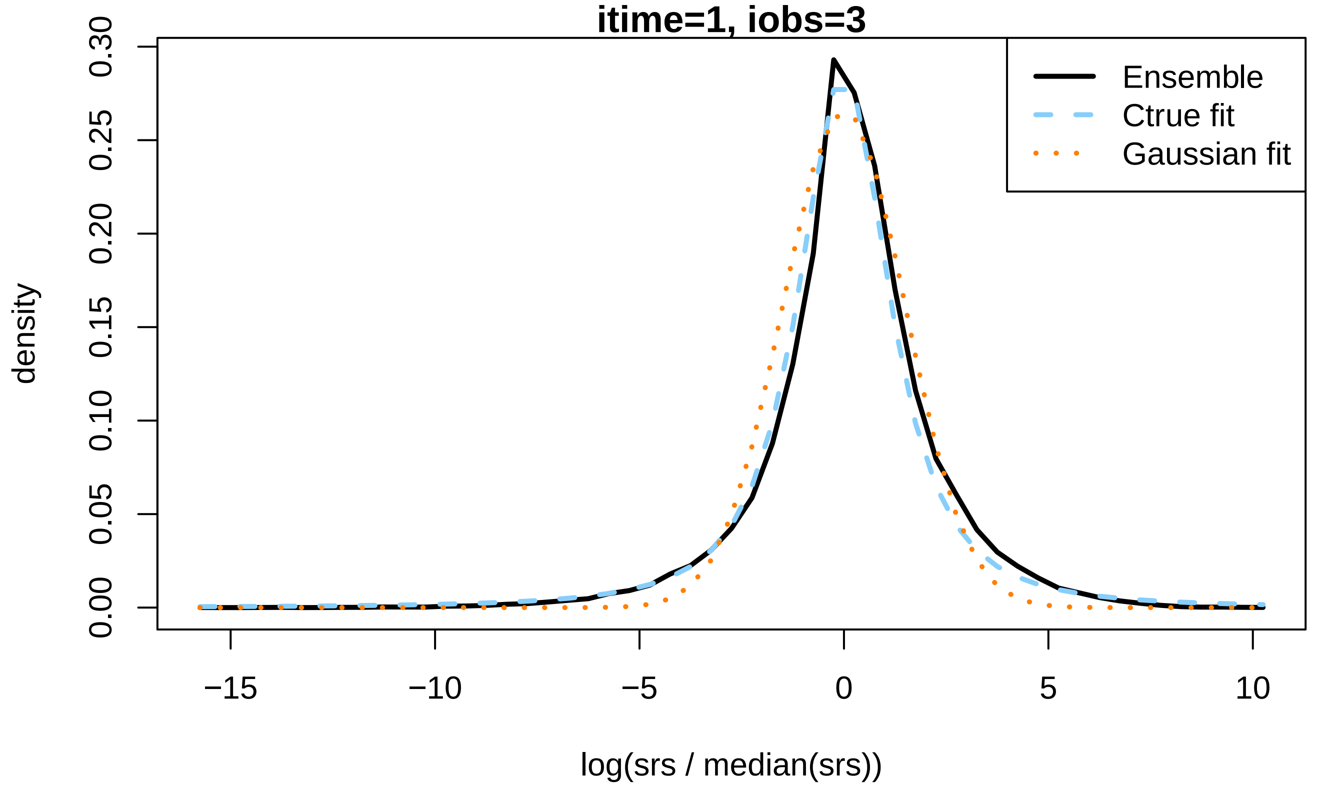
itime=1, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



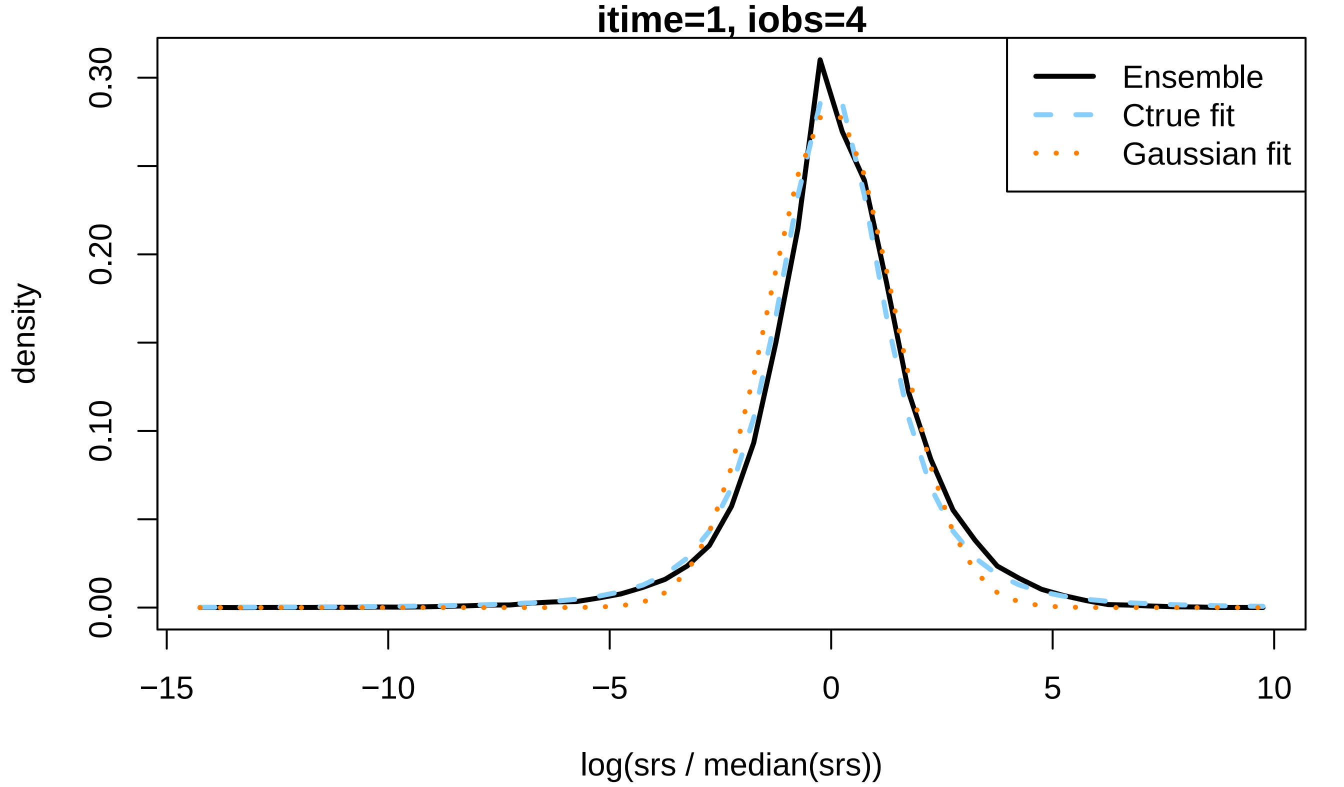
itime=1, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−10
−5
0
5
10
log(srs / median(srs))


itime=1, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
15
log(srs / median(srs))

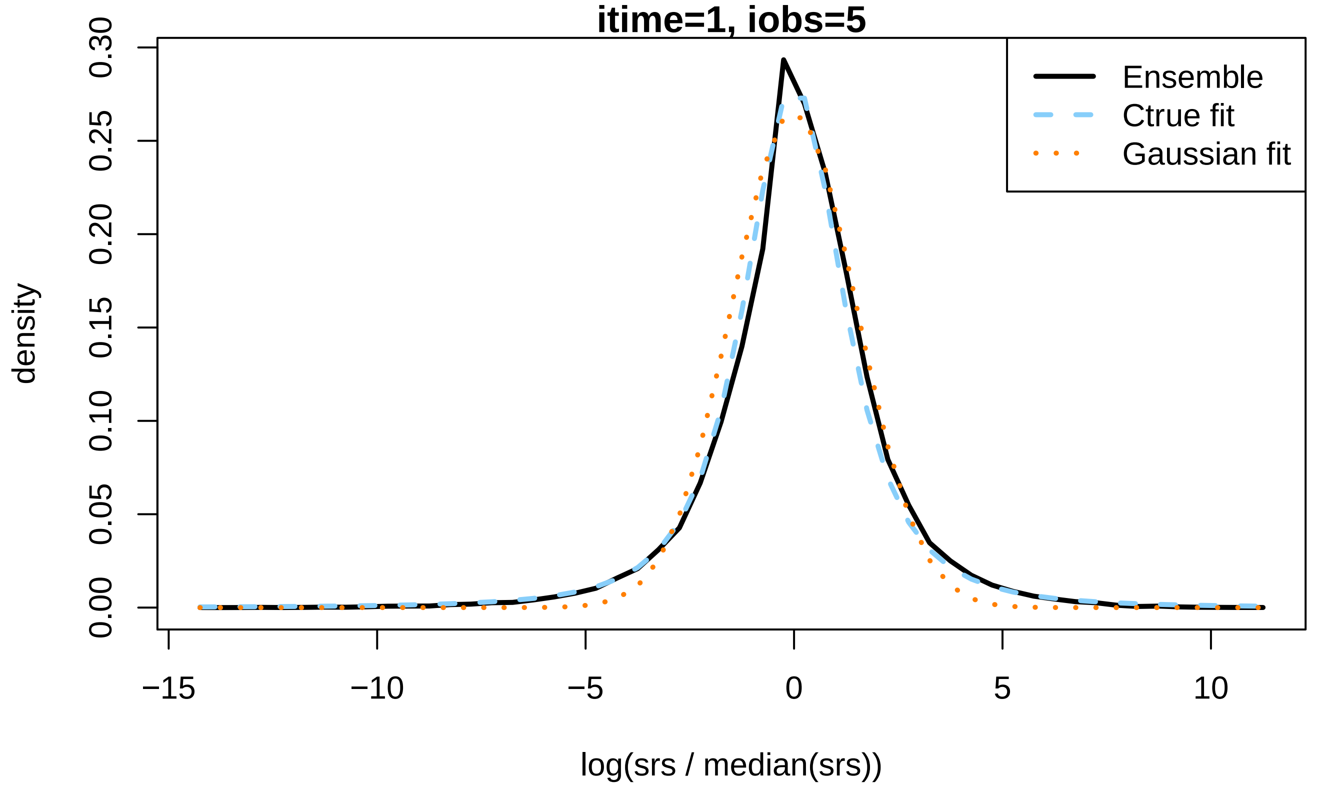

itime=1, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−20
−15
−10
−5
0
5
10
log(srs / median(srs))



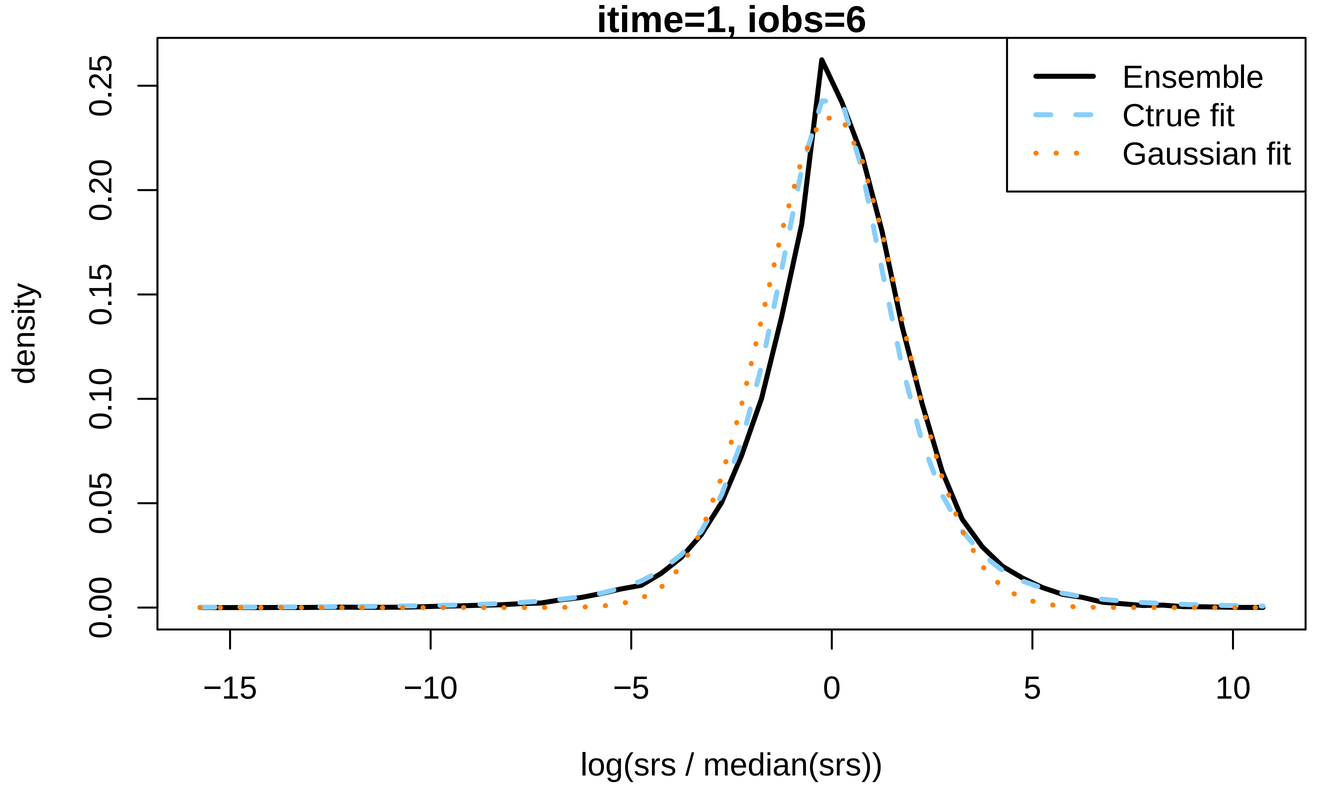
itime=1, iobs=3
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

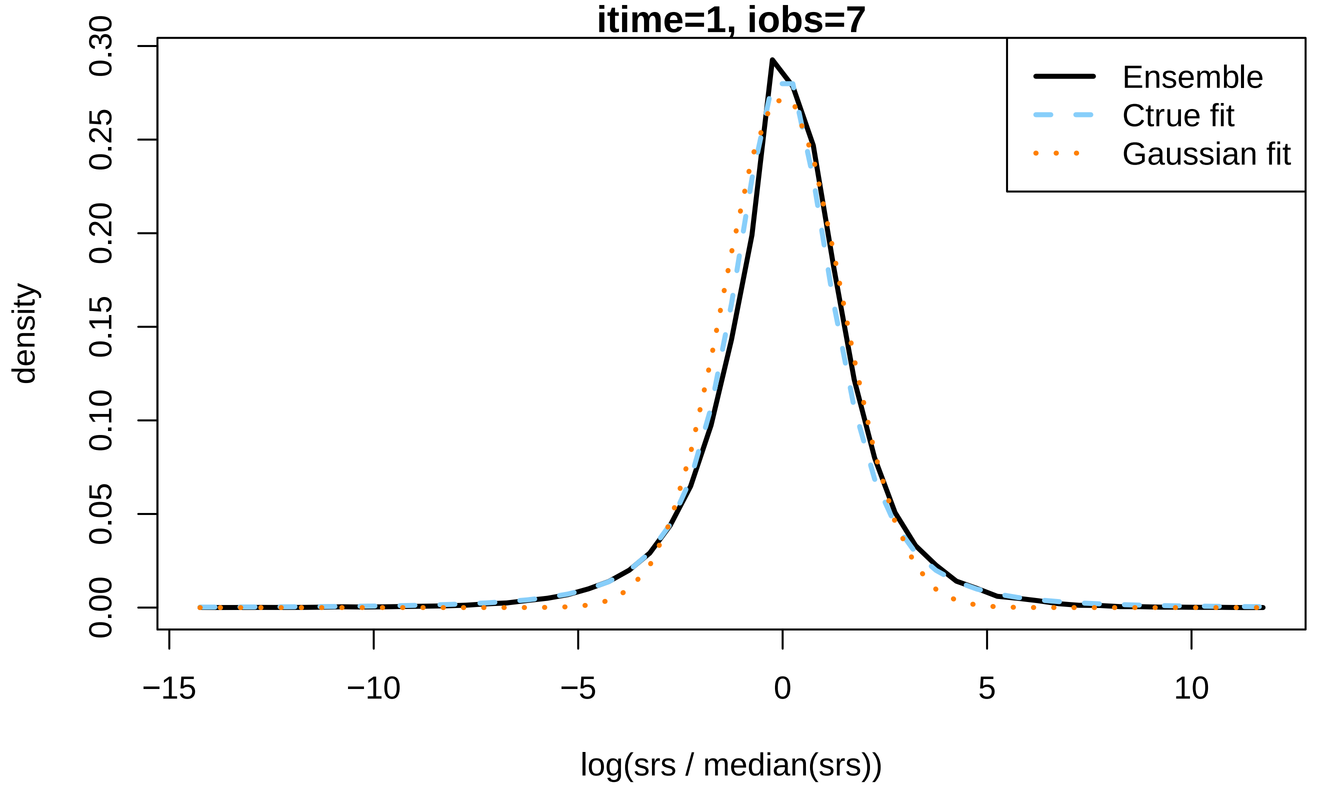


itime=1, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

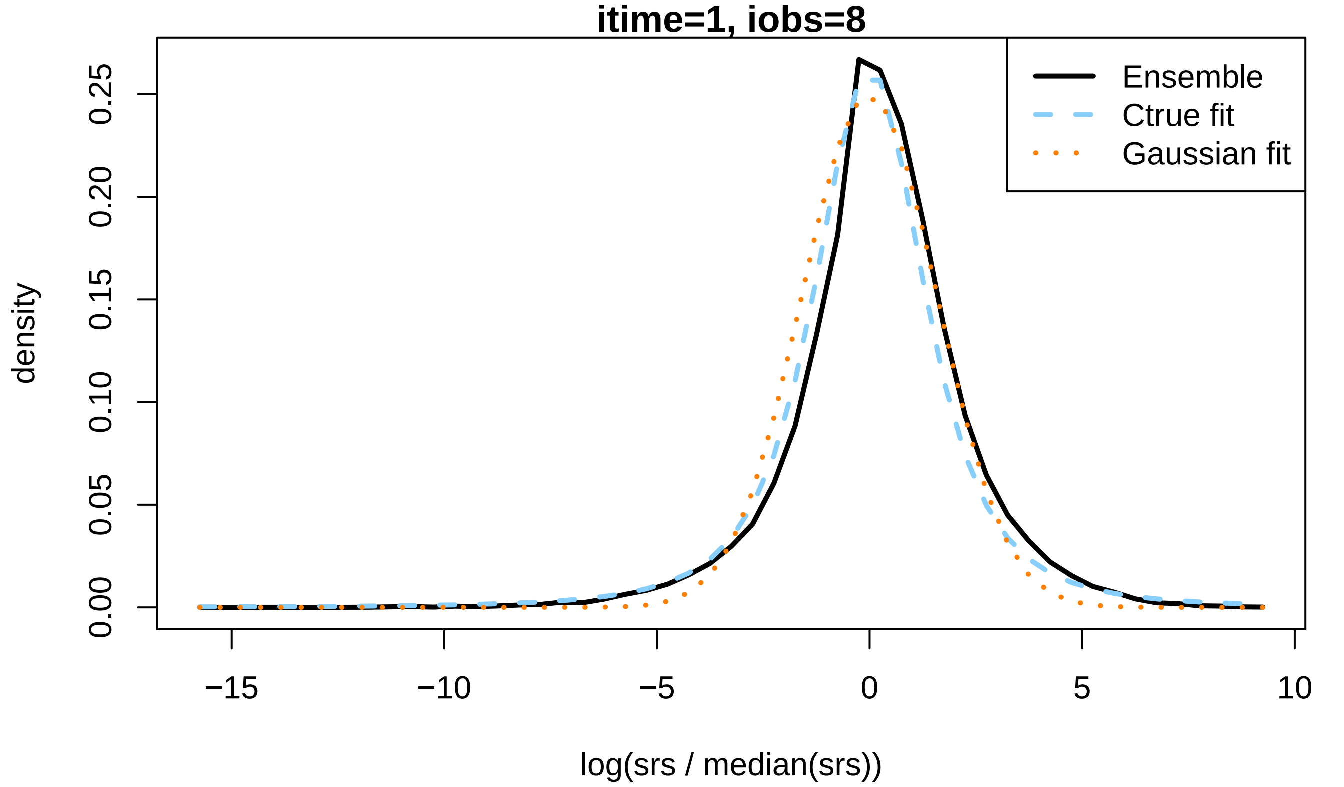


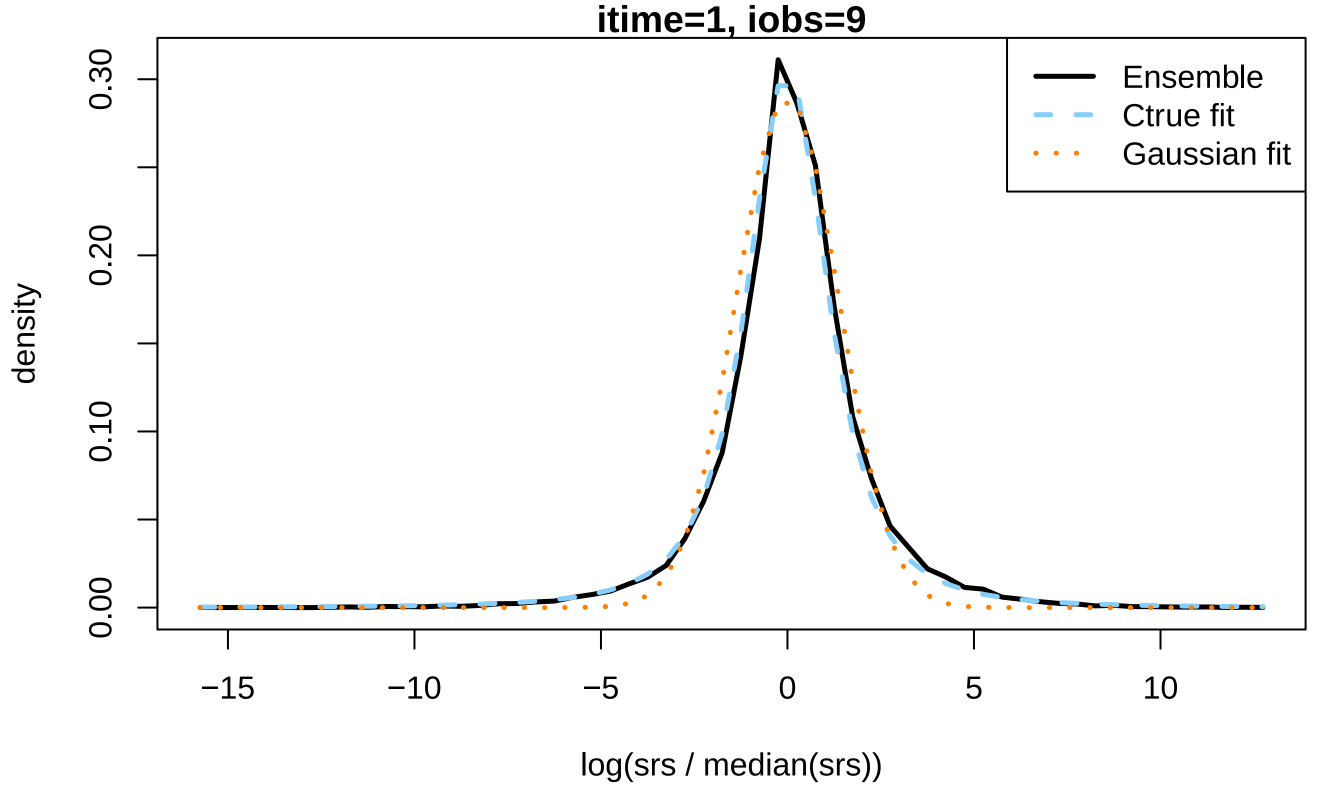
itime=1, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

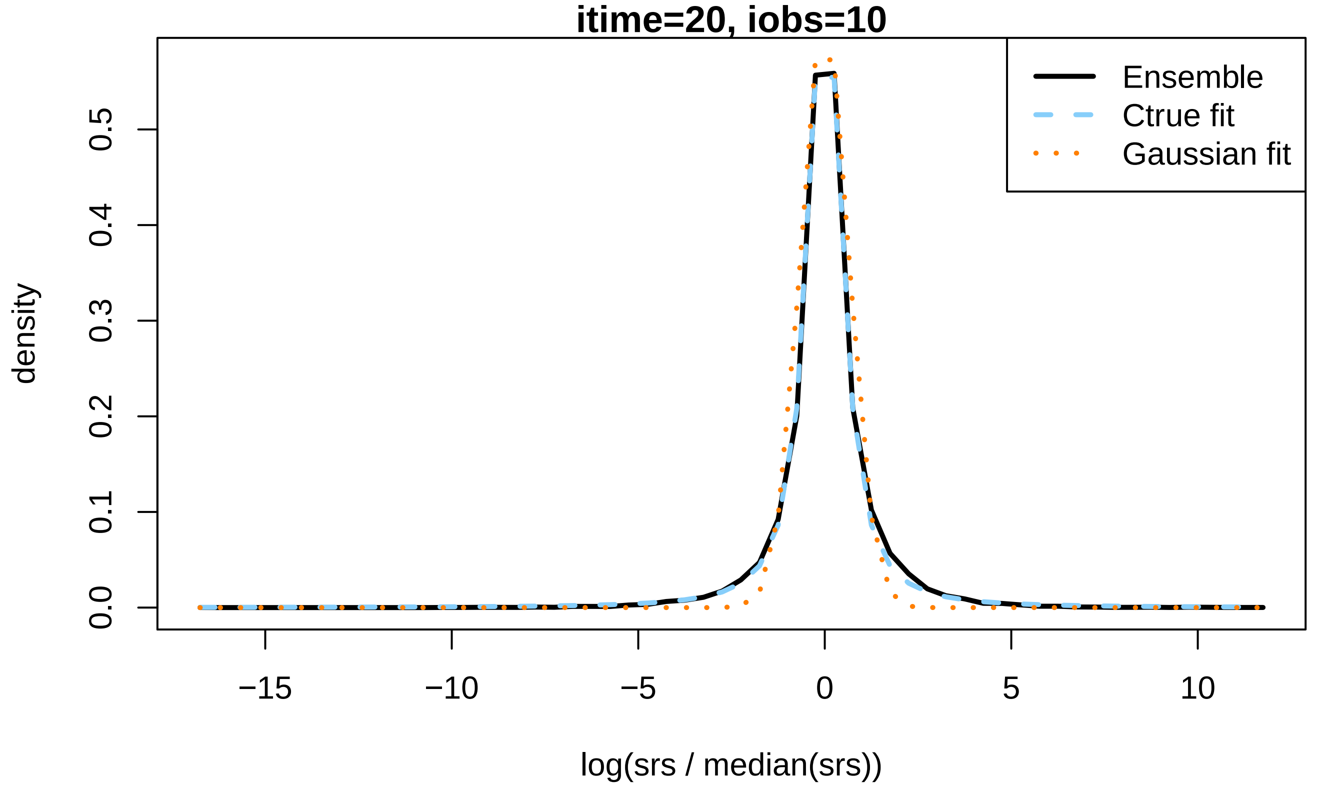

itime=1, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

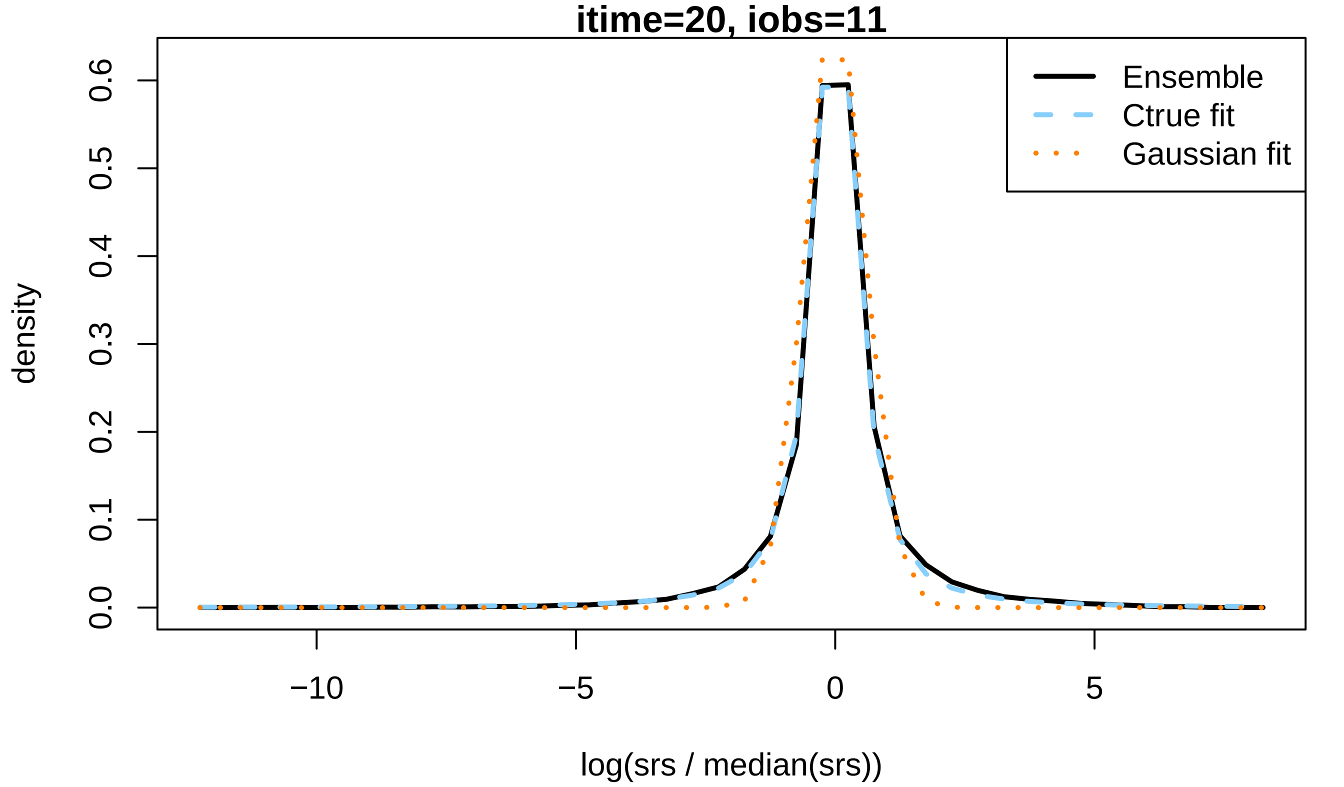


itime=1, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

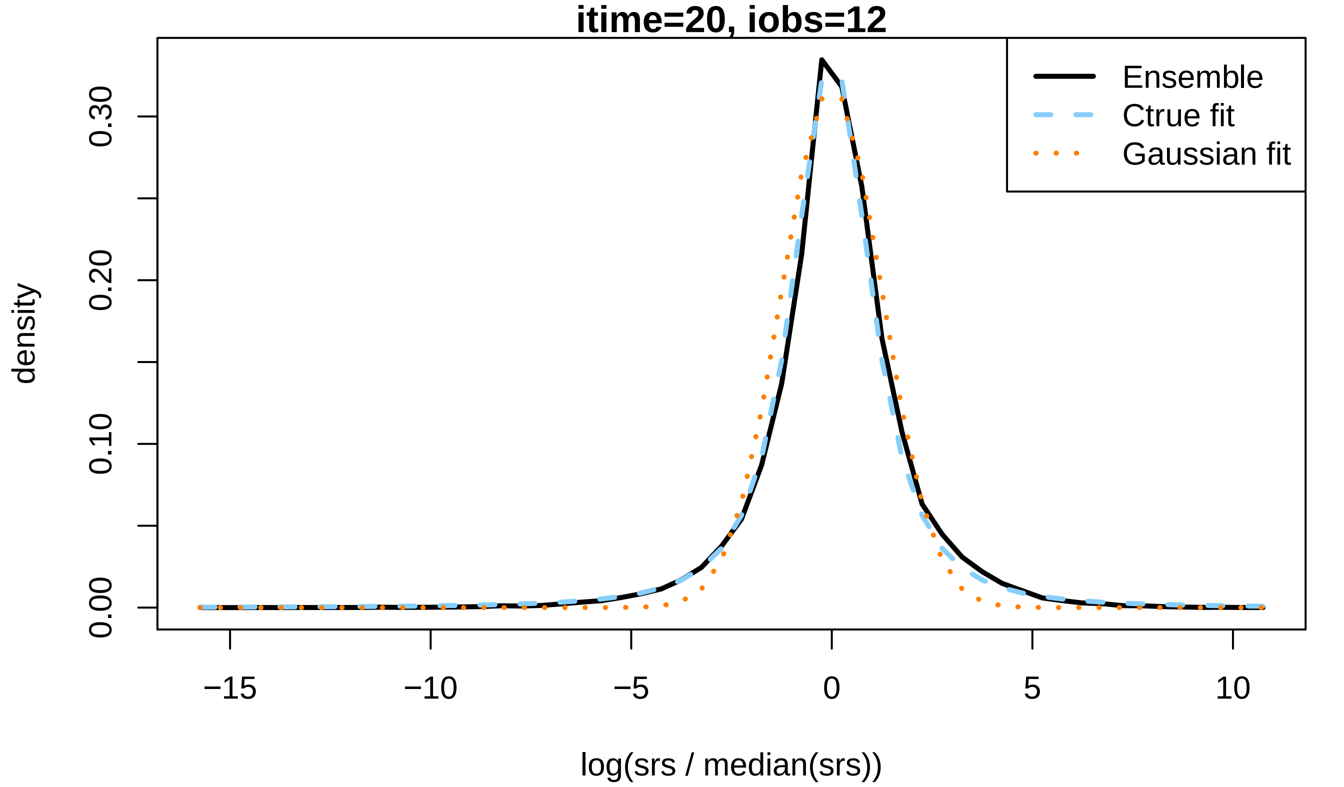

itime=1, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

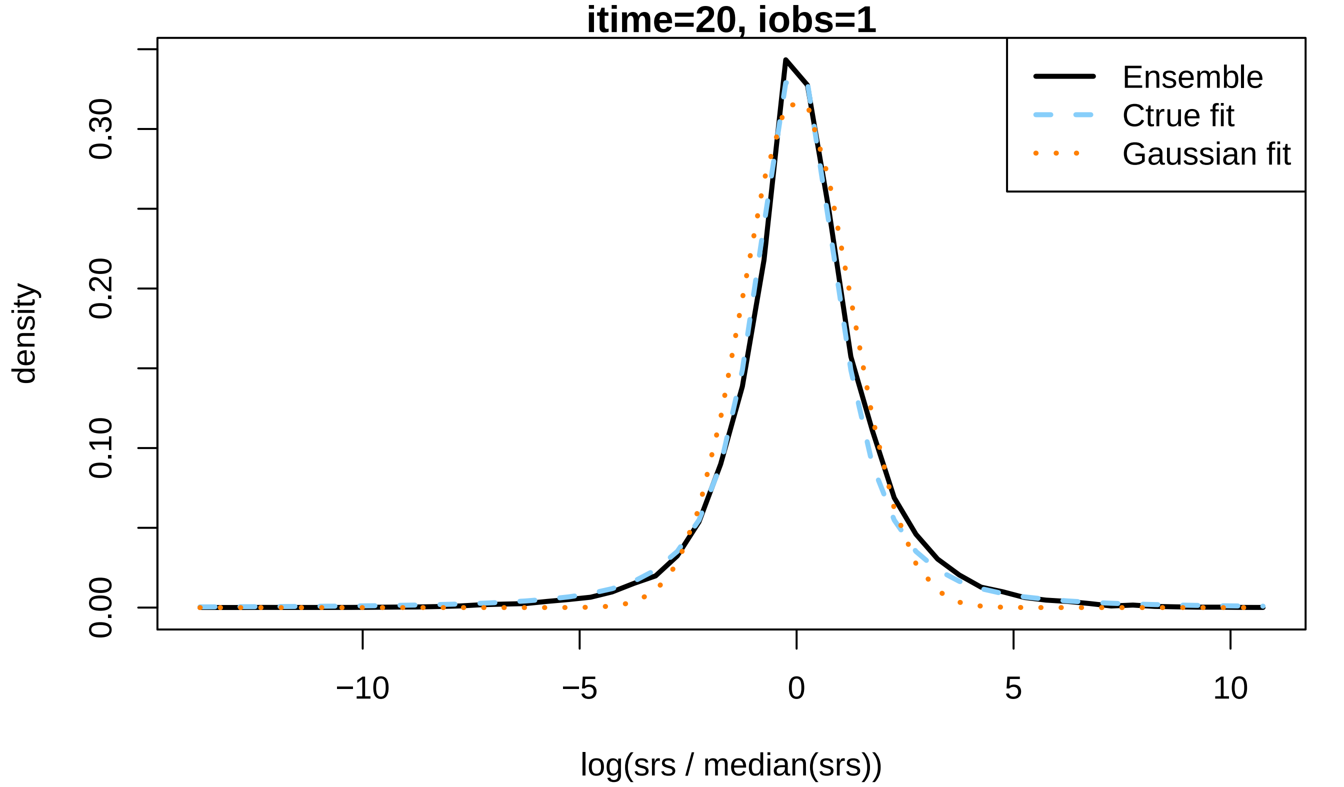

itime=1, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=20, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
−15
−10
−5
0
5
10
log(srs / median(srs))

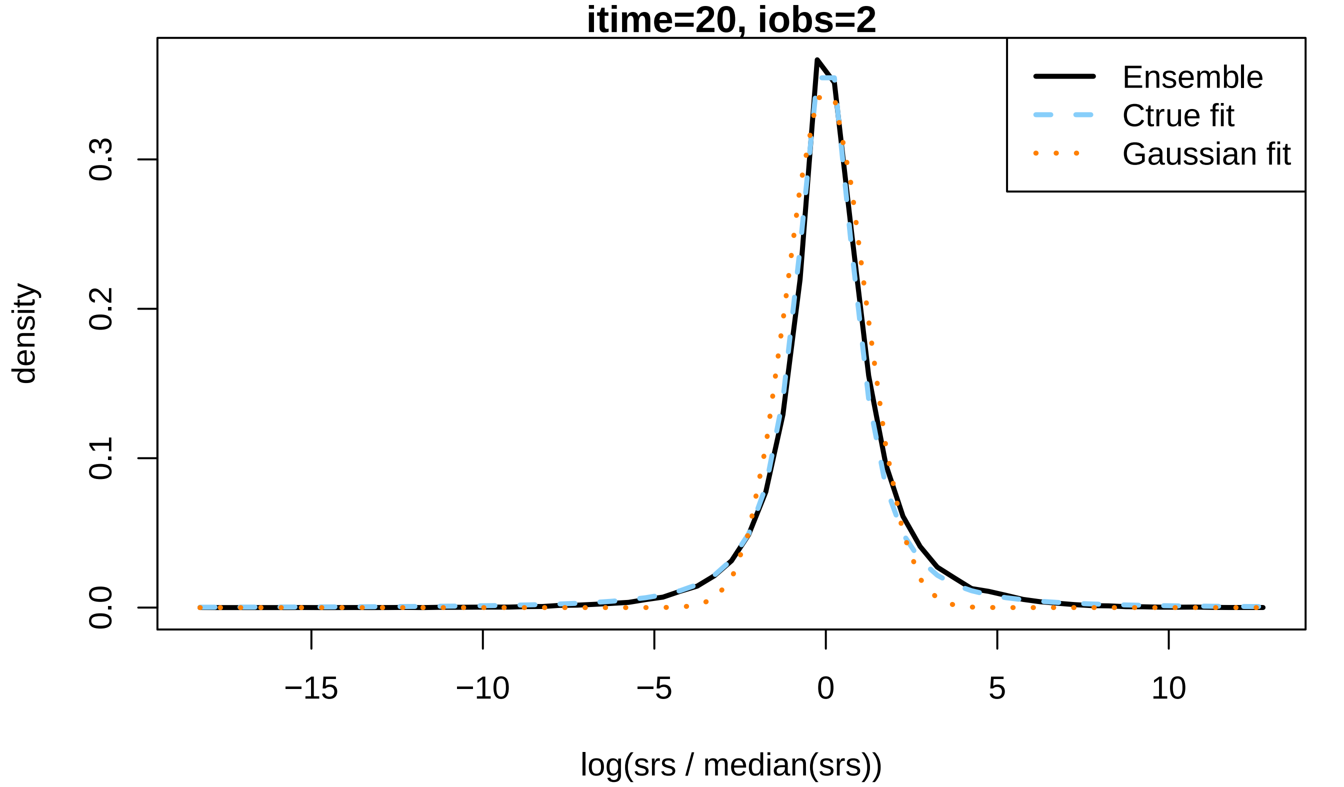


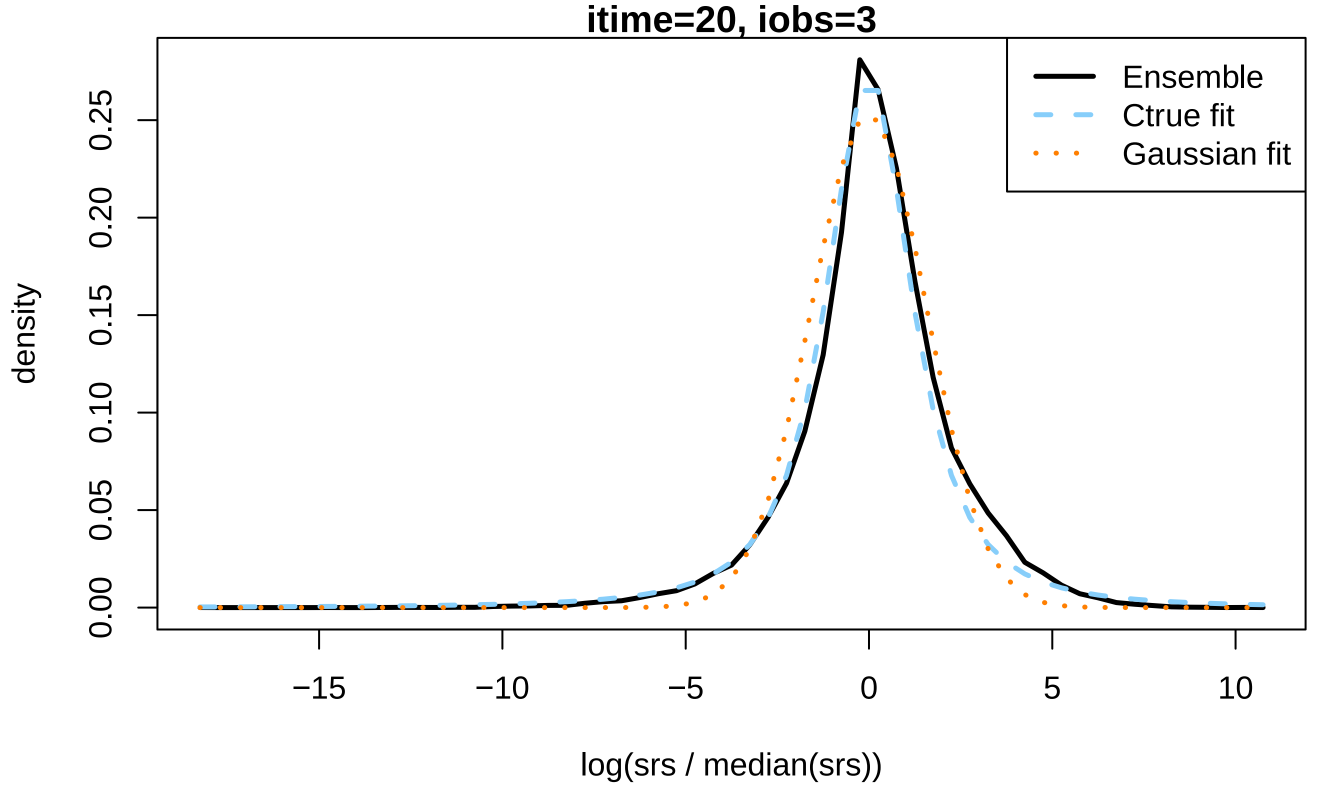
itime=20, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−10
−5
0
5
log(srs / median(srs))


itime=20, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

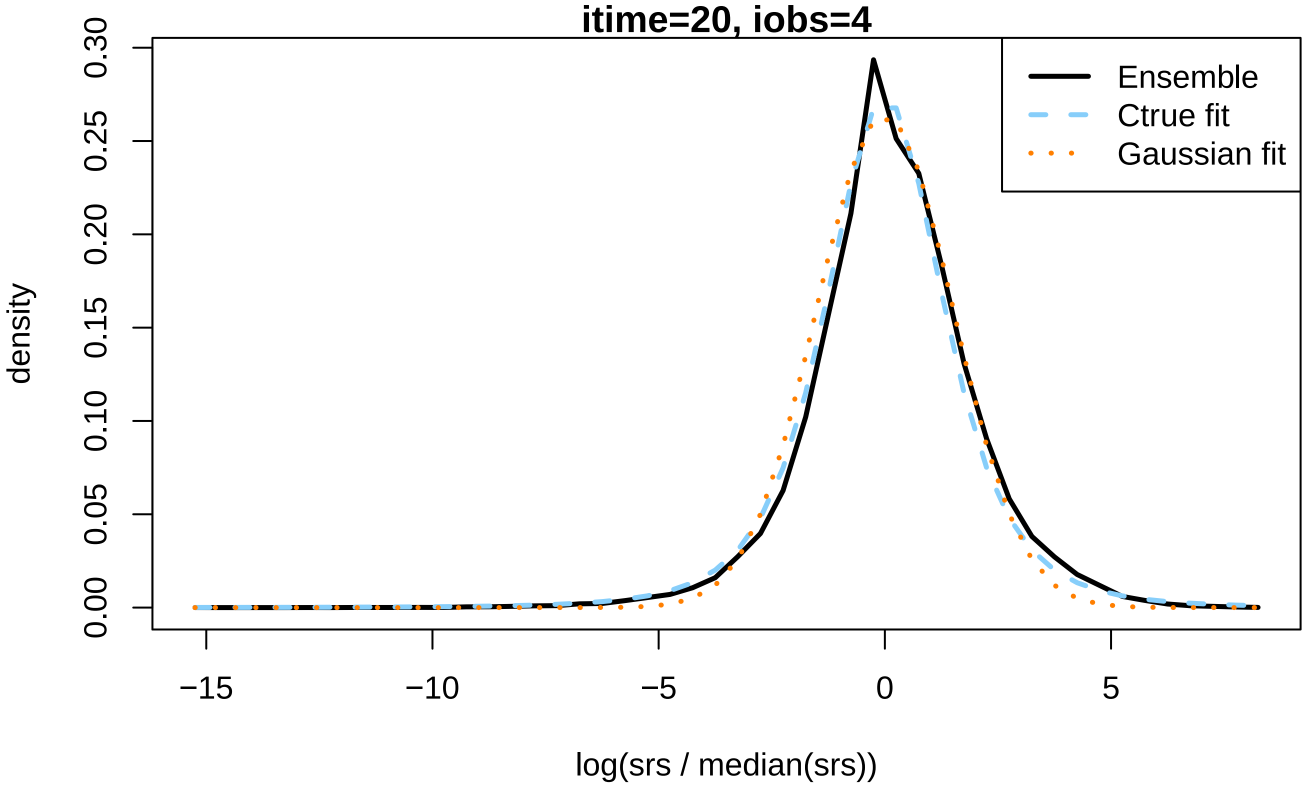


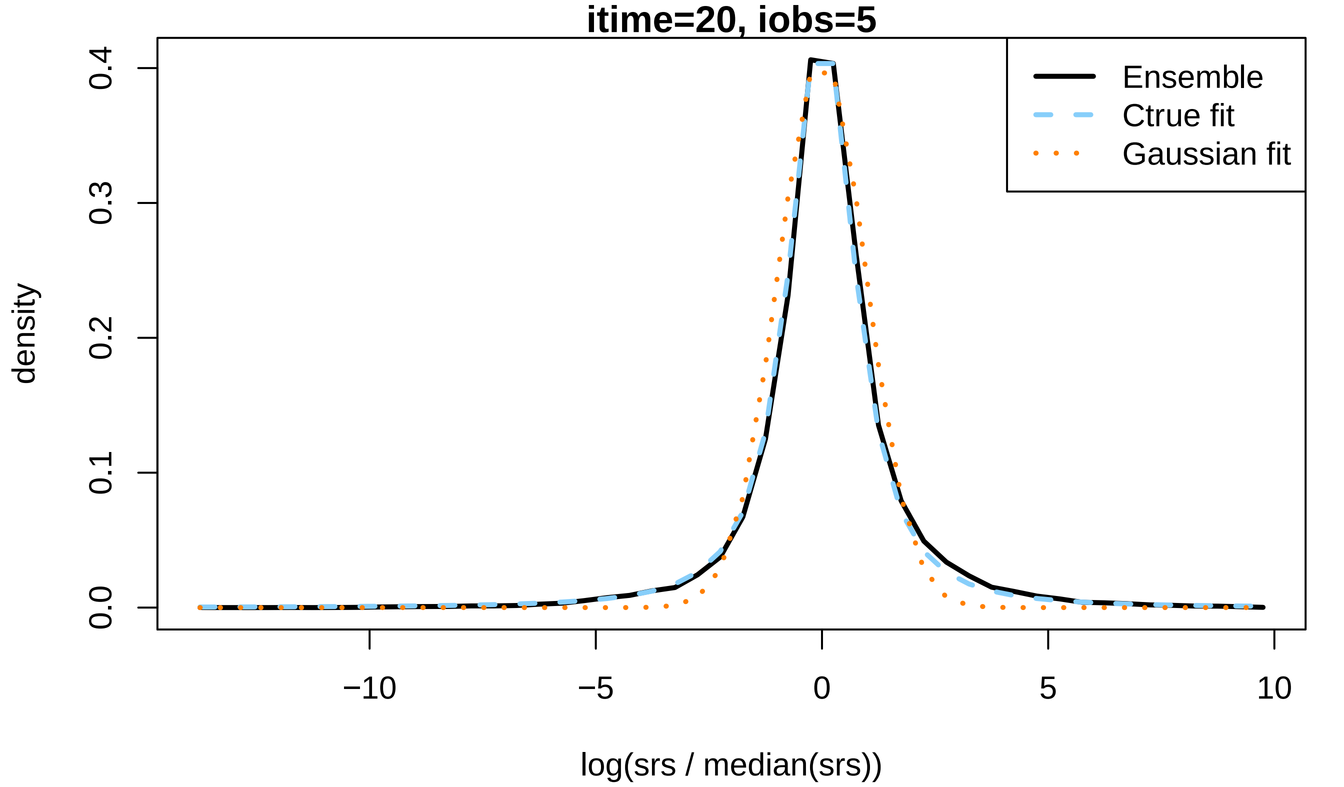
itime=20, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))

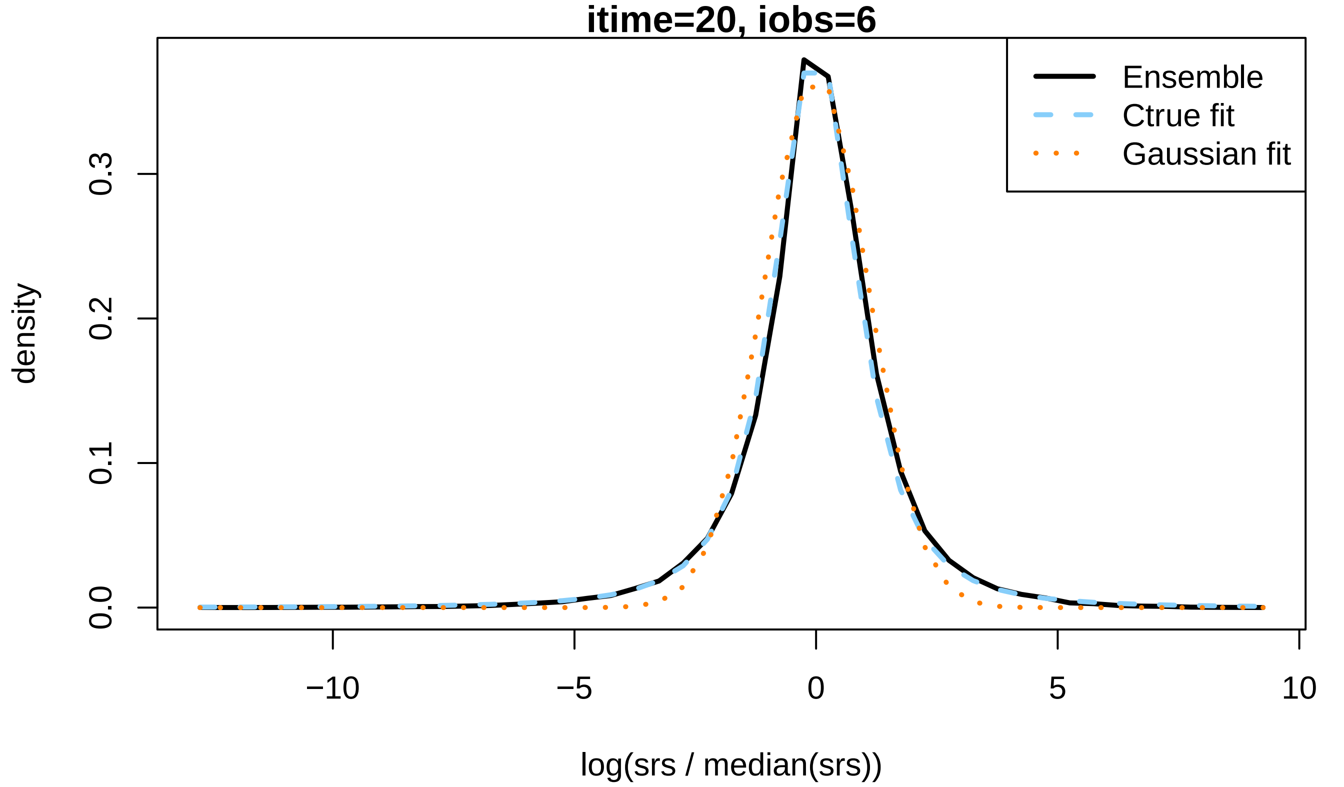

itime=20, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))



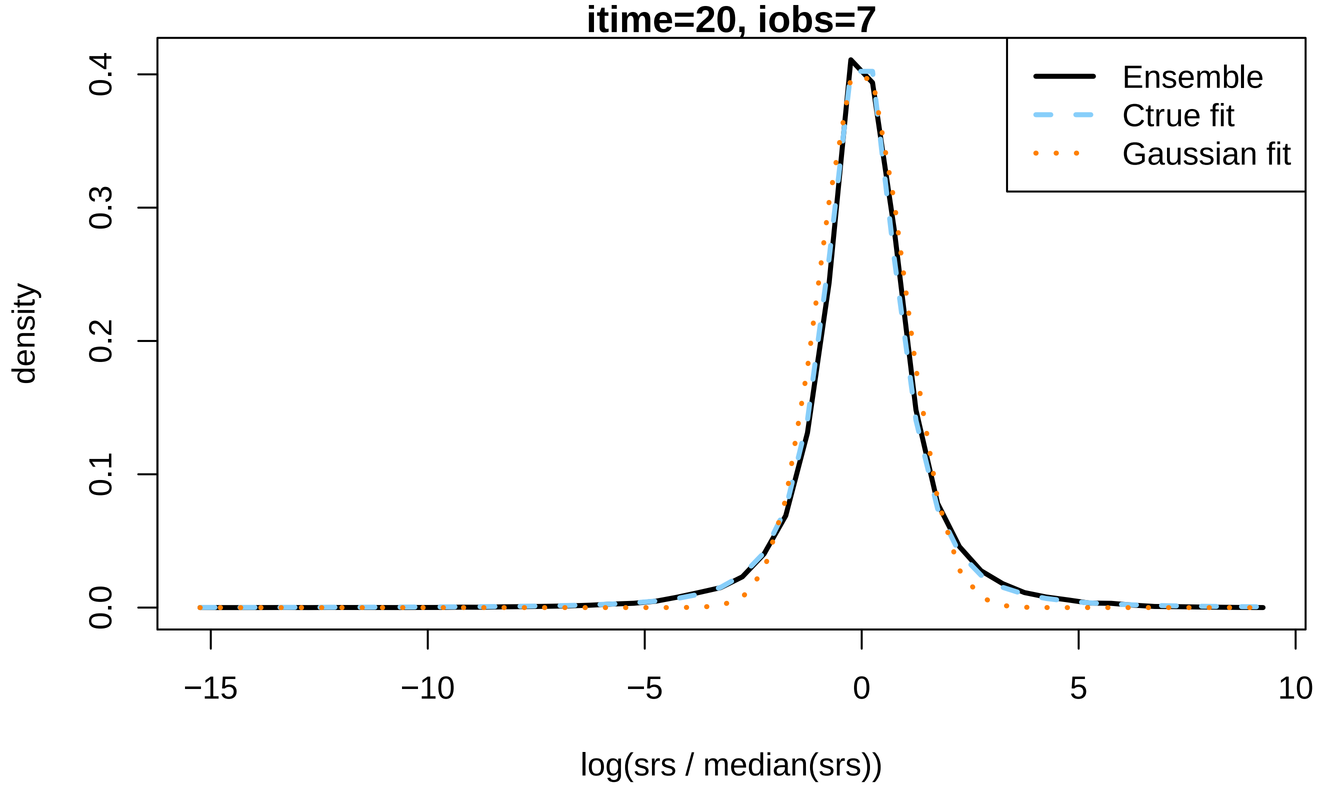
itime=20, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=20, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
log(srs / median(srs))

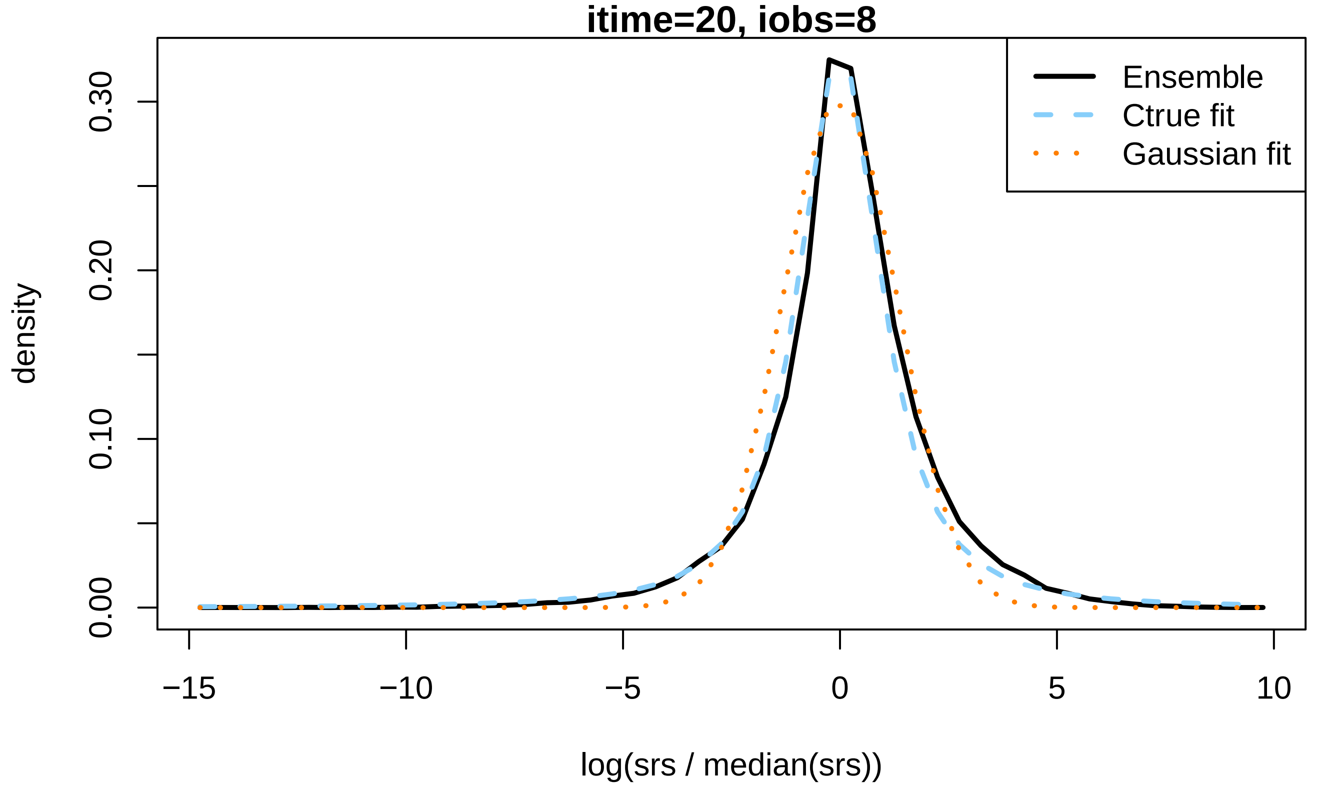

itime=20, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
log(srs / median(srs))



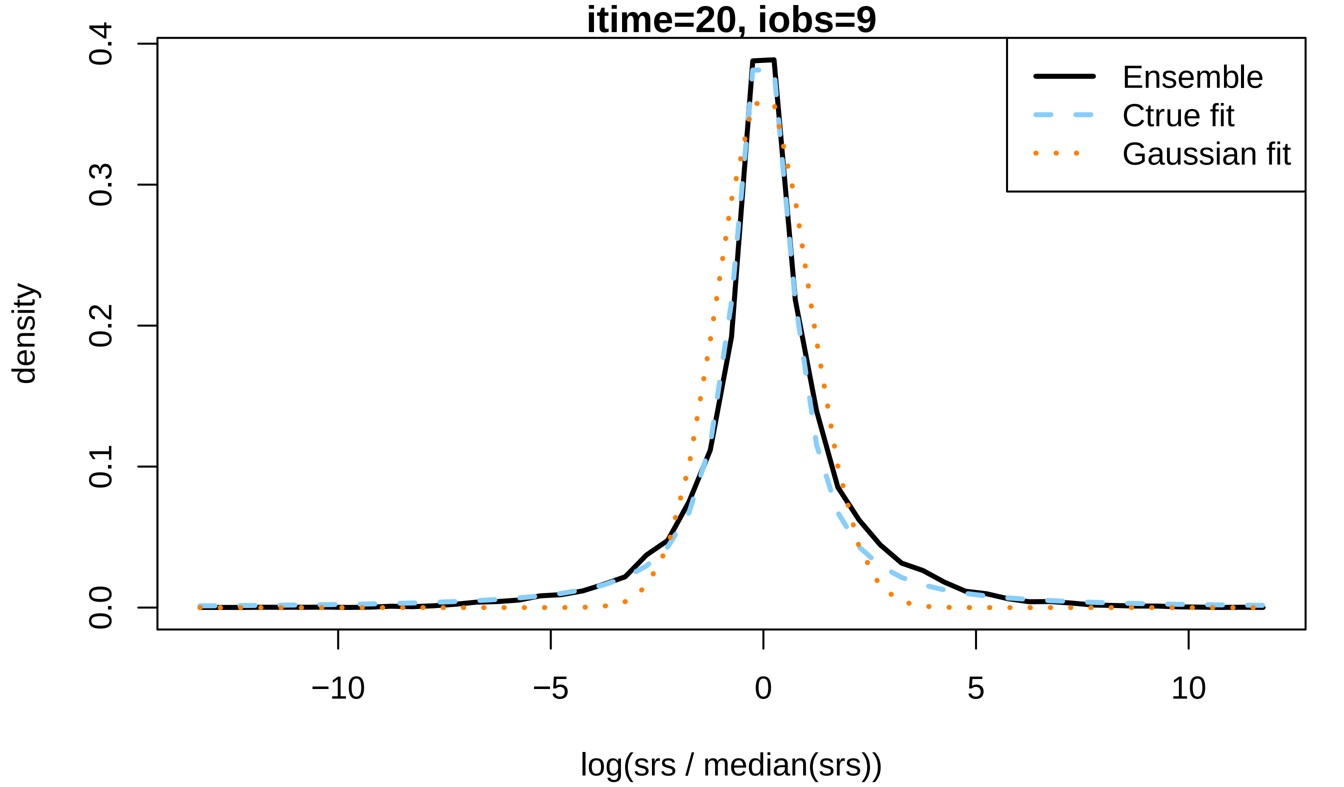
itime=20, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−10
−5
0
5
10
log(srs / median(srs))



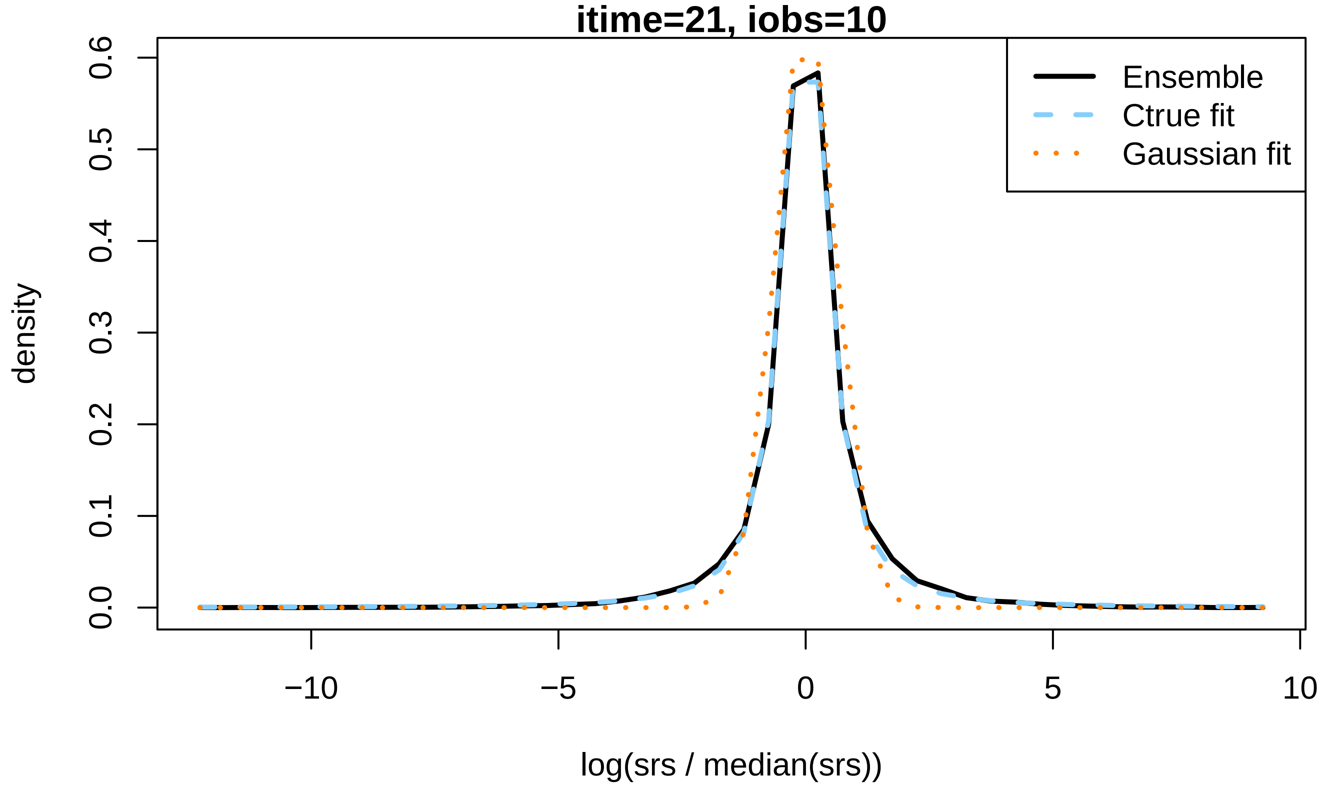
itime=20, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))

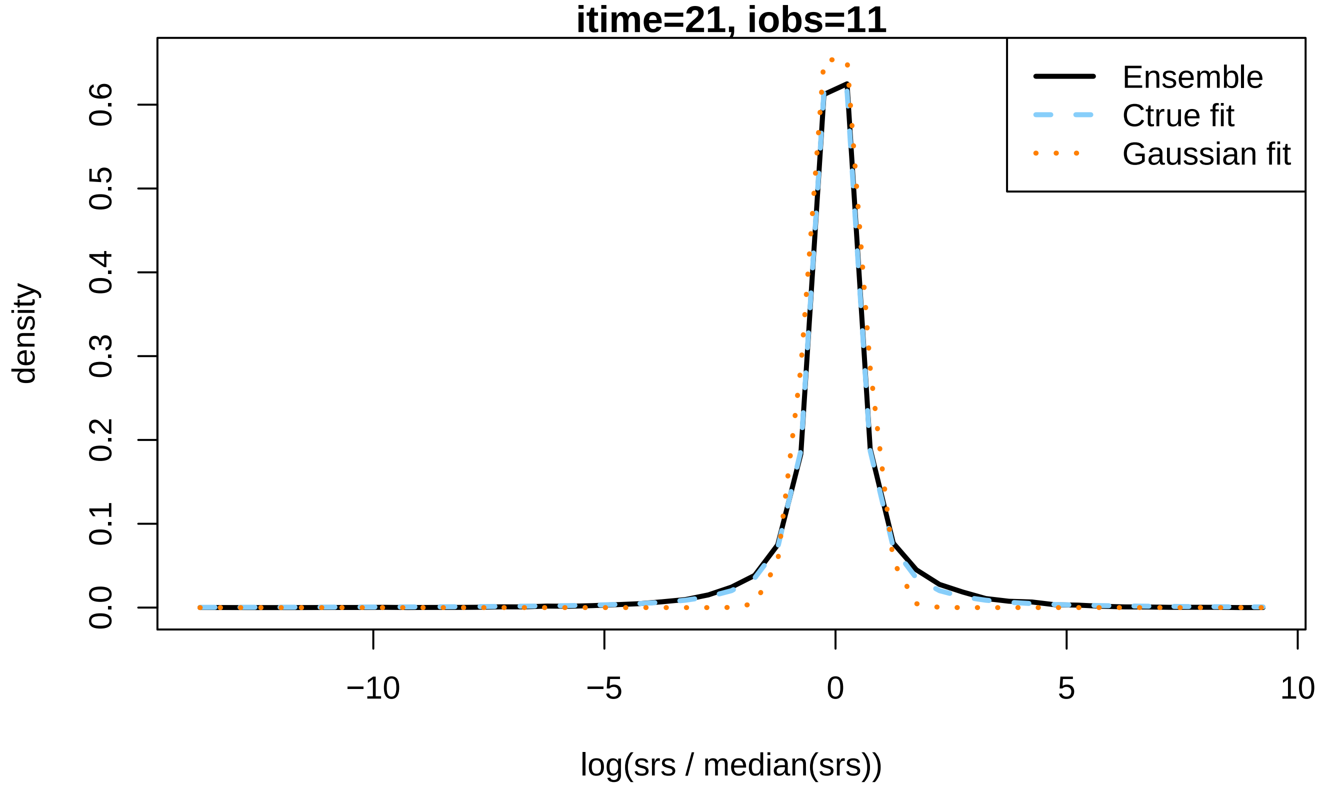


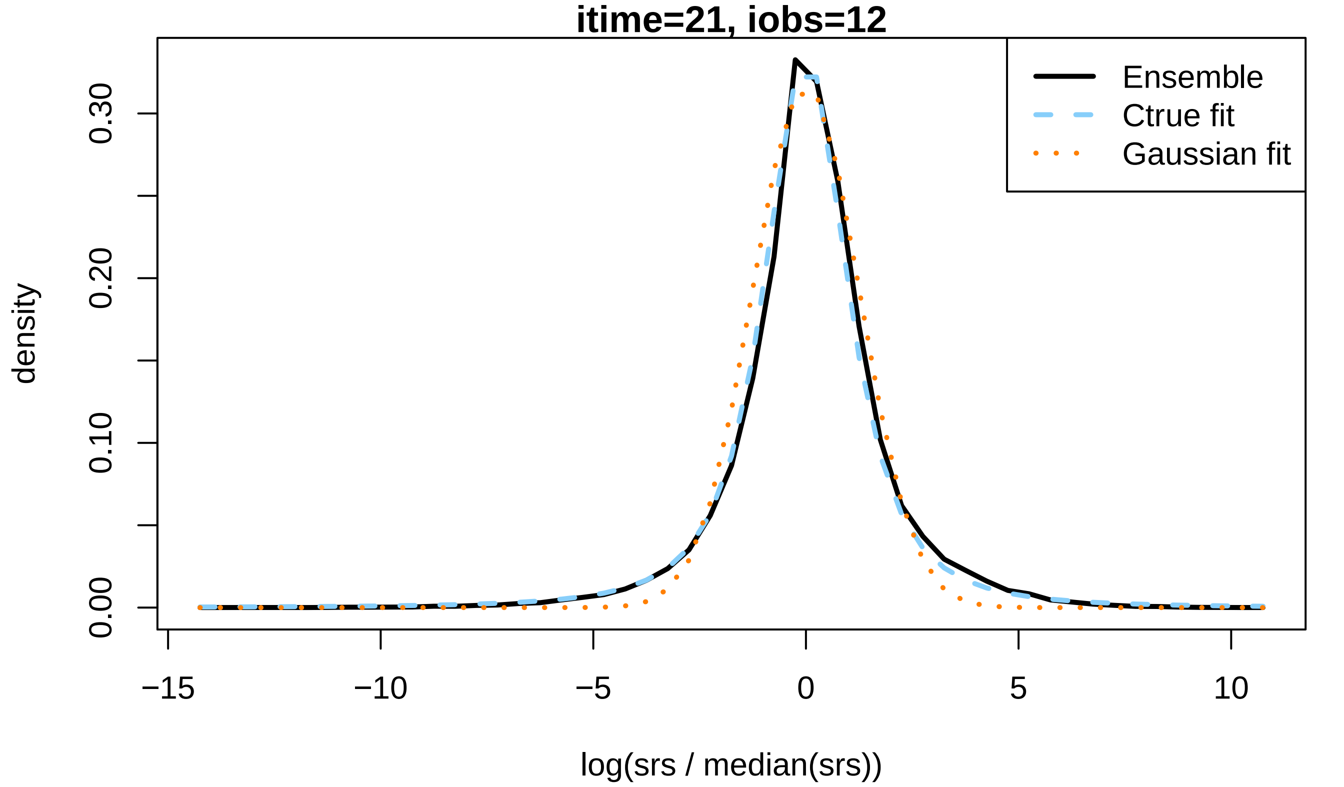
itime=20, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

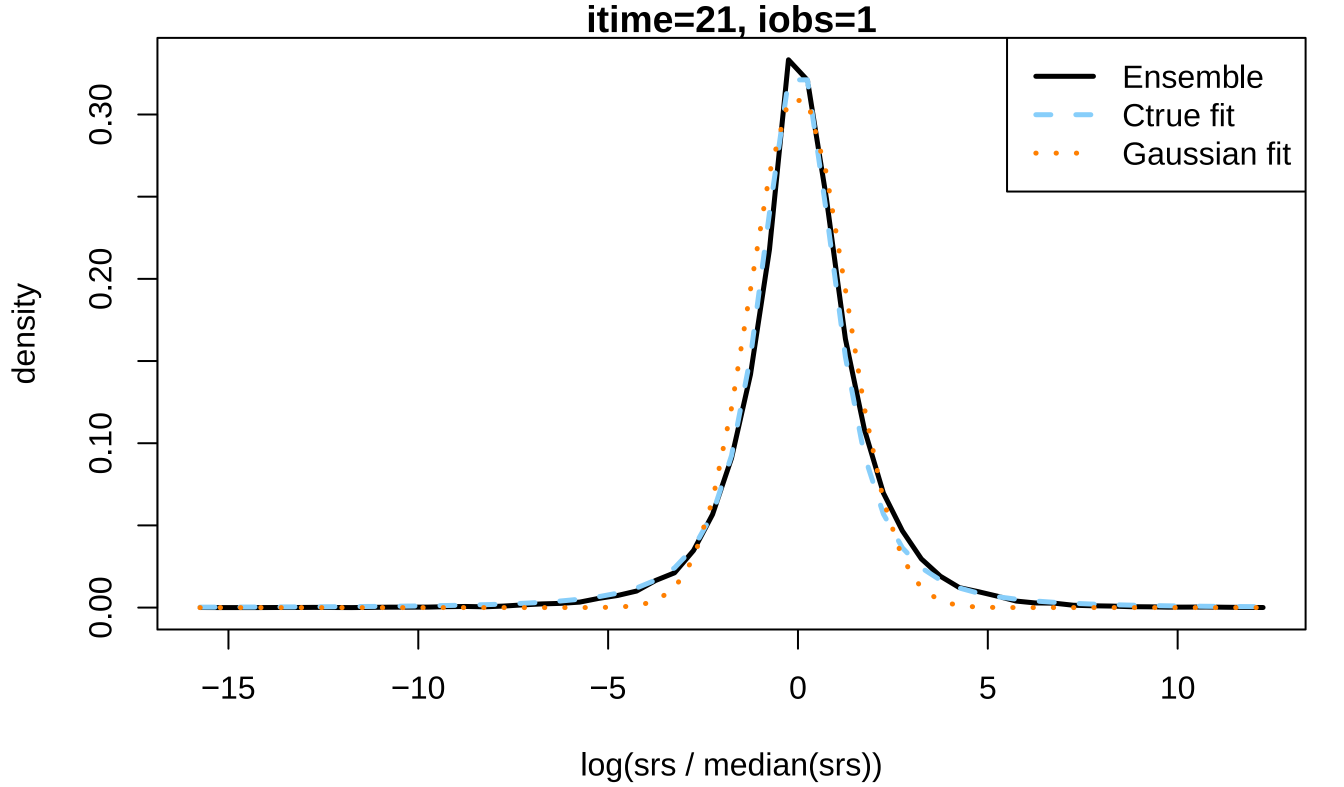


itime=20, iobs=9
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

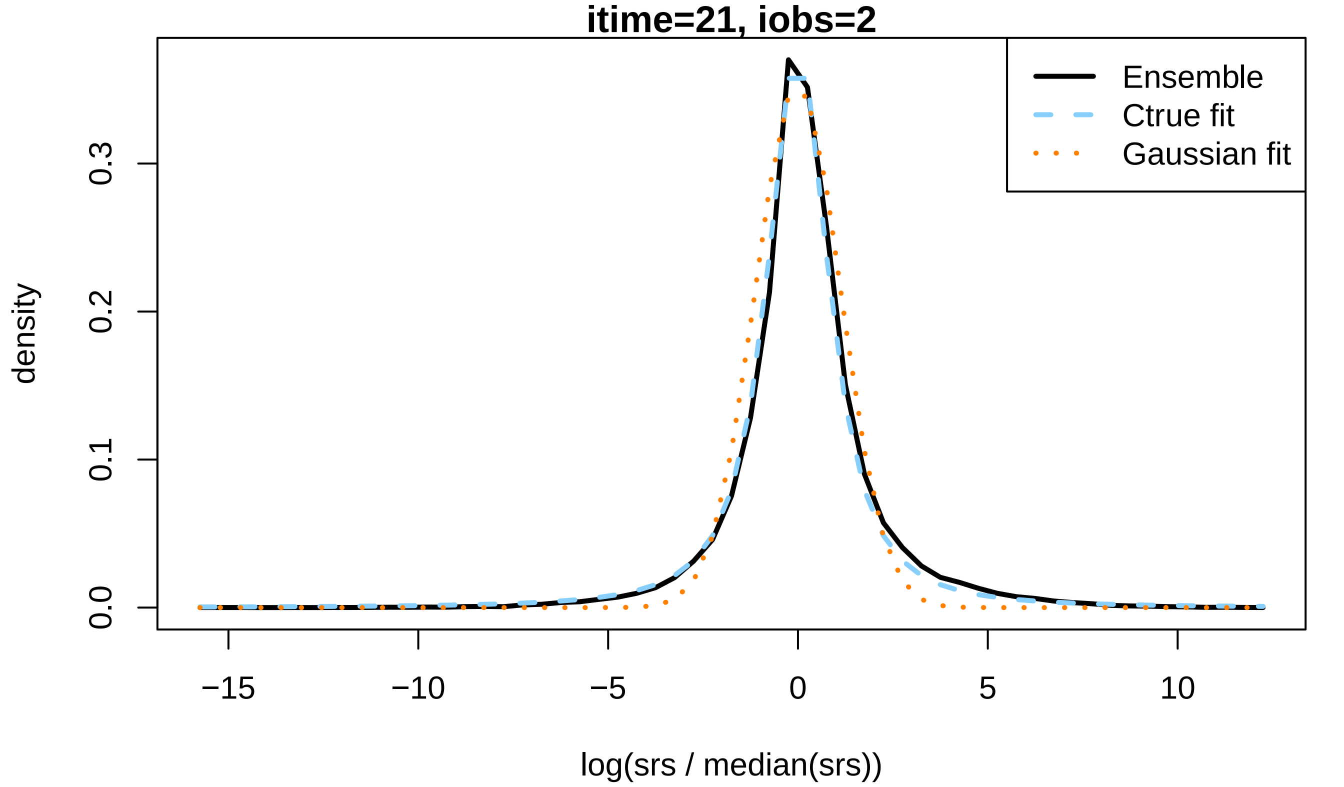

itime=21, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−10
−5
0
5
10
log(srs / median(srs))

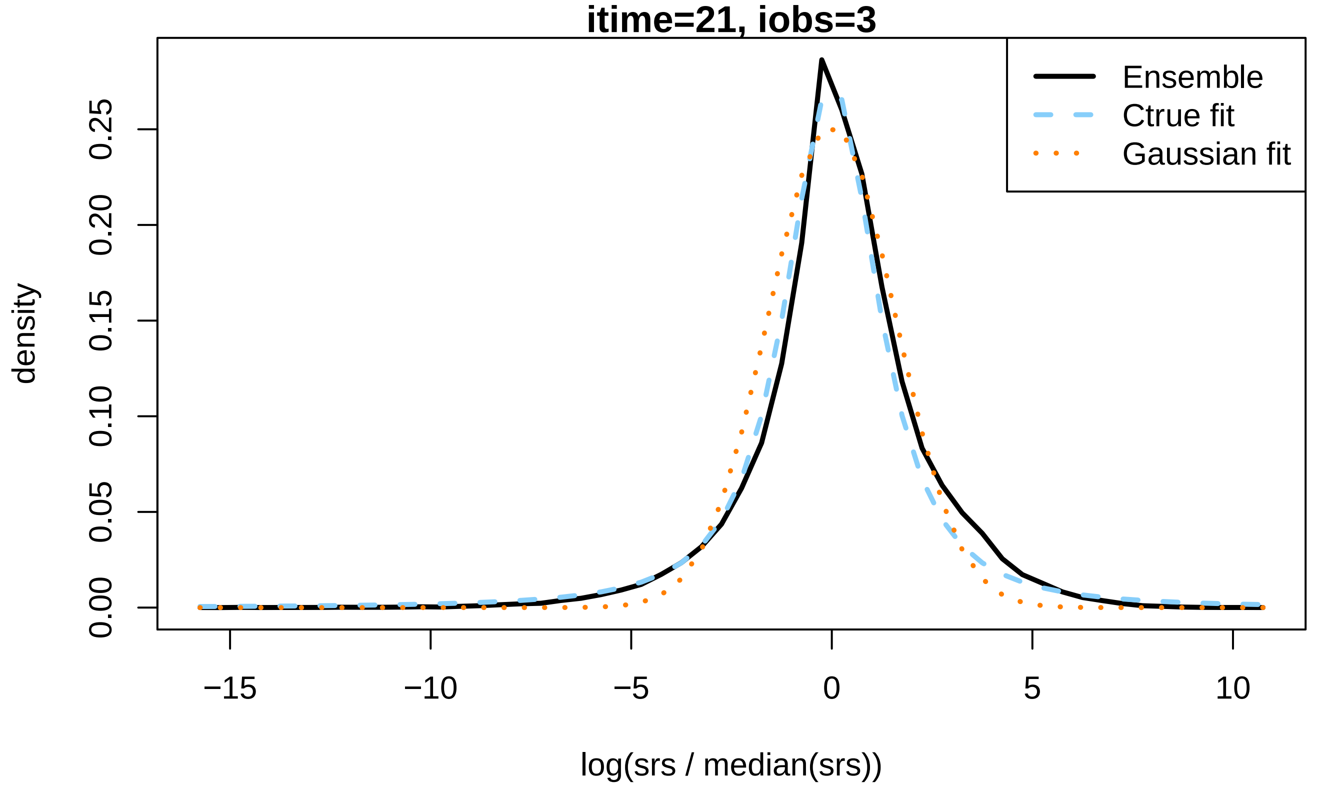

itime=21, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−10
−5
0
5
10
log(srs / median(srs))


itime=21, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

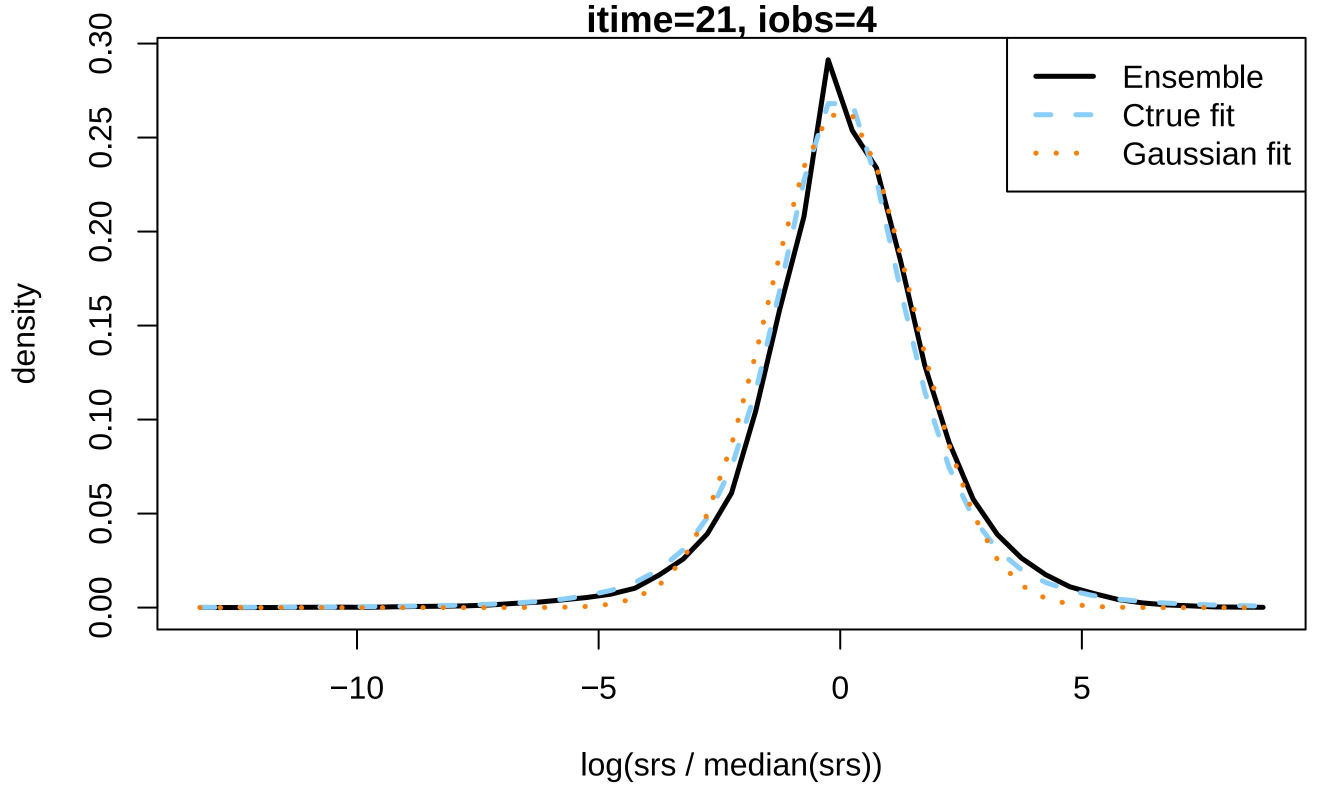


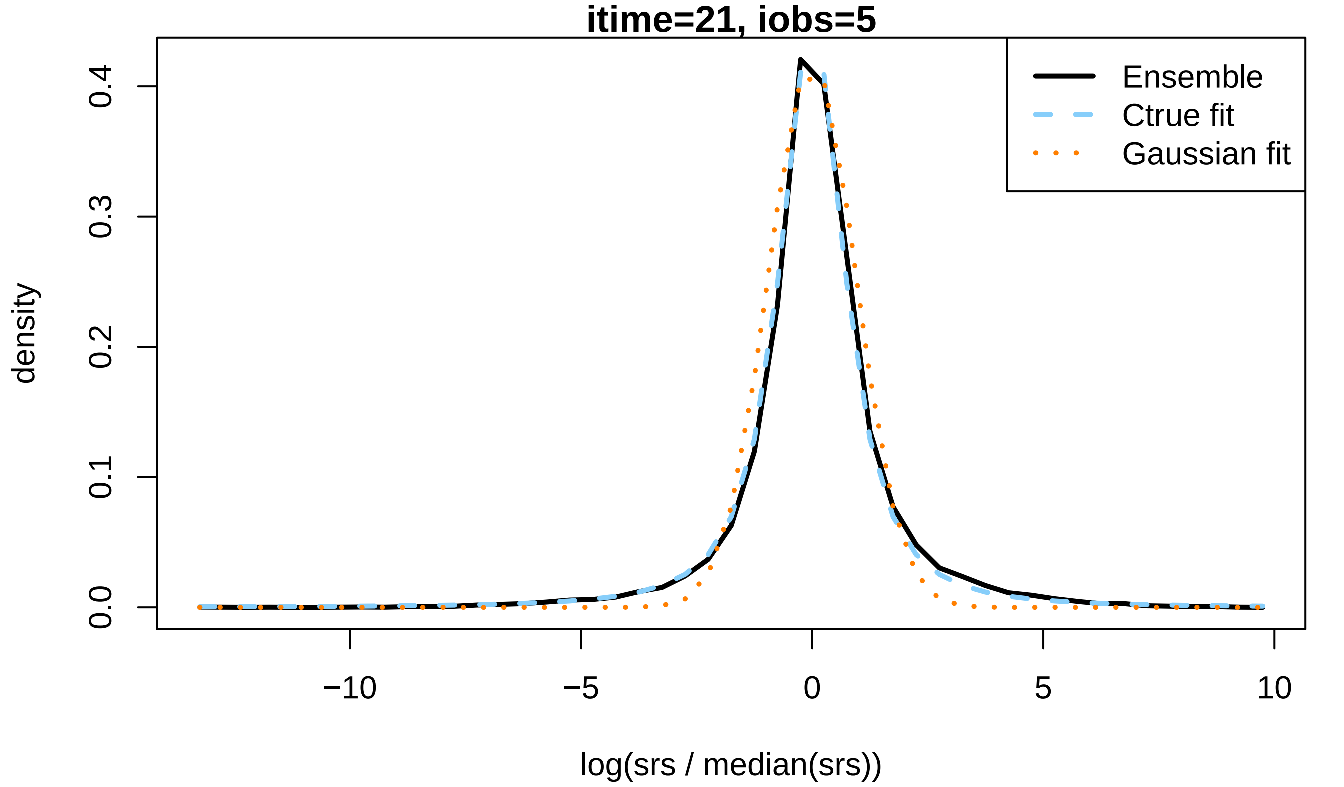
itime=21, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=21, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

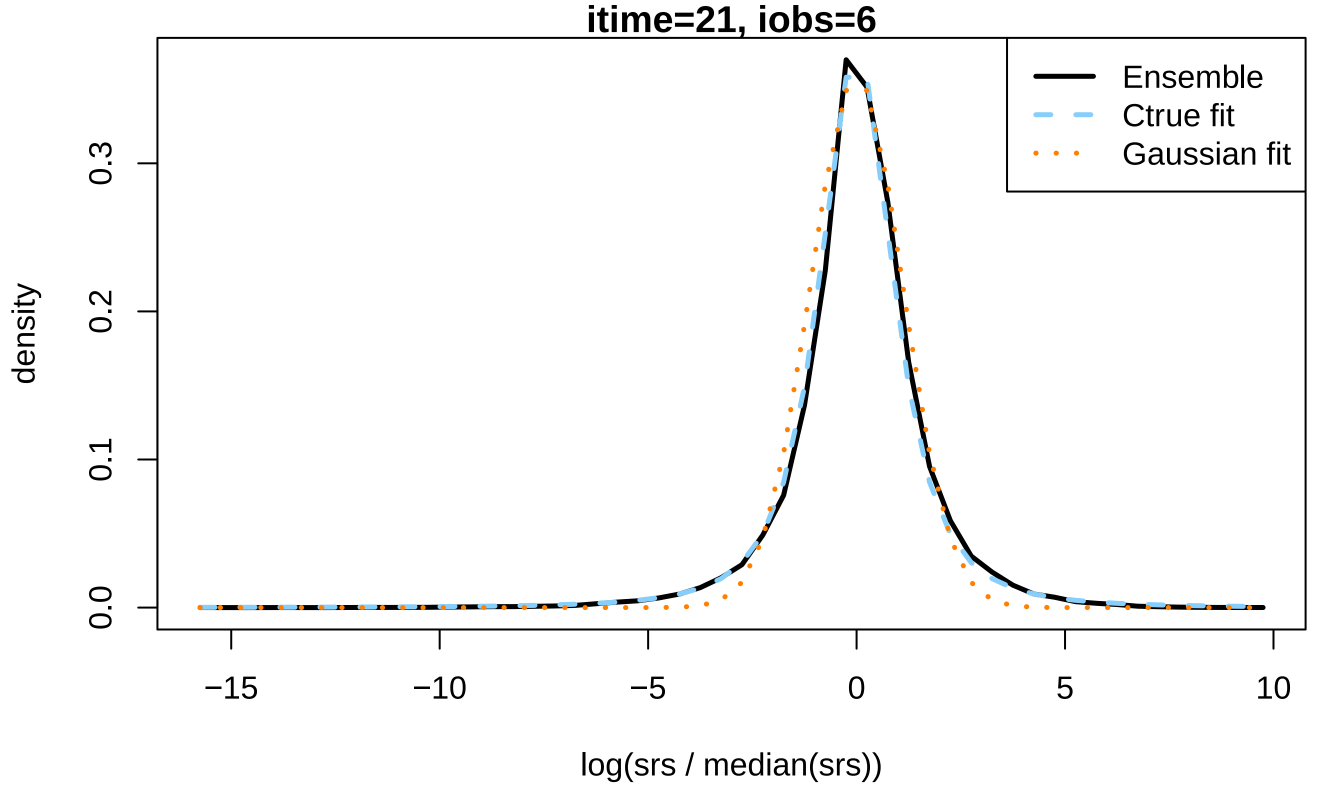


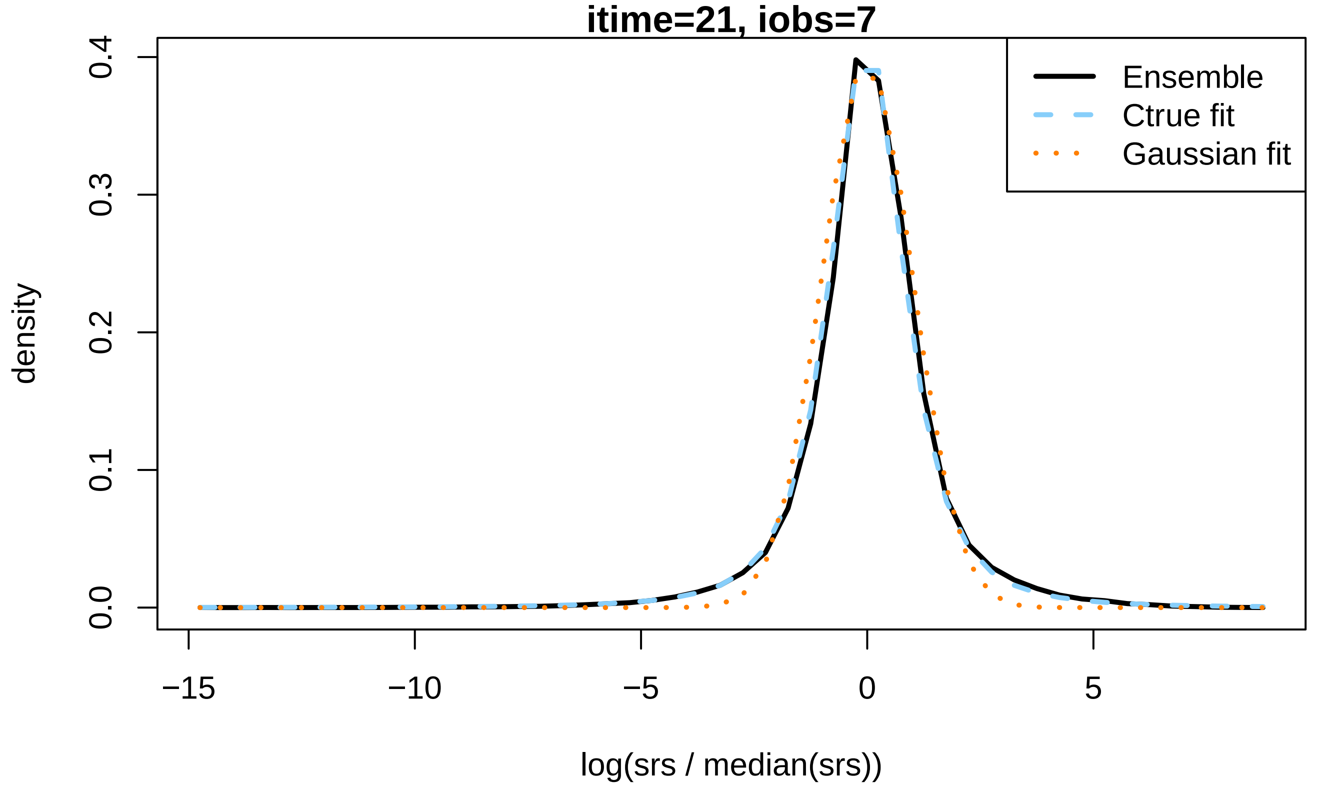
itime=21, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

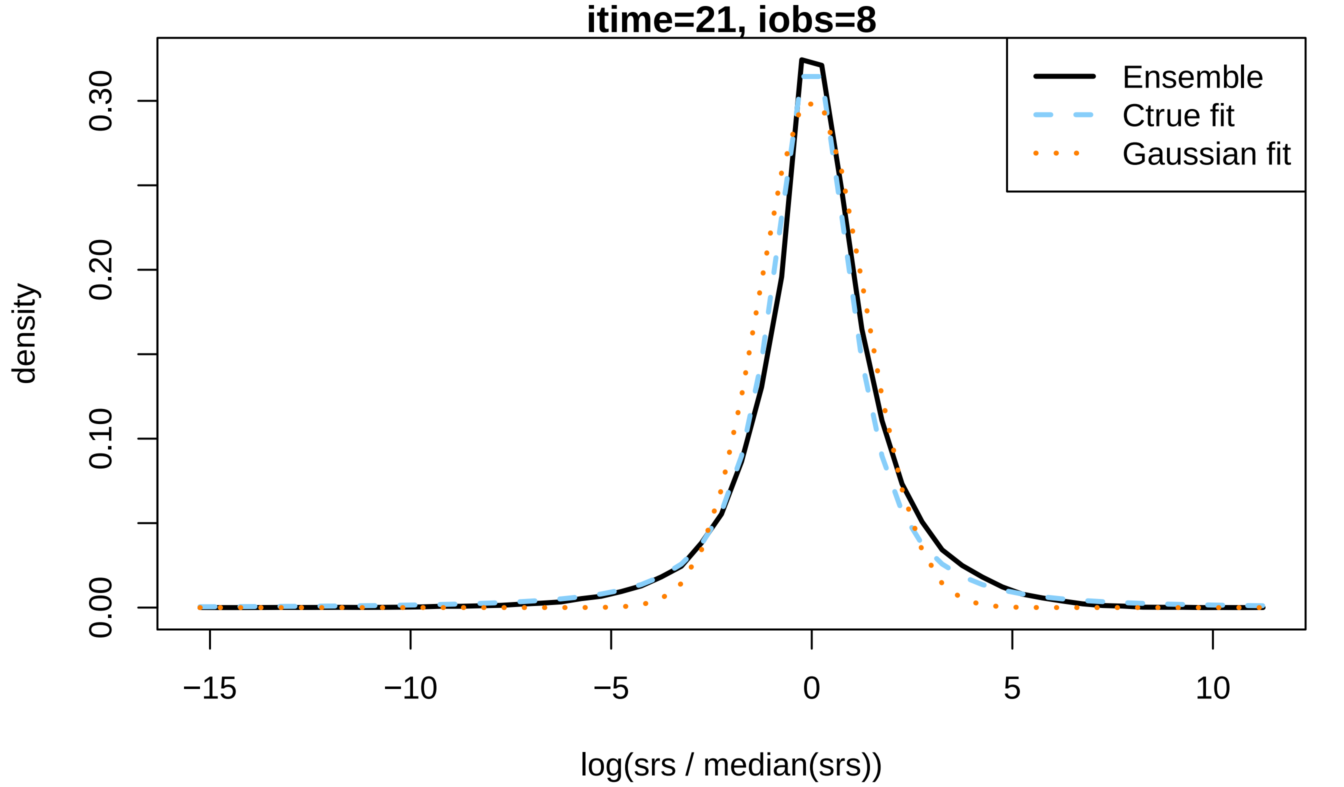

itime=21, iobs=4
density
log(srs / median(srs))
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−10
−5
0
5
Ensemble
Ctrue fit
Gaussian fit

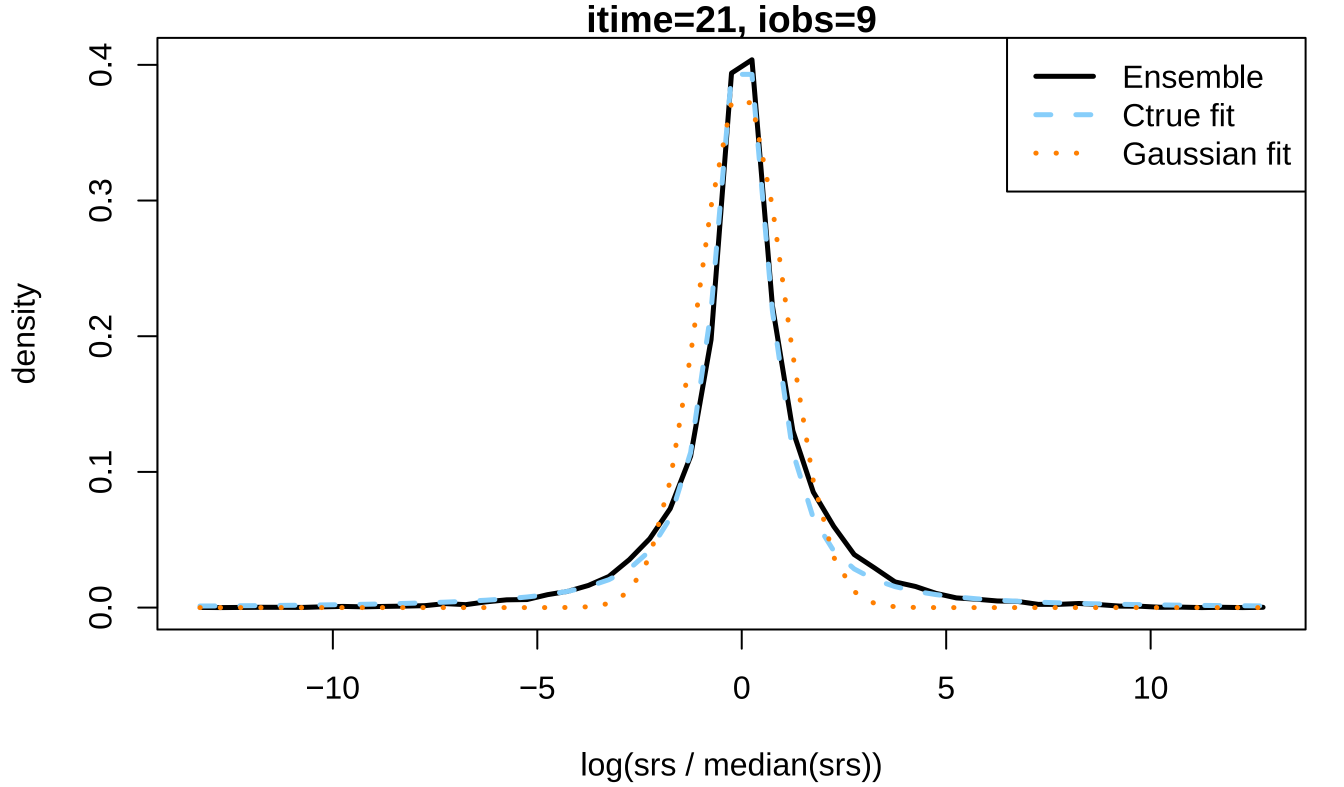


itime=21, iobs=5
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

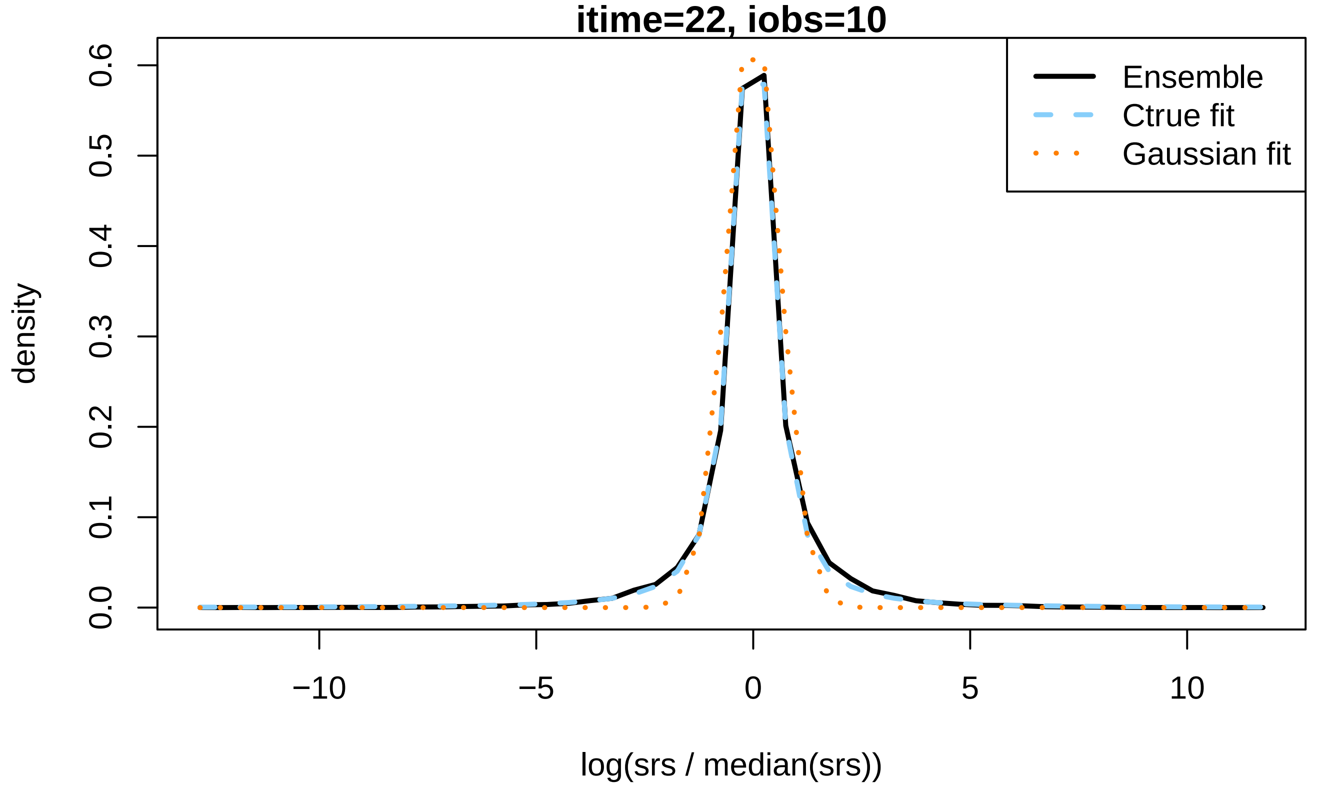

itime=21, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=21, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
log(srs / median(srs))

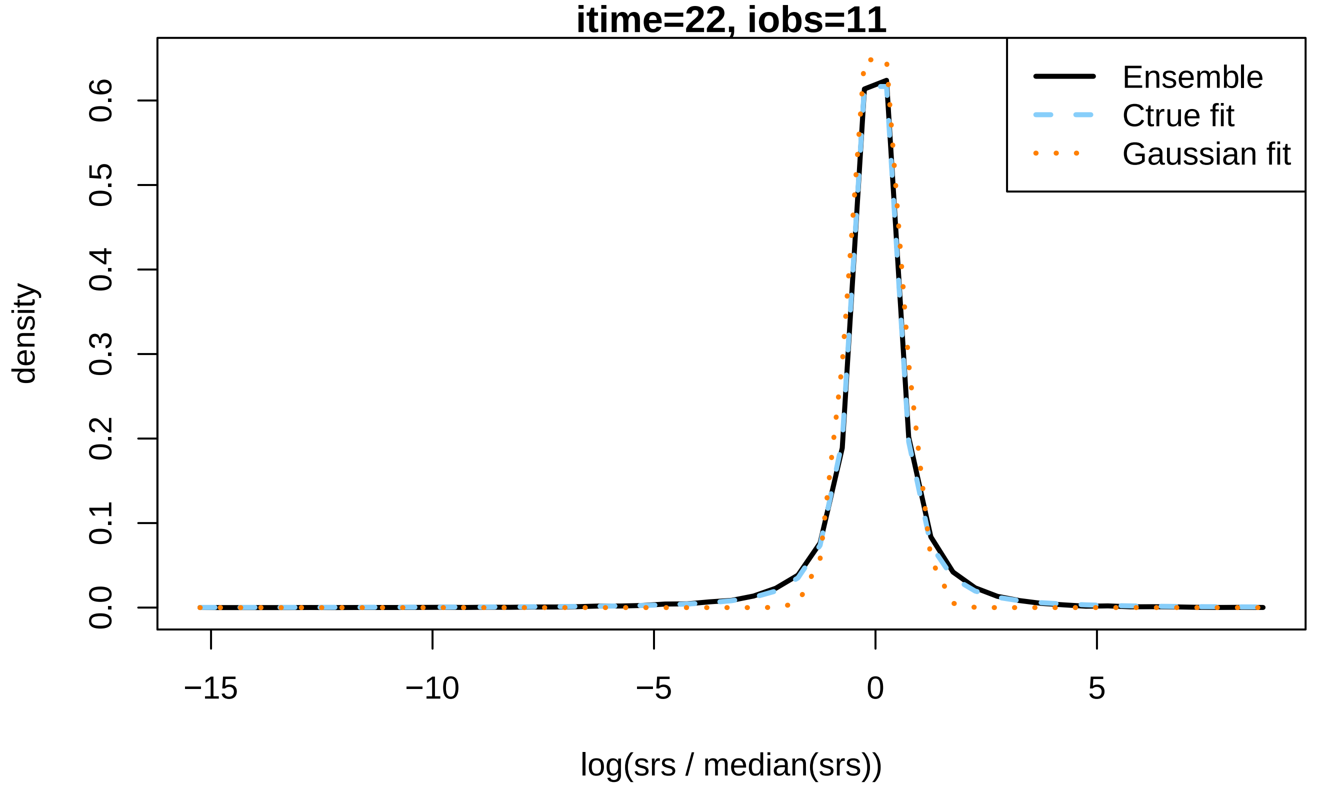

itime=21, iobs=8
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit



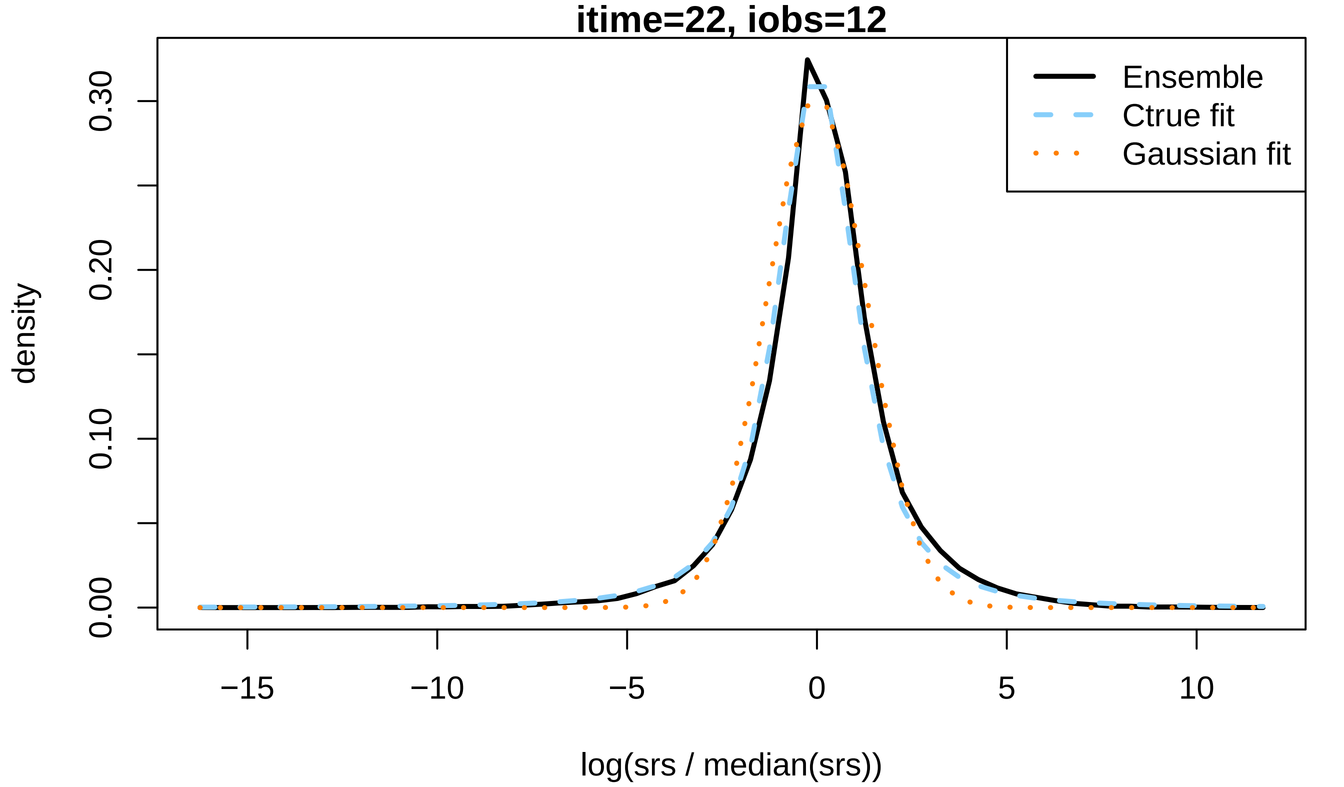
itime=21, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
log(srs / median(srs))



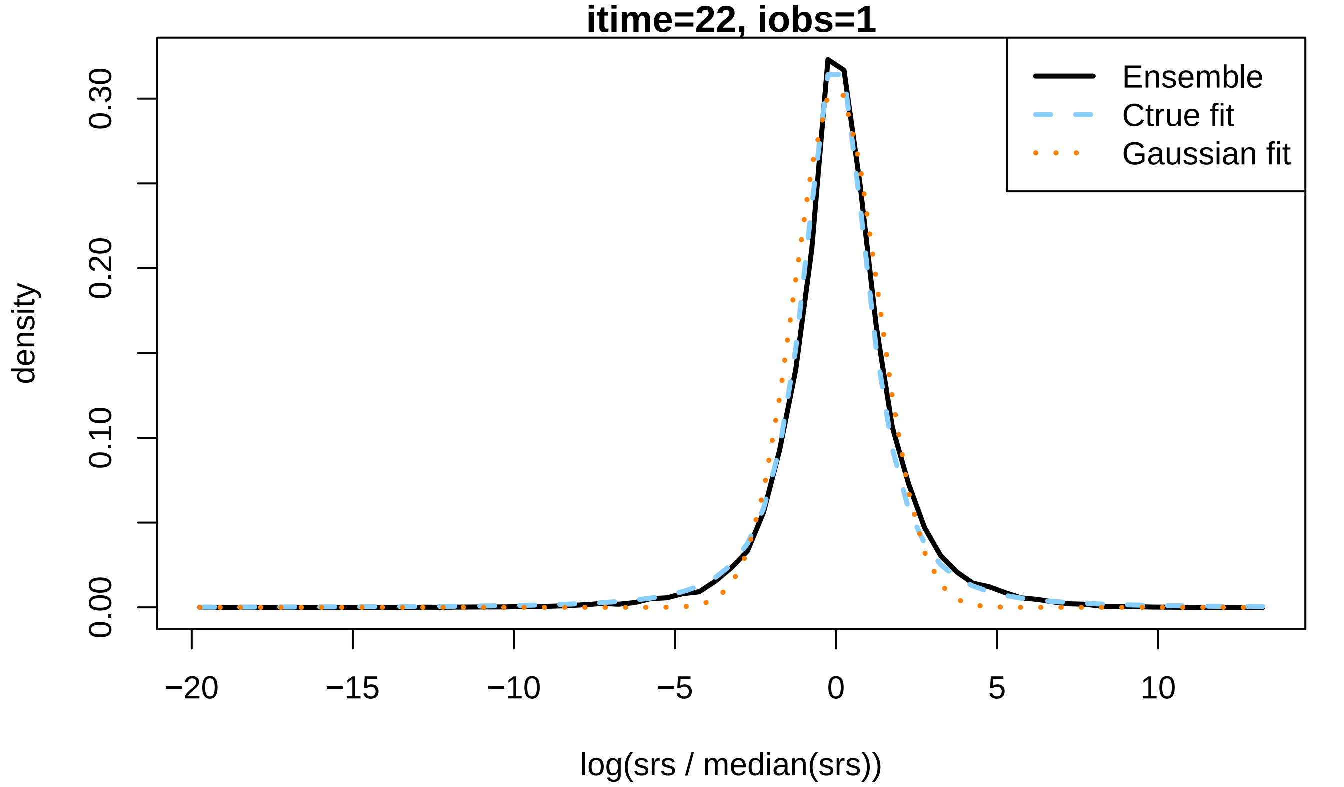
itime=22, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−10
−5
0
5
10
log(srs / median(srs))



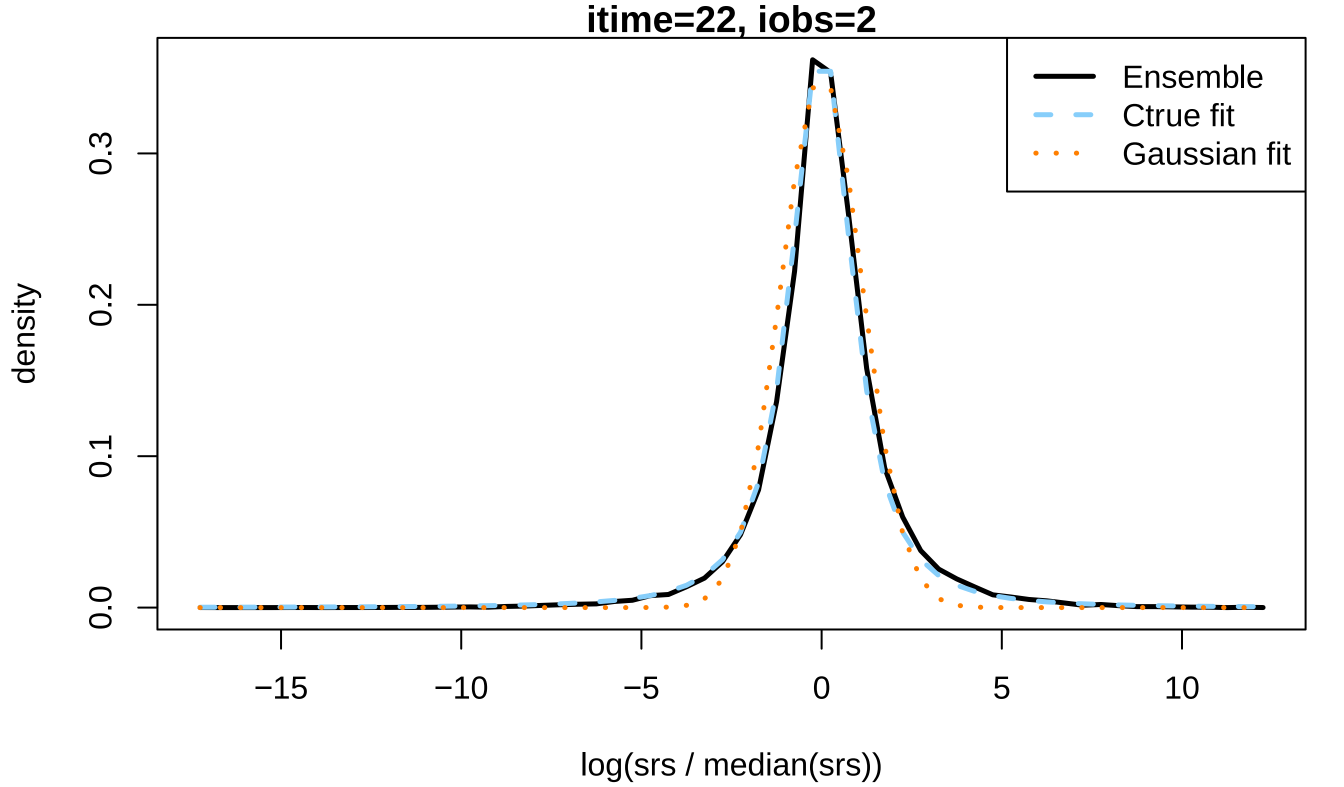
itime=22, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−15
−10
−5
0
5
log(srs / median(srs))

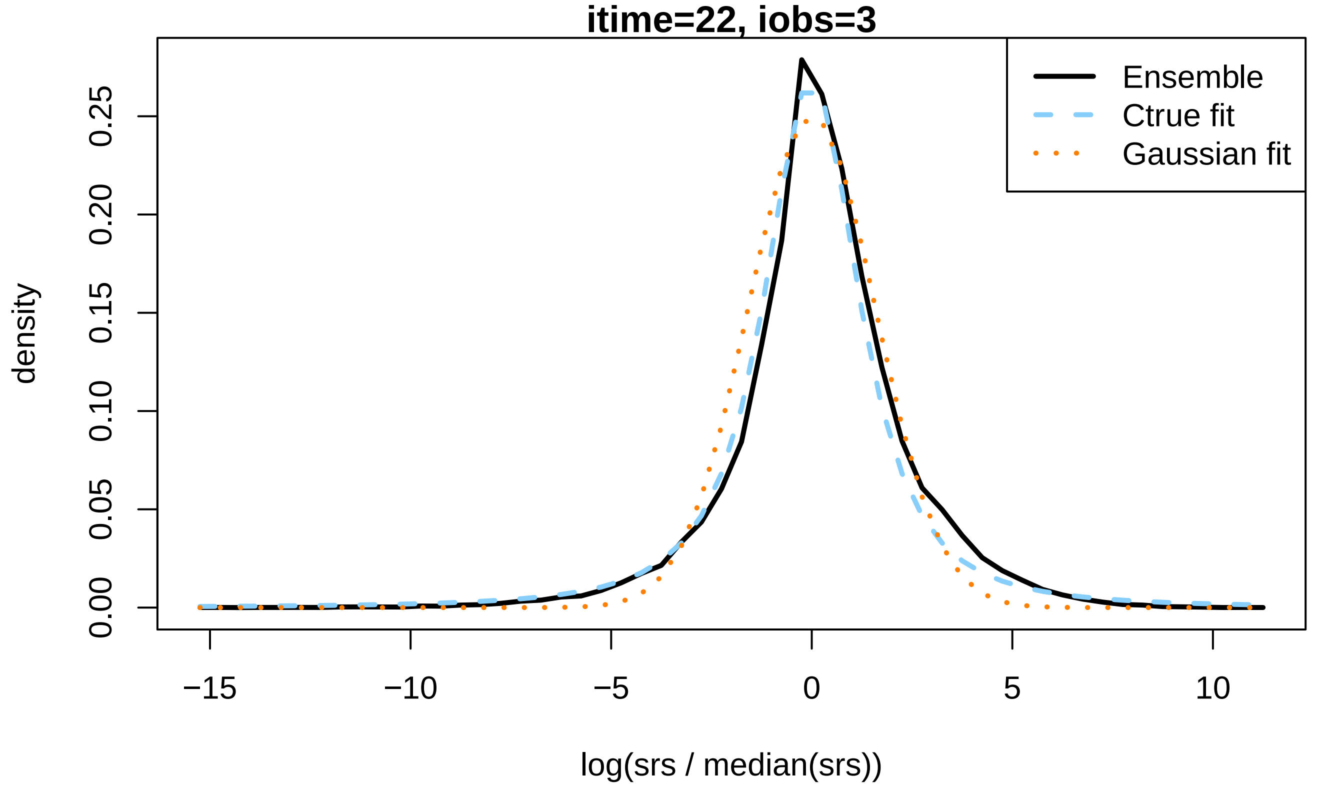


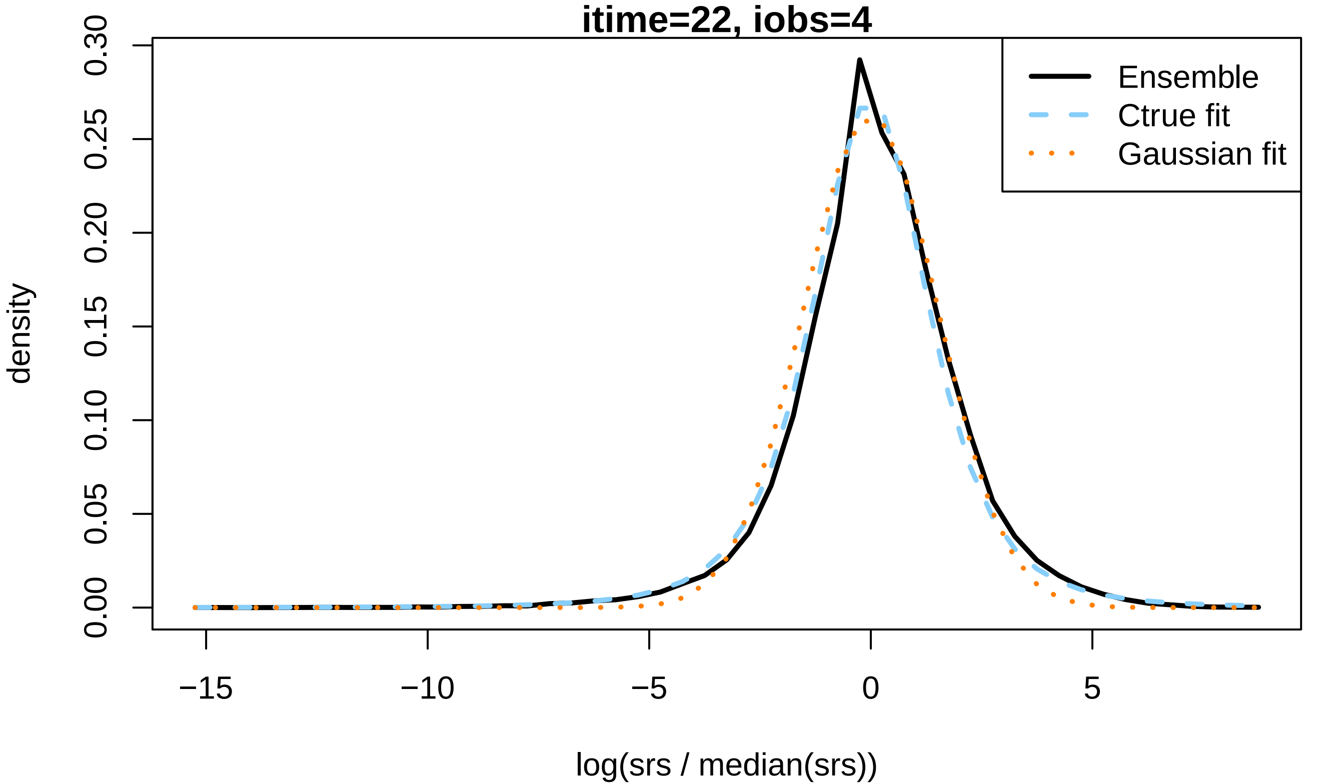
itime=22, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



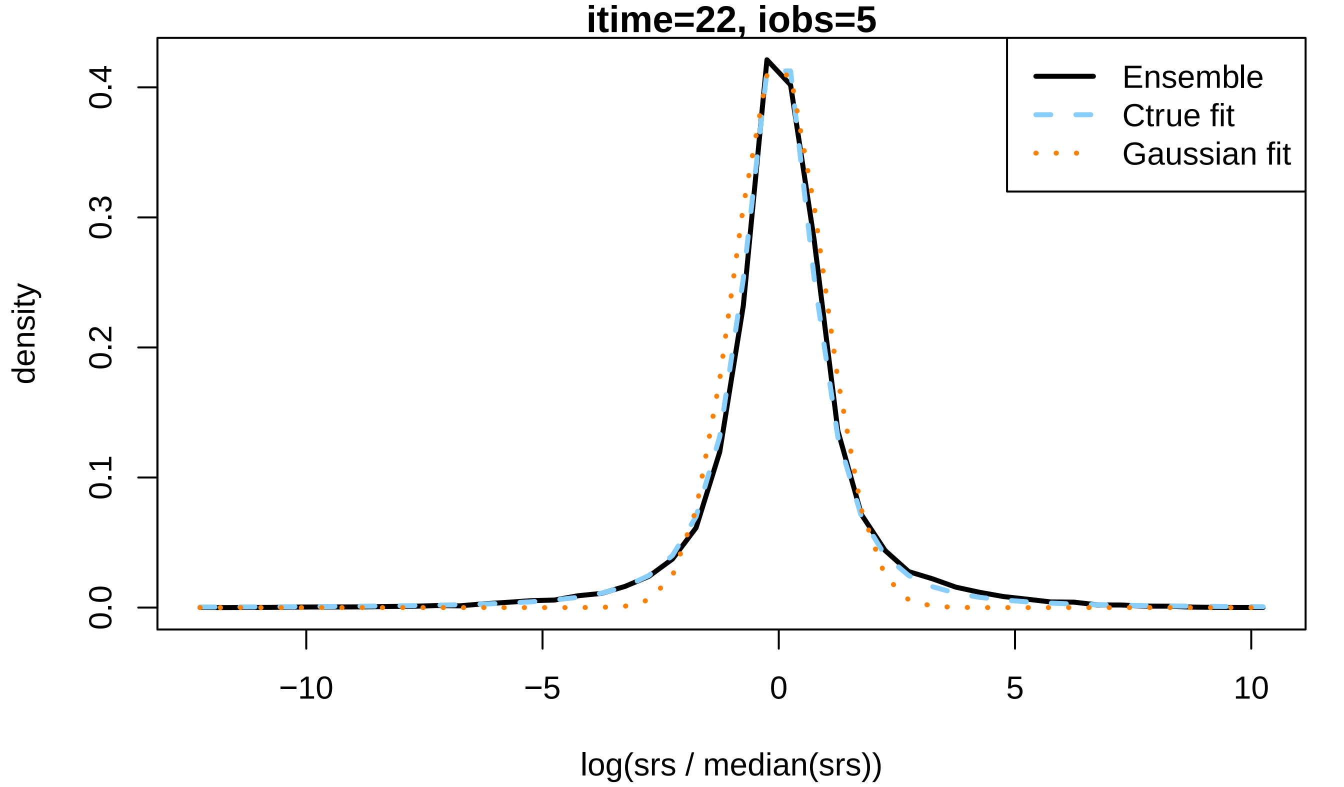
itime=22, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−20
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=22, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

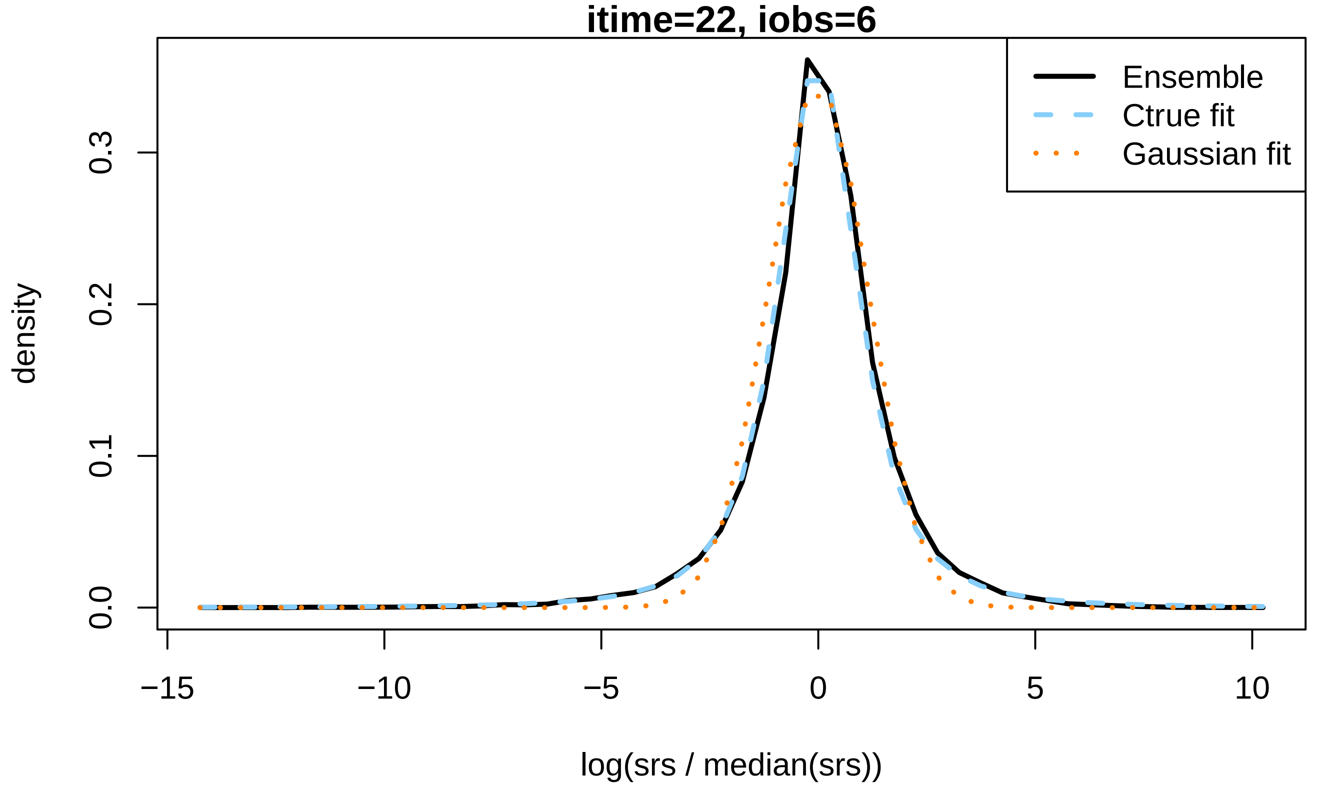

itime=22, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))



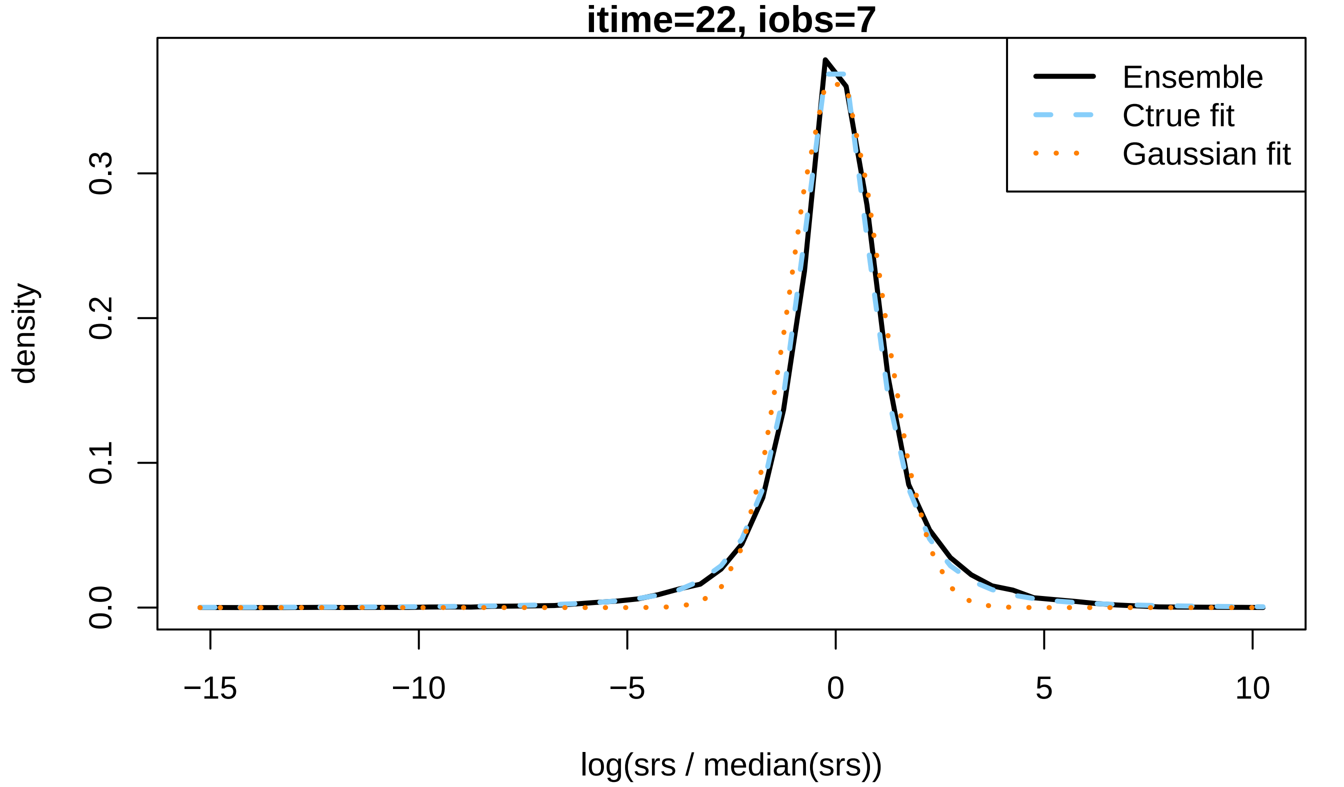
itime=22, iobs=4
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit



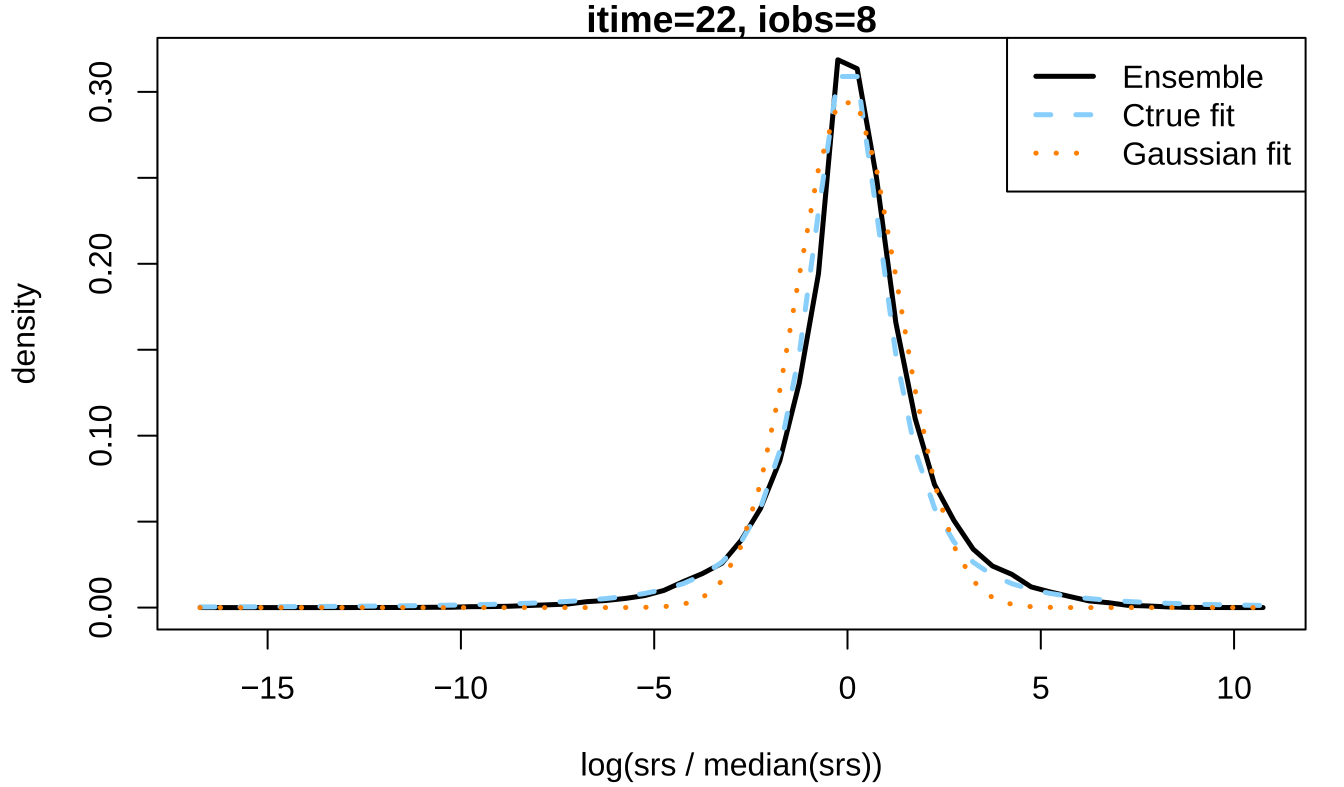
itime=22, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
log(srs / median(srs))



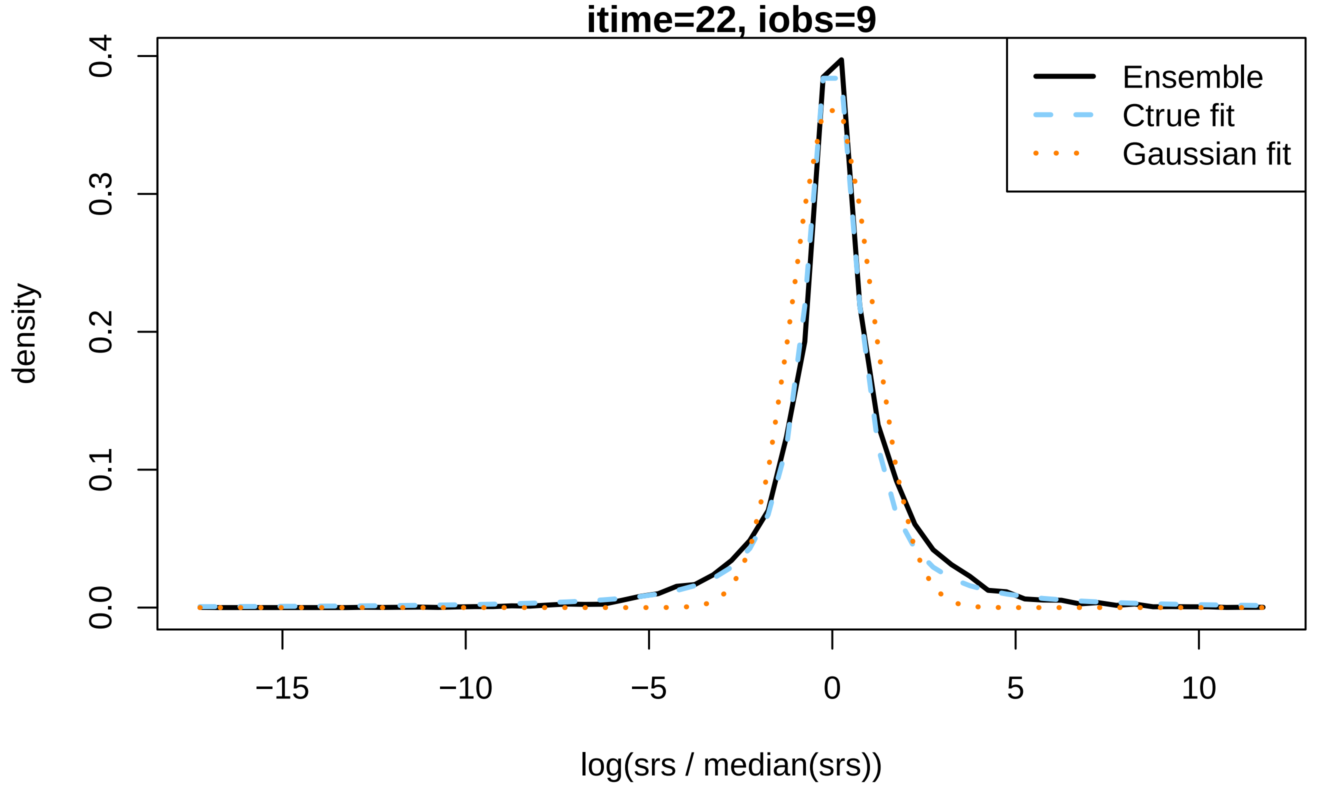
itime=22, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))



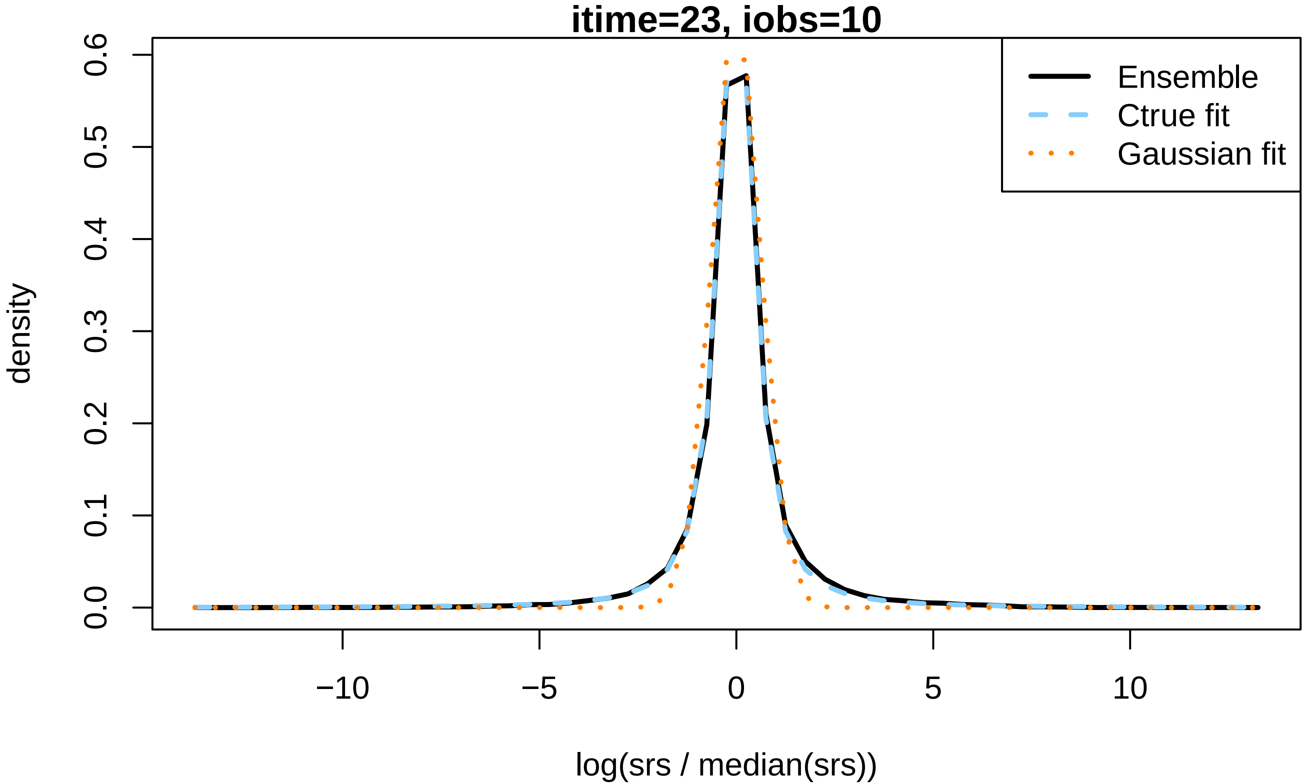
itime=22, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

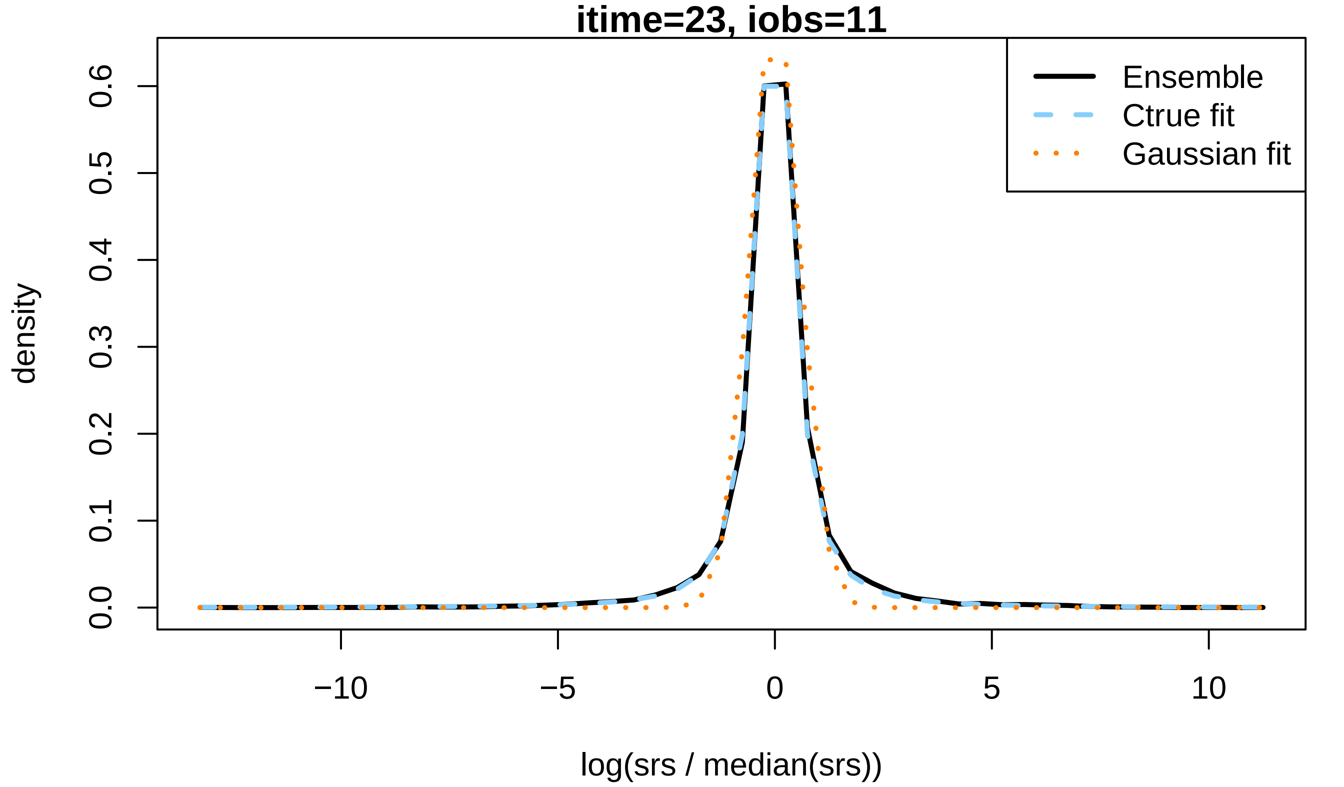


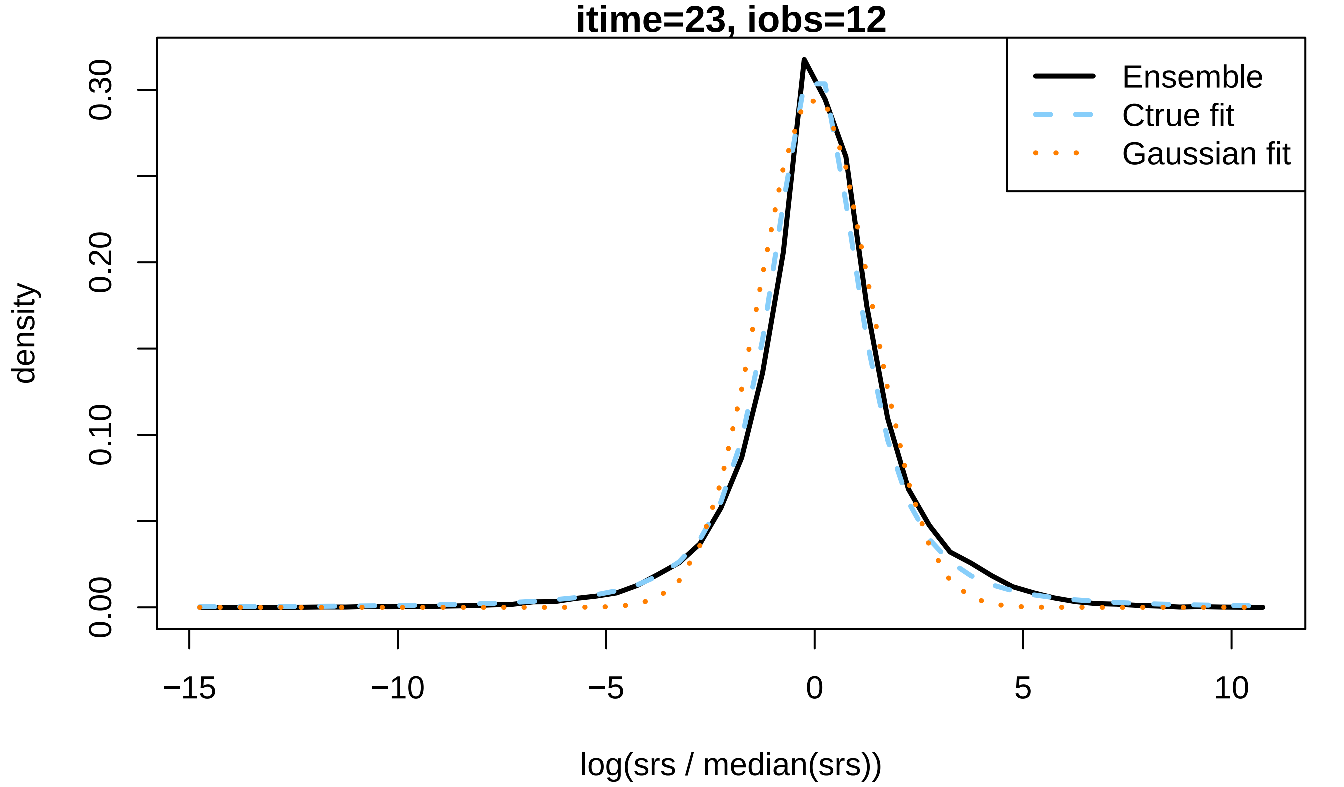
itime=22, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



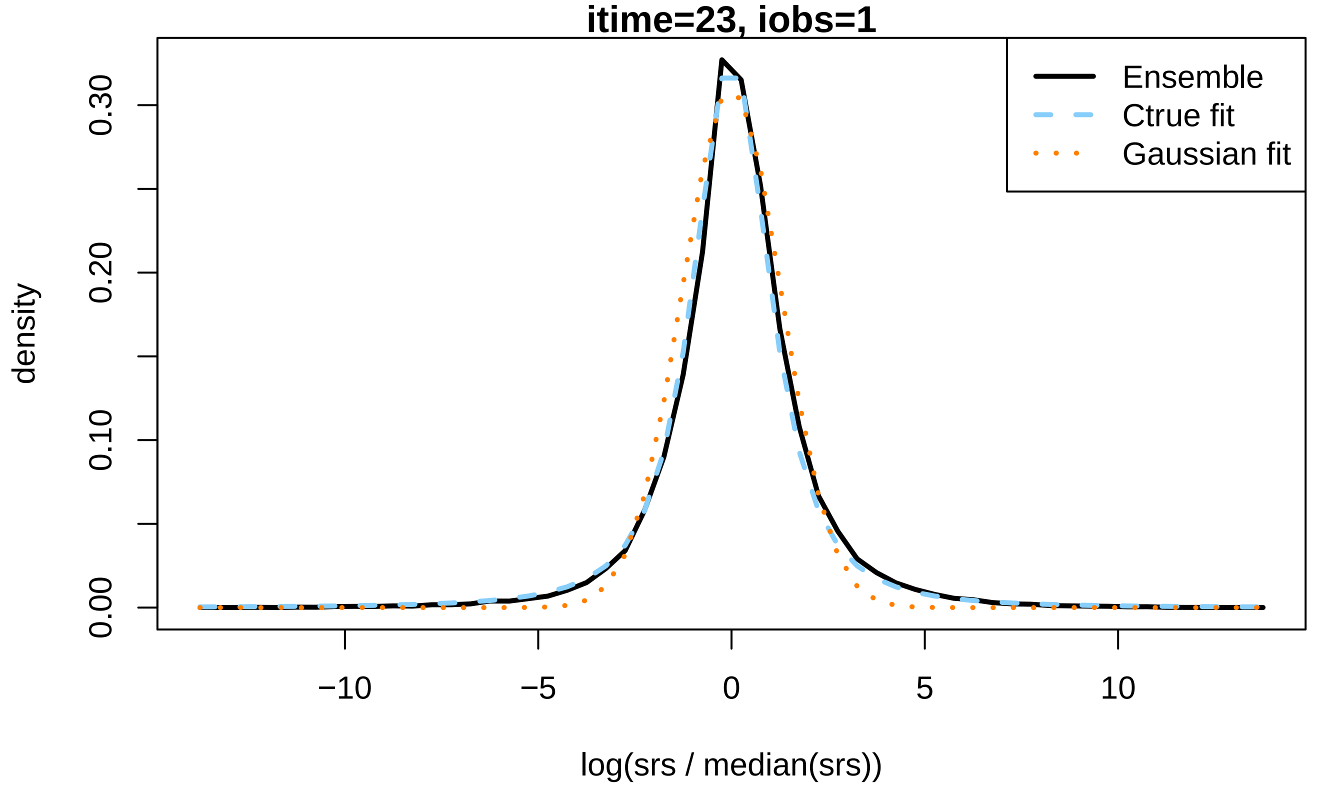
itime=22, iobs=9
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

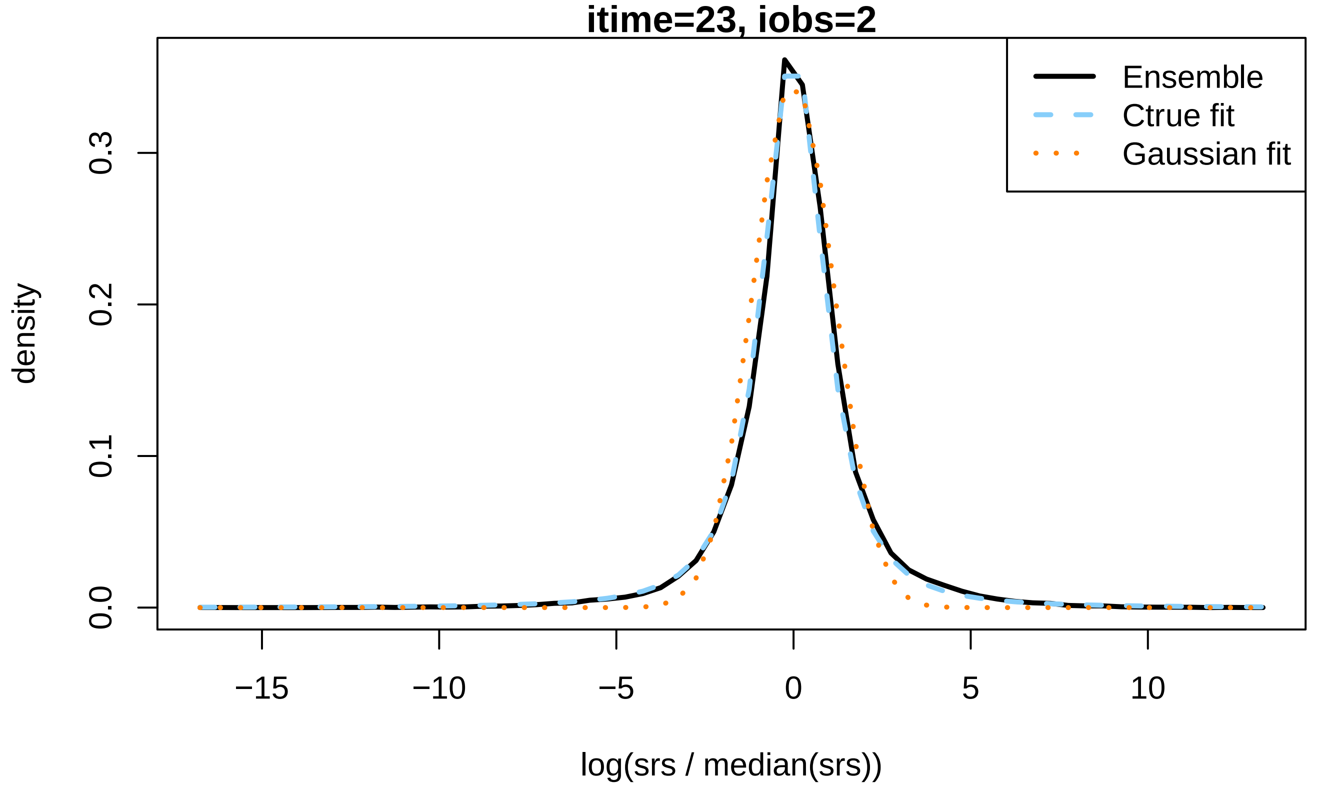

itime=23, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−10
−5
0
5
10
log(srs / median(srs))


itime=23, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−10
−5
0
5
10
log(srs / median(srs))

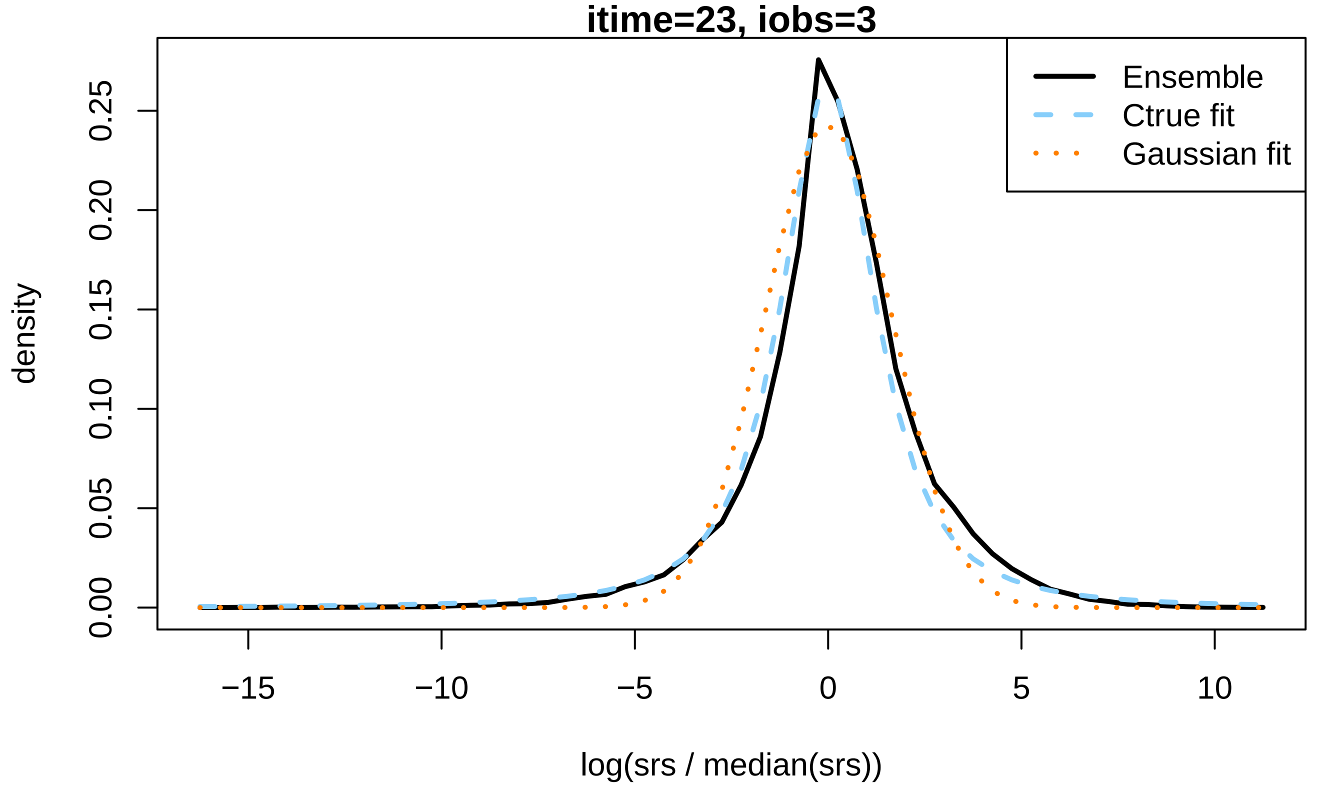


itime=23, iobs=12
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

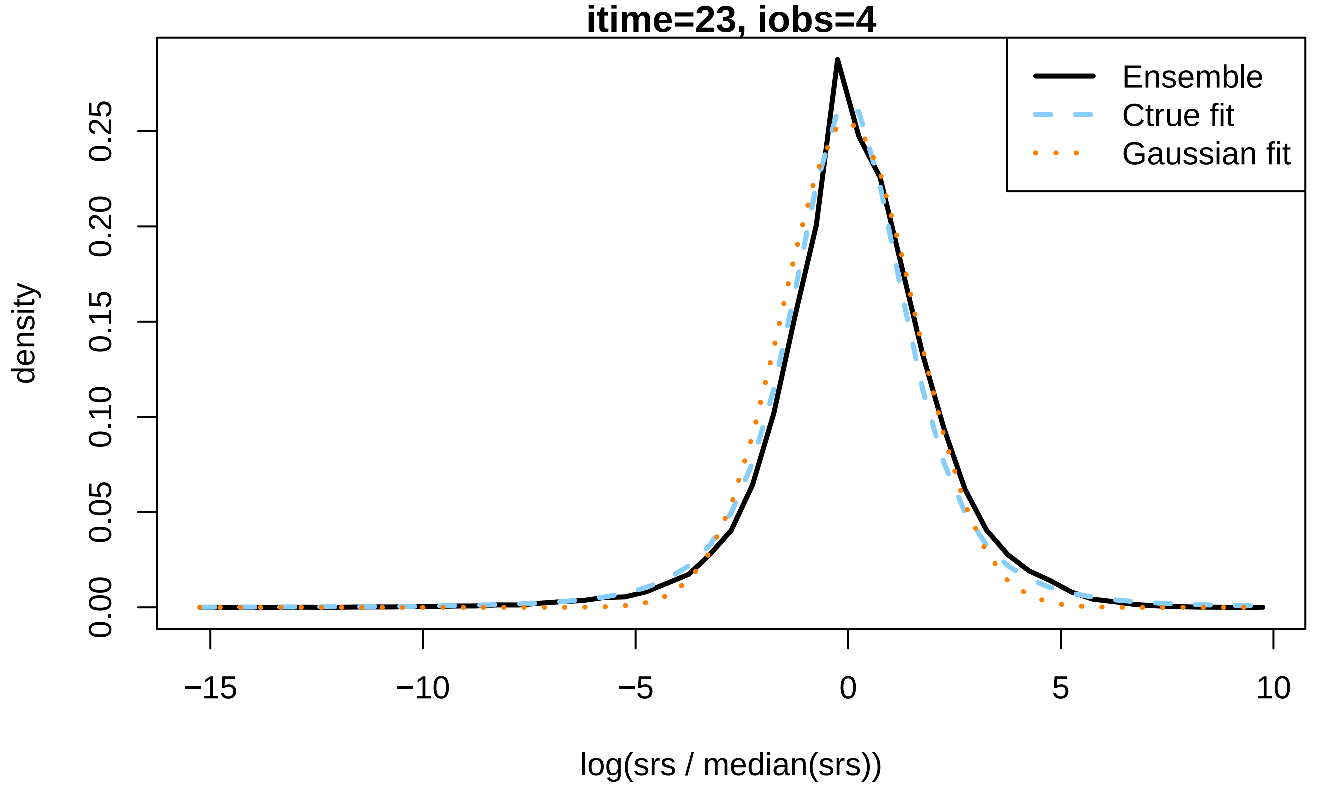

itime=23, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))

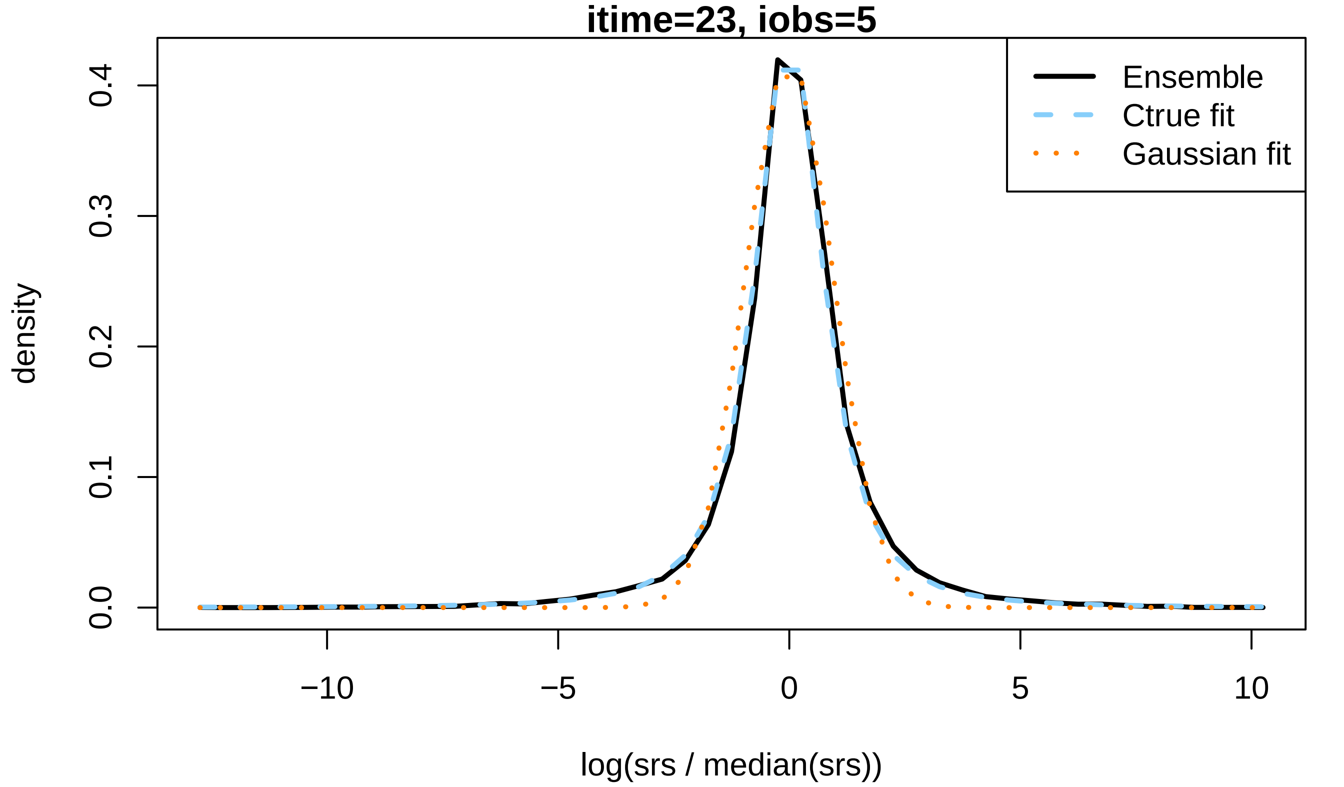


itime=23, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

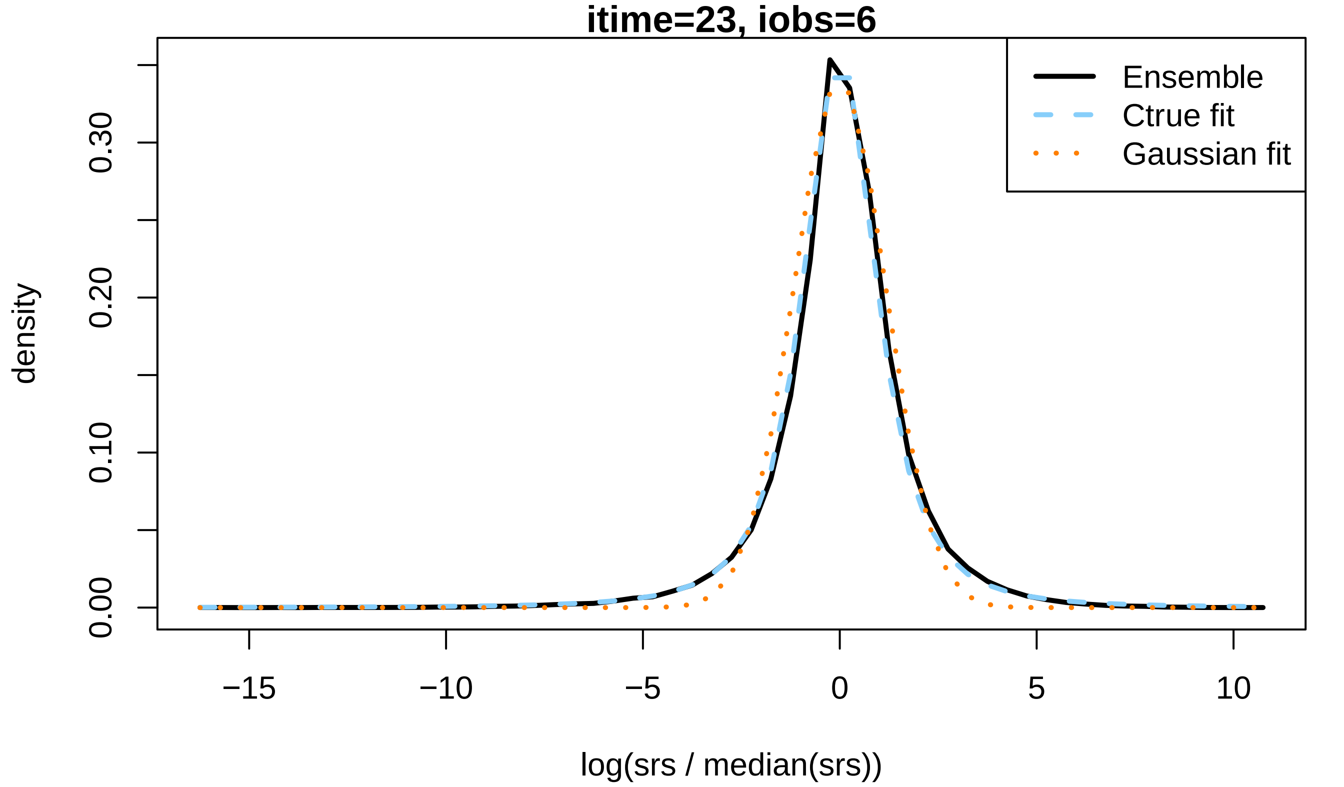


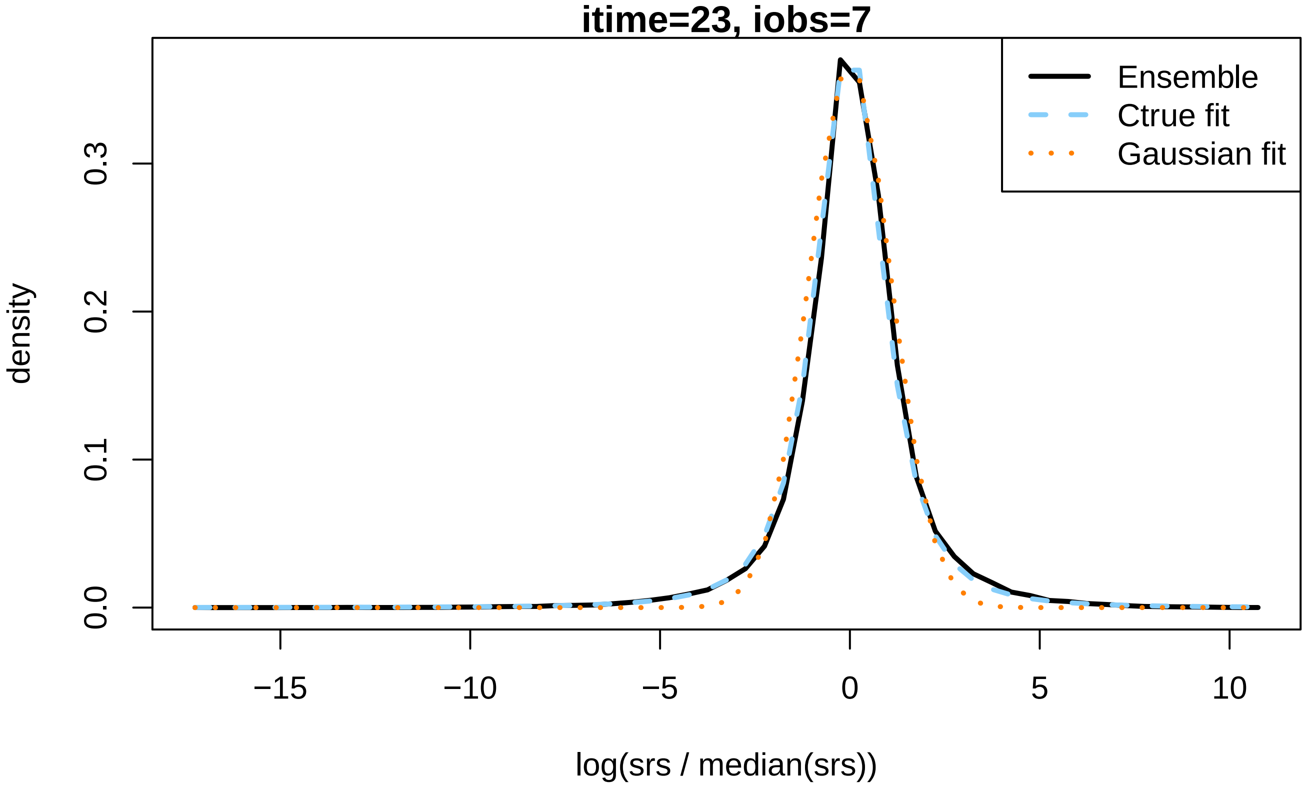
itime=23, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=23, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

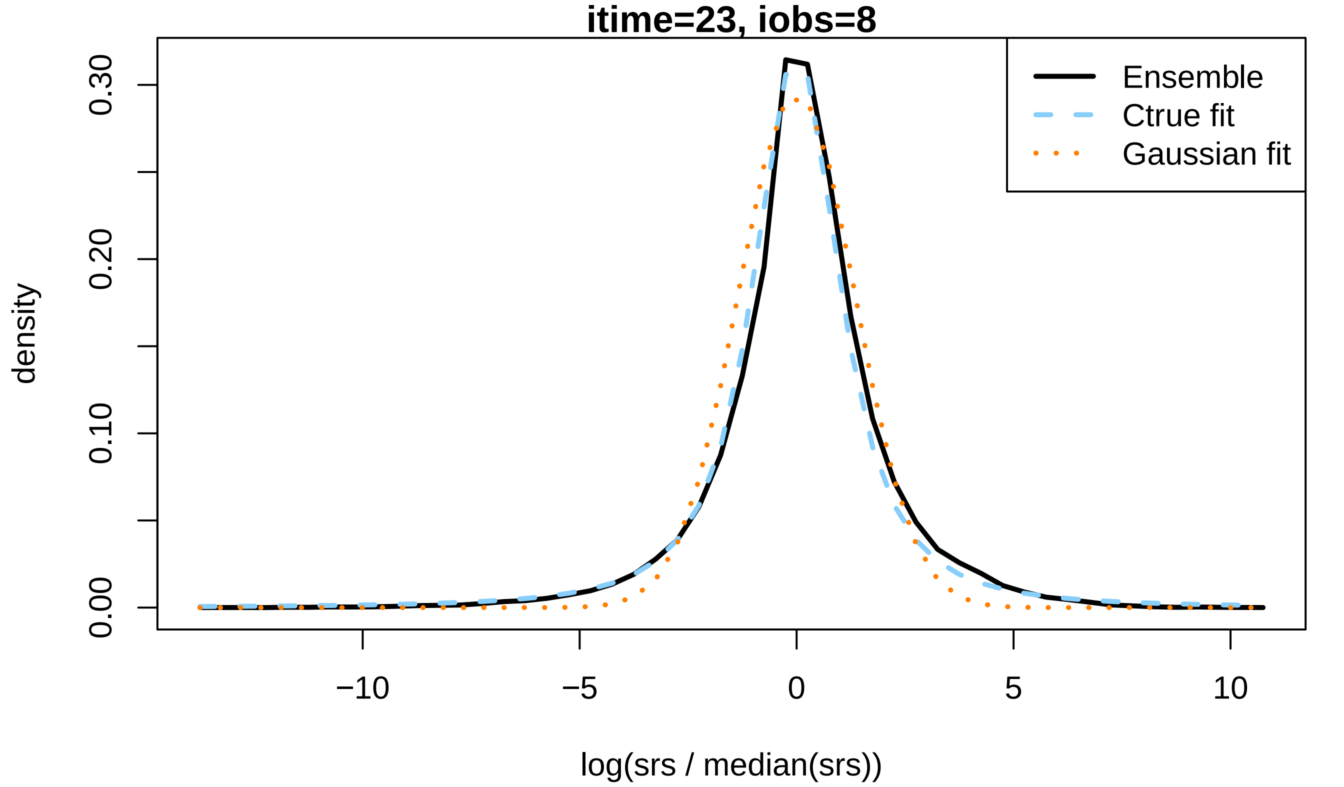


itime=23, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
log(srs / median(srs))

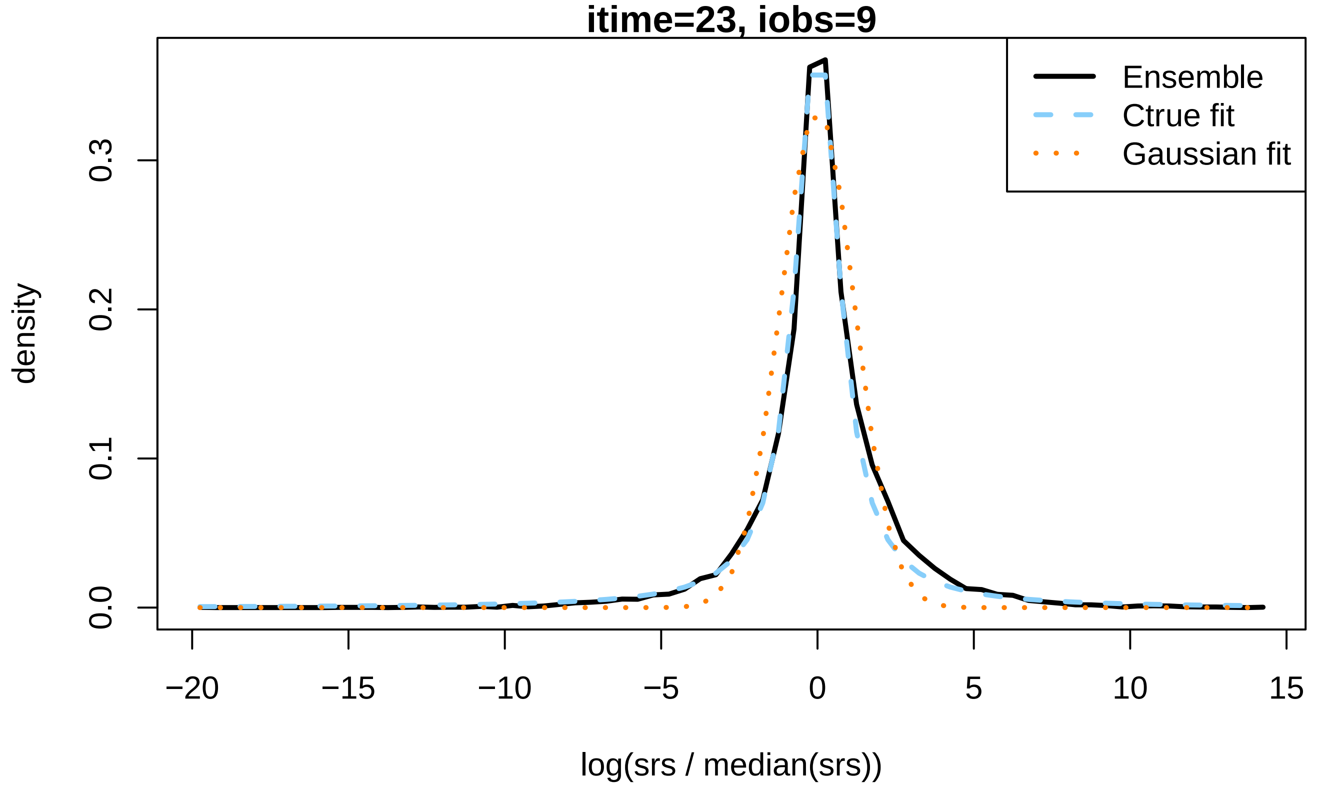

itime=23, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

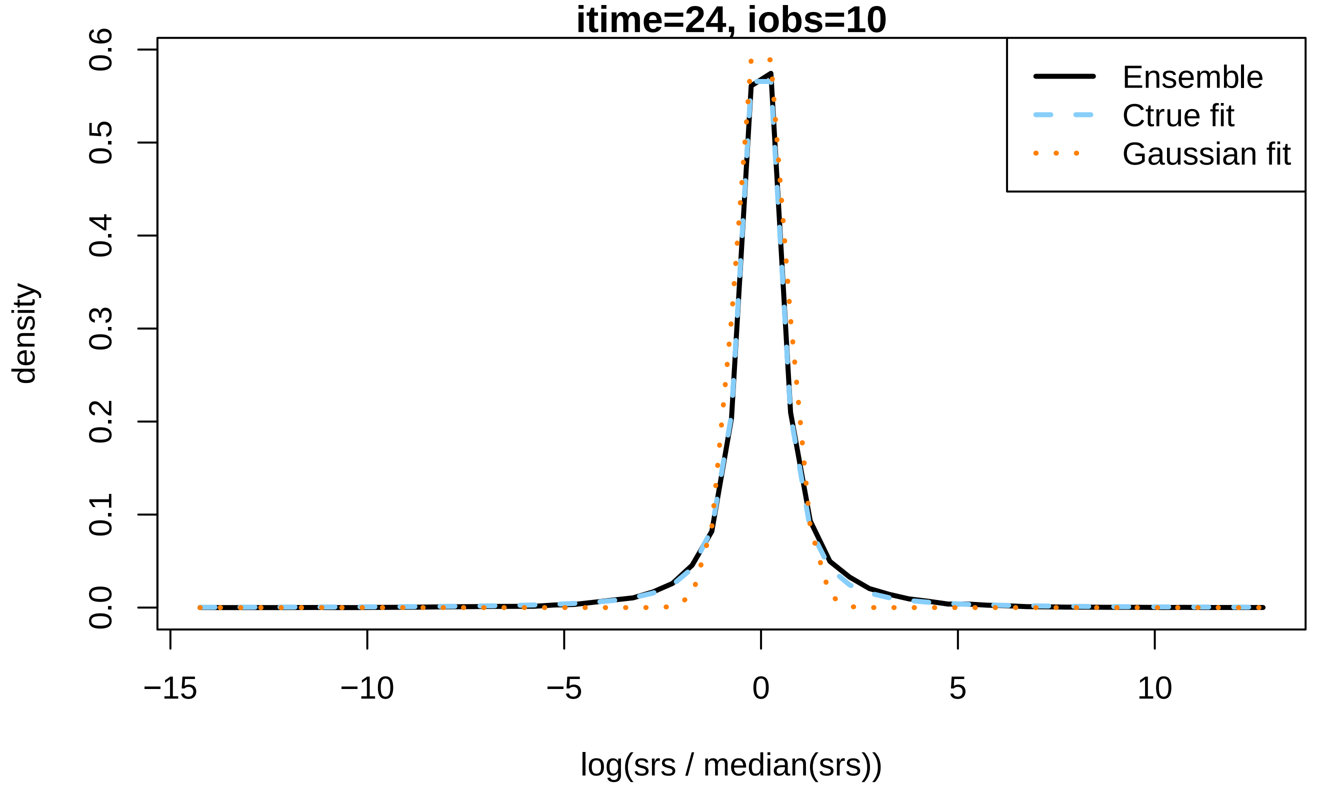


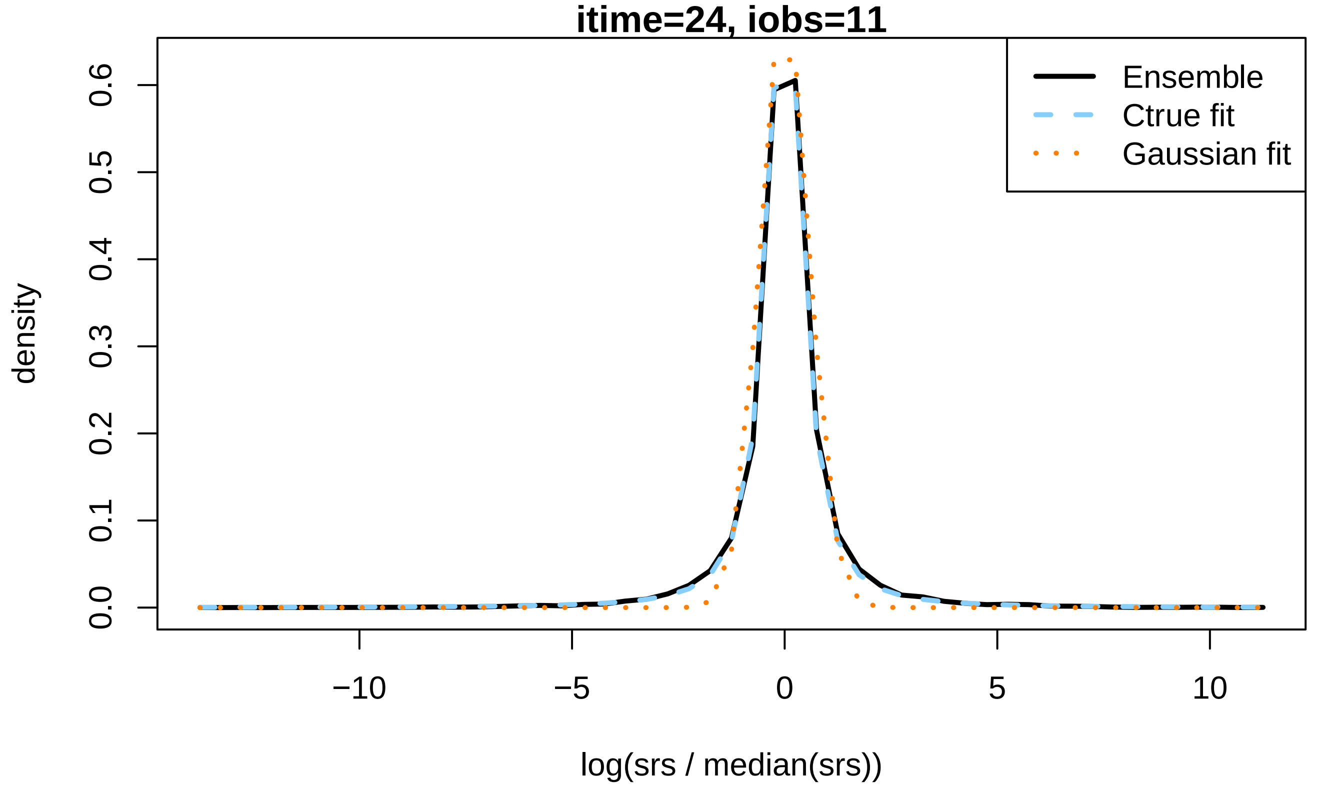
itime=23, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

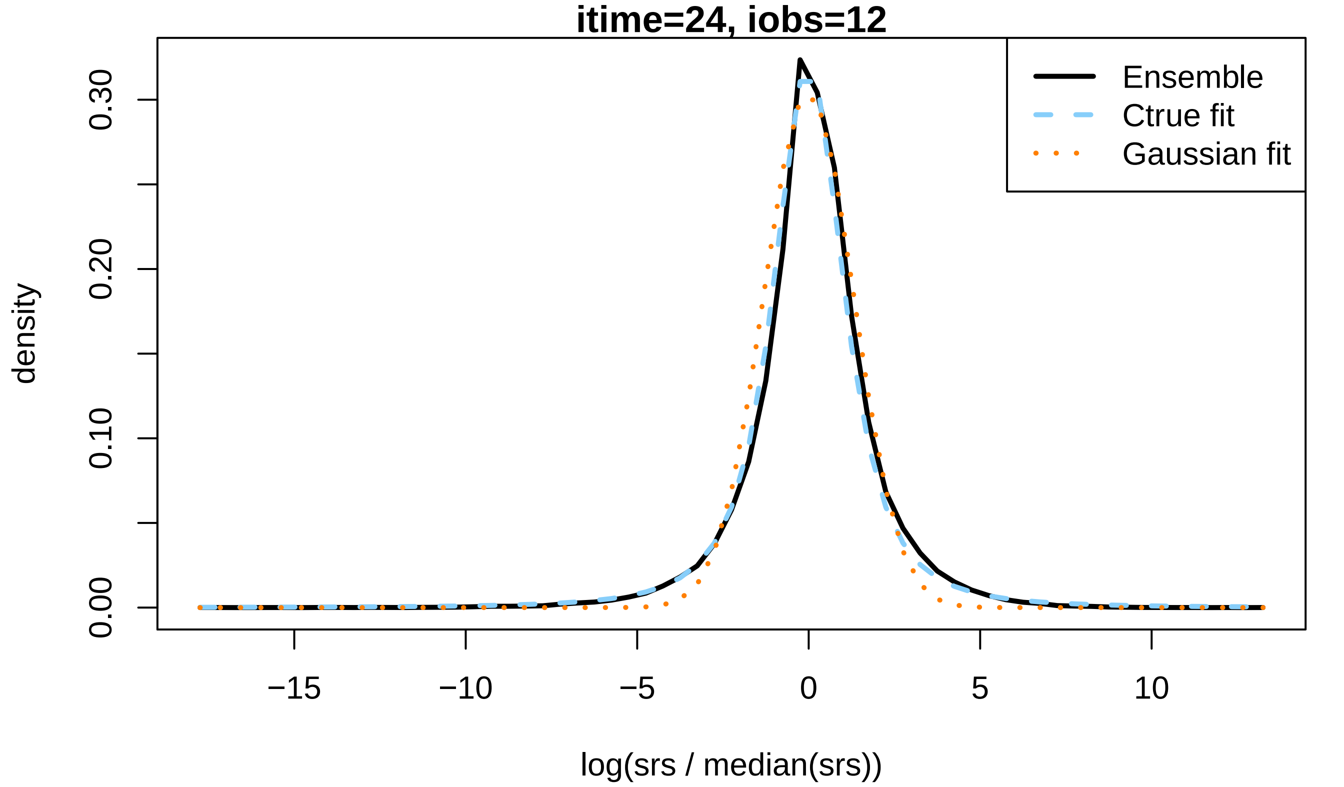


itime=23, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))

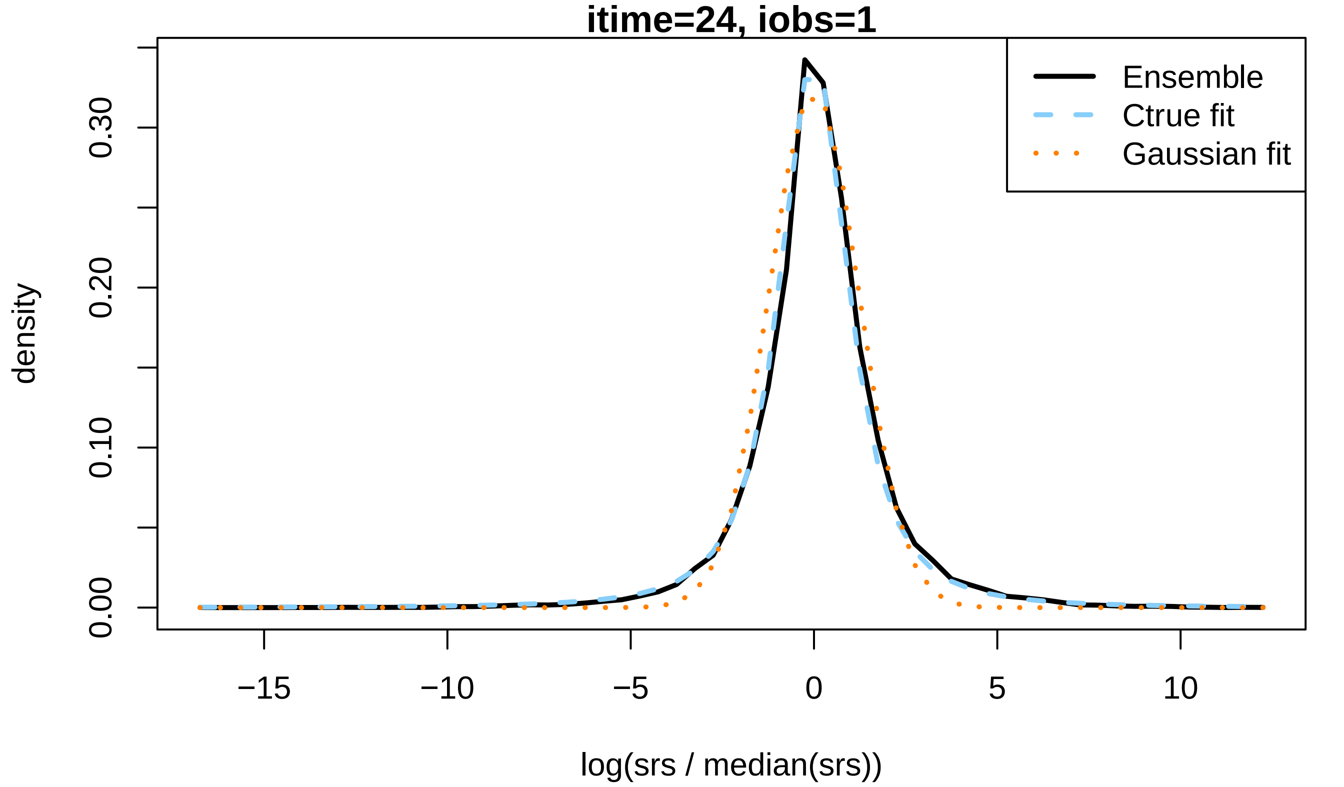

itime=23, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−20
−15
−10
−5
0
5
10
15
log(srs / median(srs))

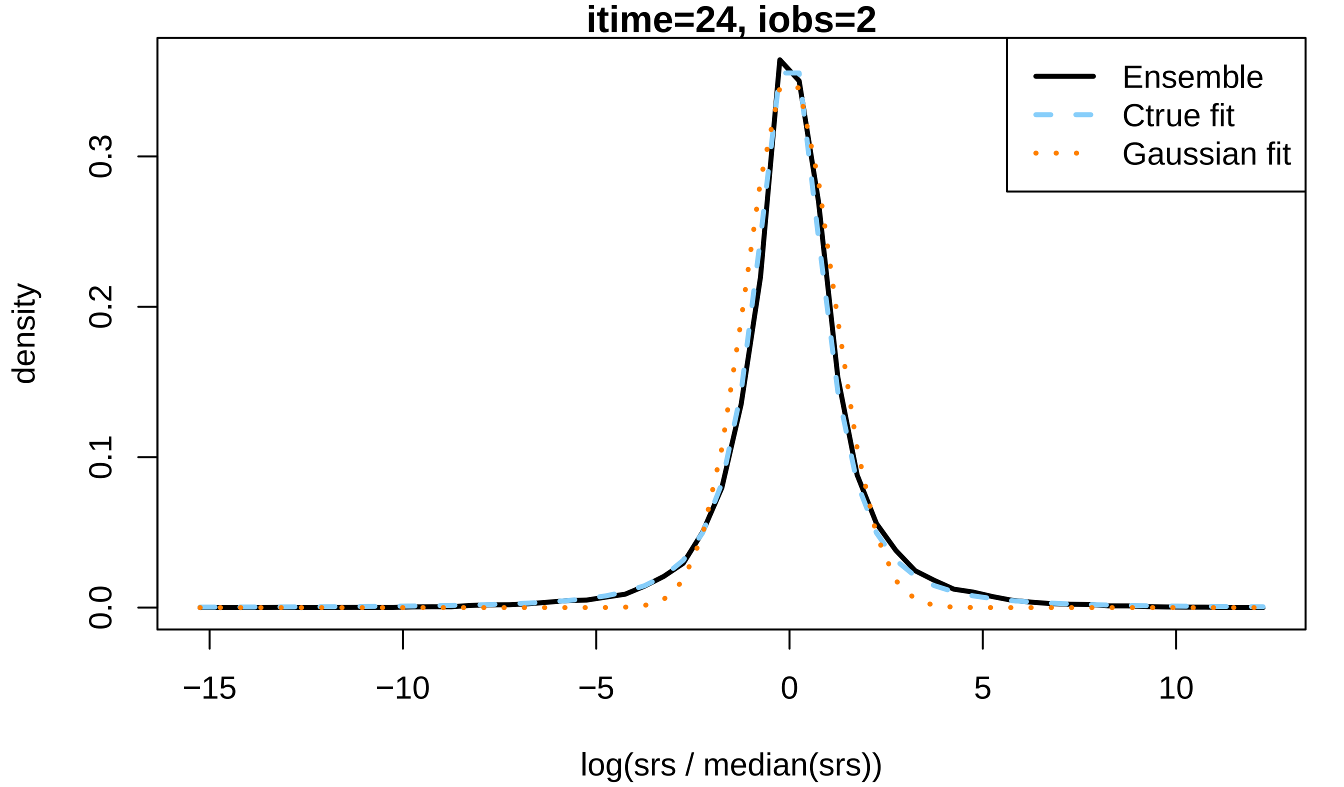

itime=24, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−15
−10
−5
0
5
10
log(srs / median(srs))

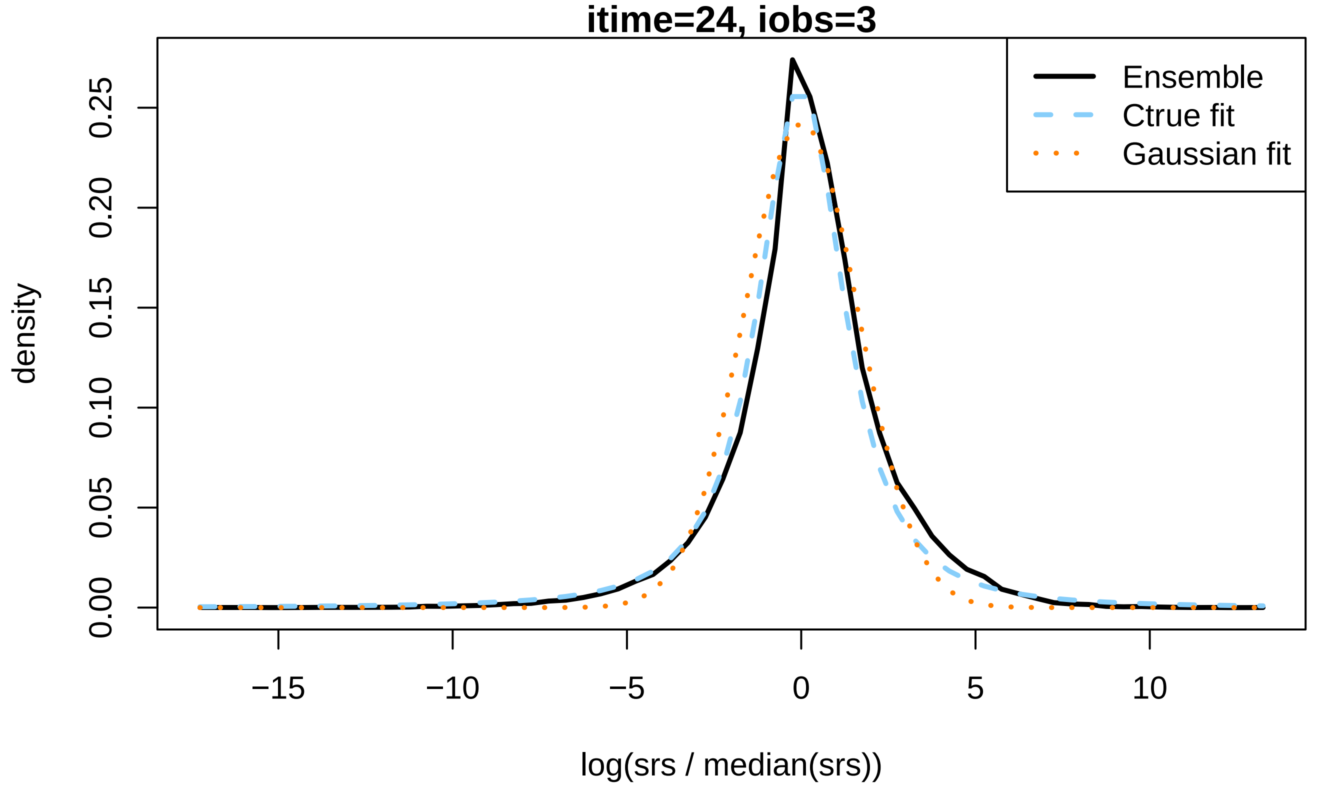

itime=24, iobs=11
density
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit
0.0
0.1
0.2
0.3
0.4
0.5
0.6
−10
−5
0
5
10



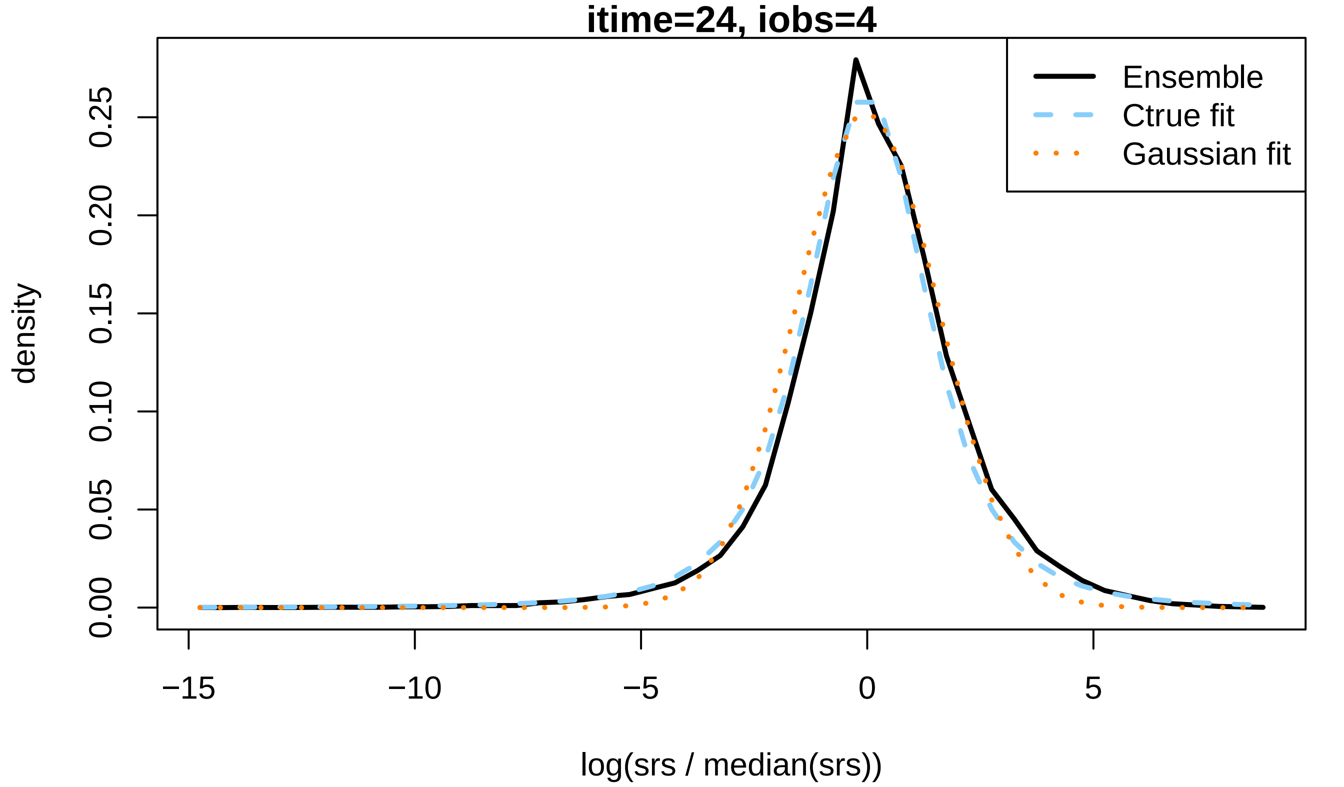
itime=24, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

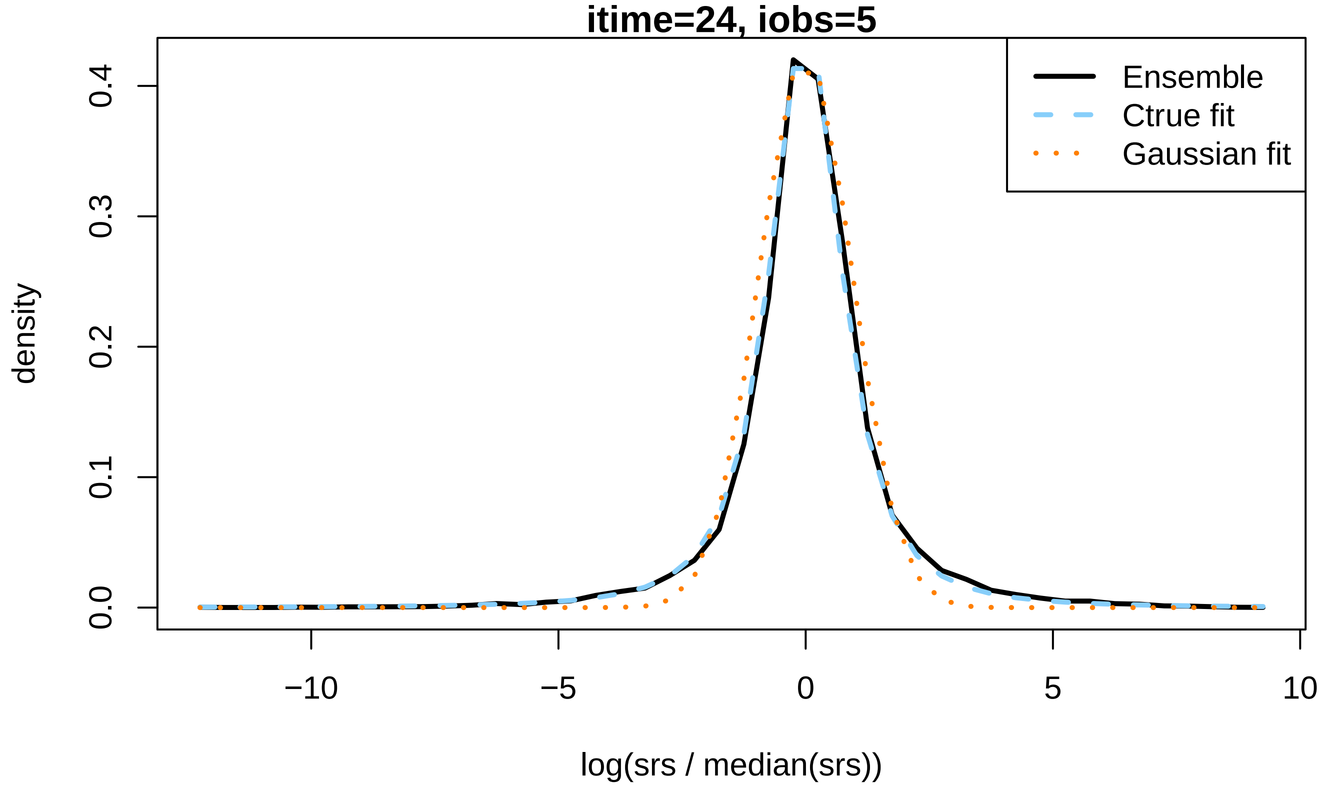

itime=24, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

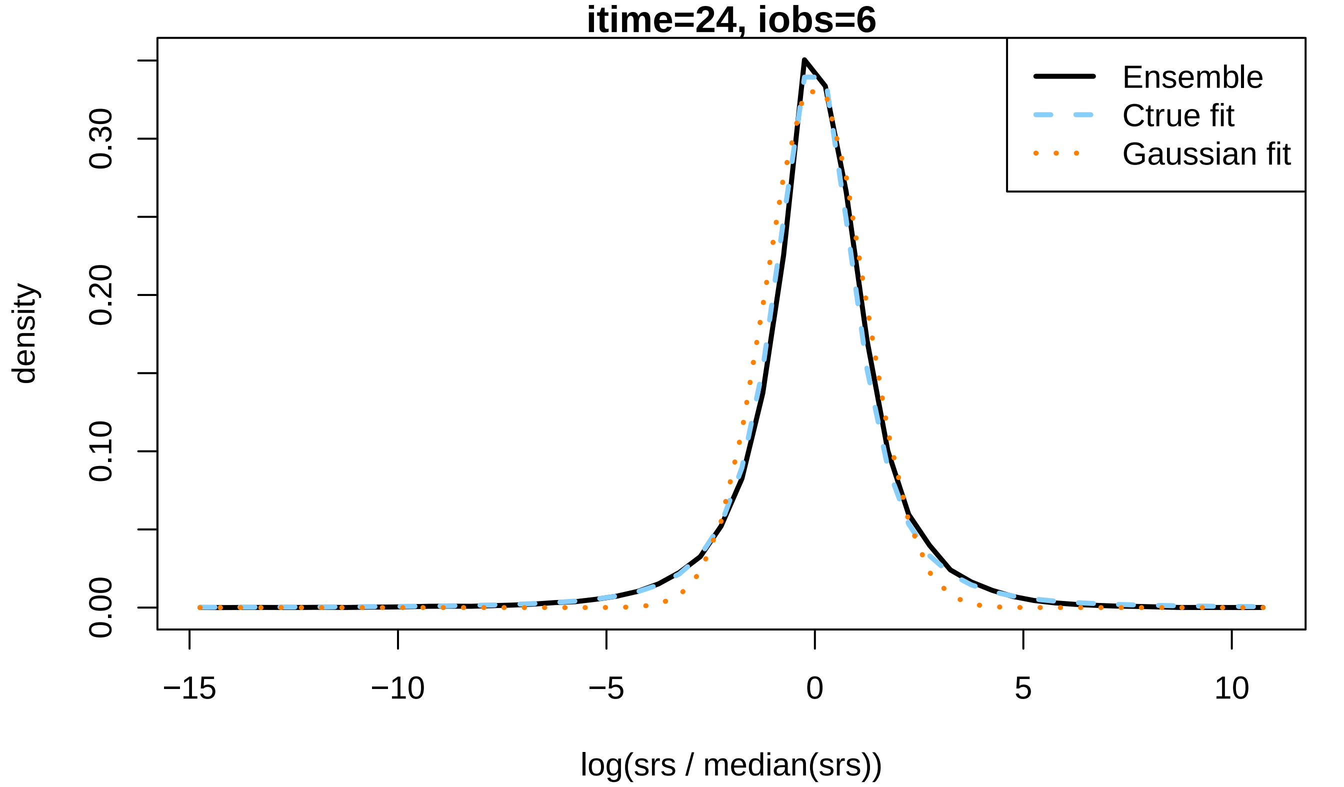

itime=24, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

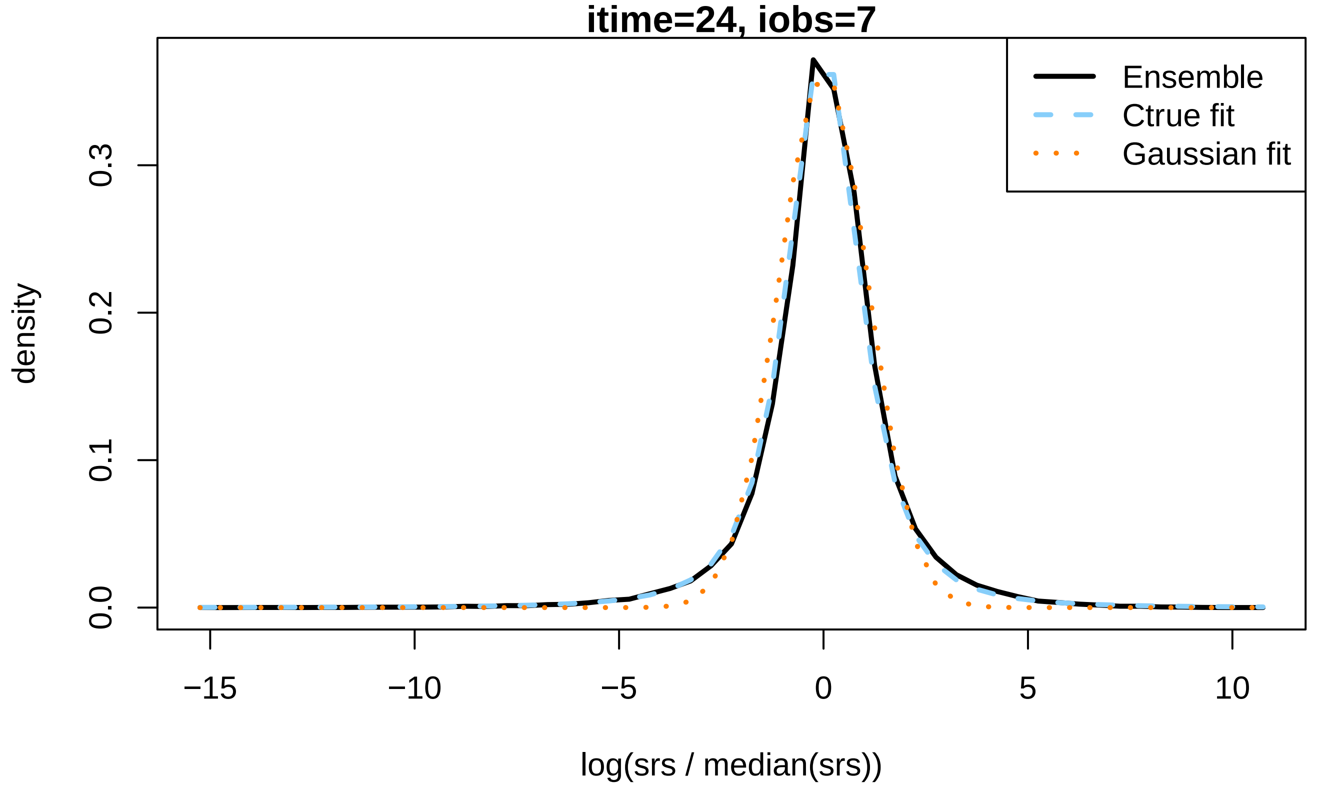


itime=24, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

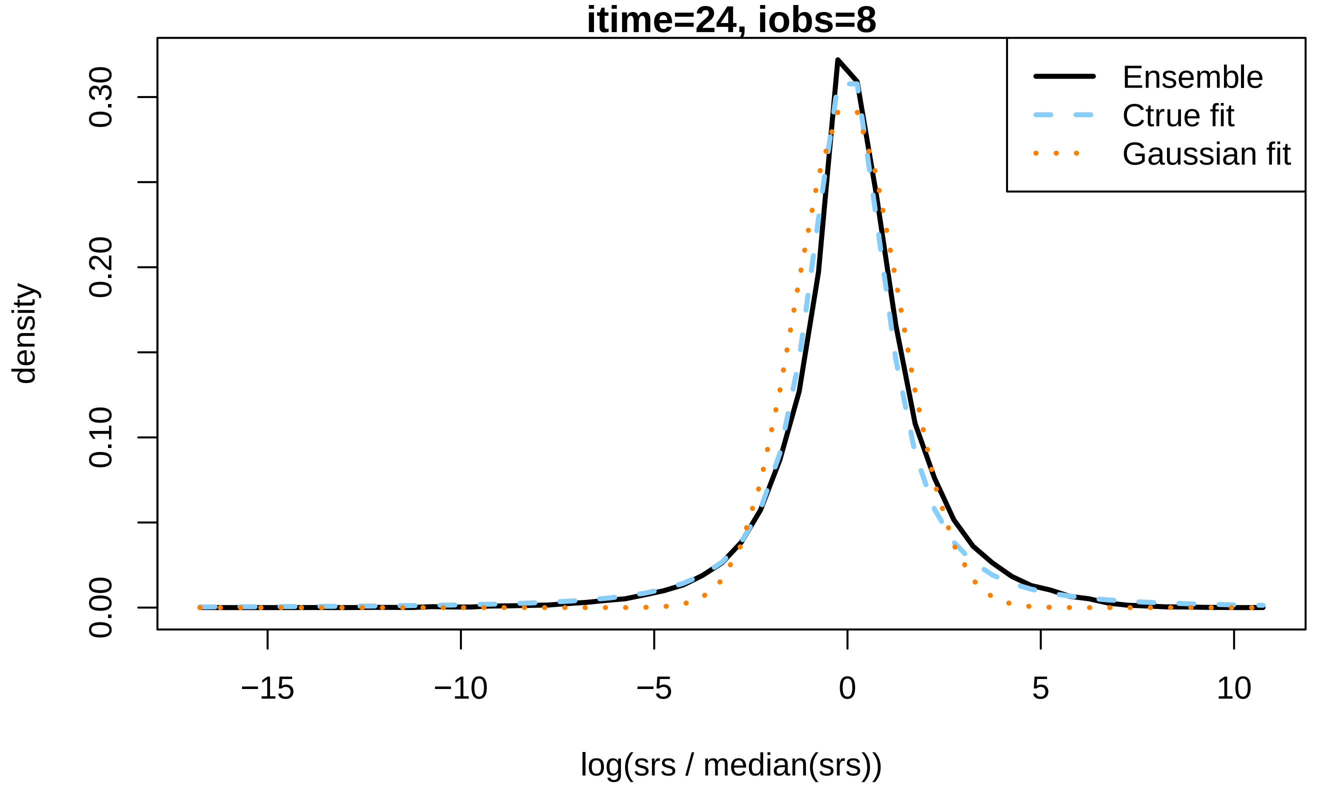

itime=24, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
log(srs / median(srs))

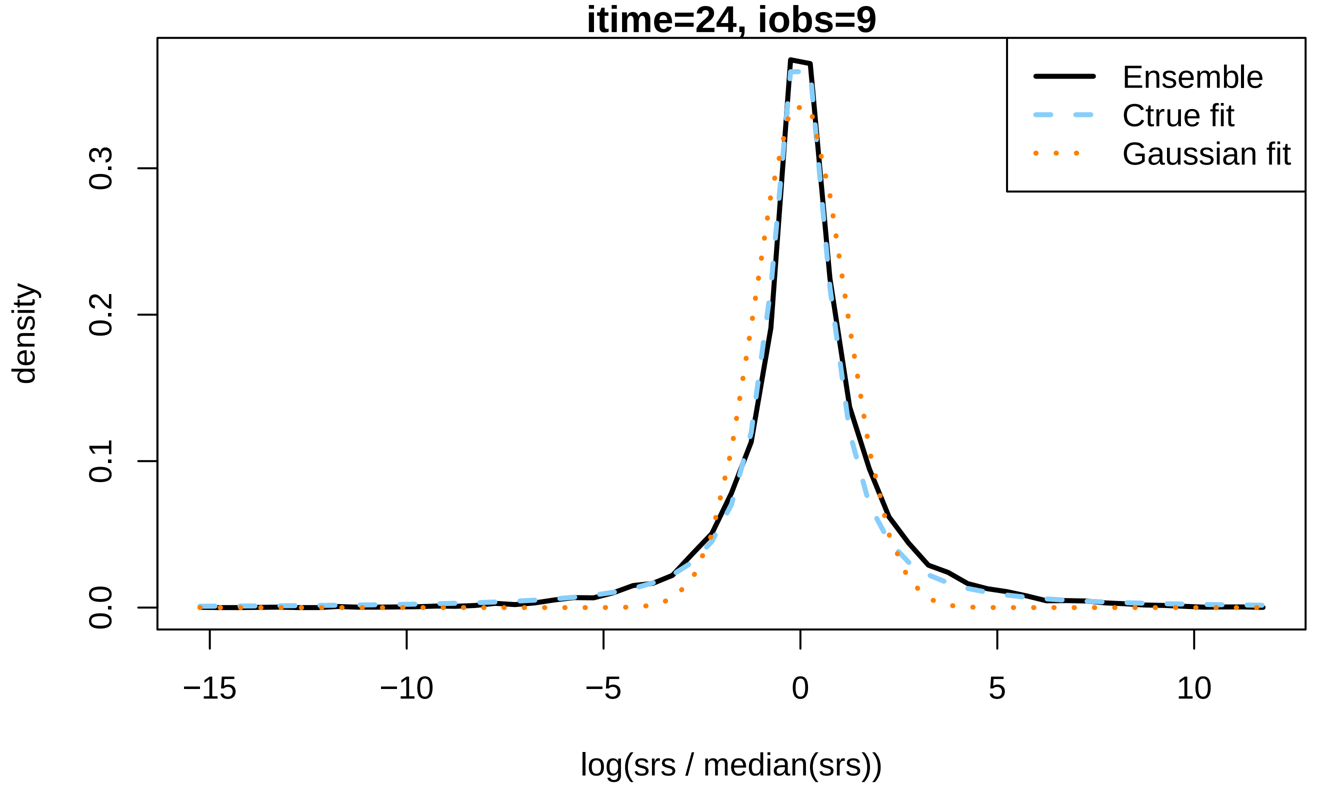

itime=24, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
log(srs / median(srs))


itime=24, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

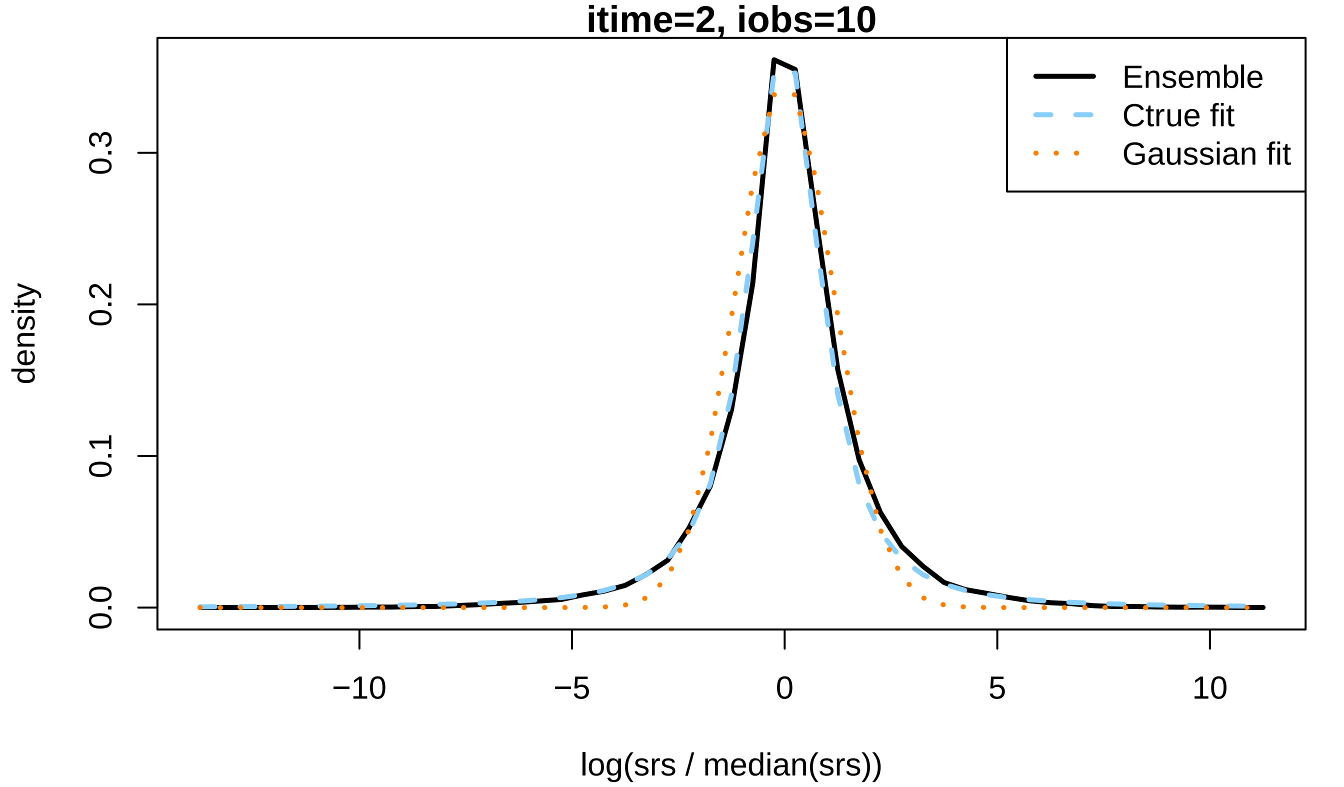


itime=24, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

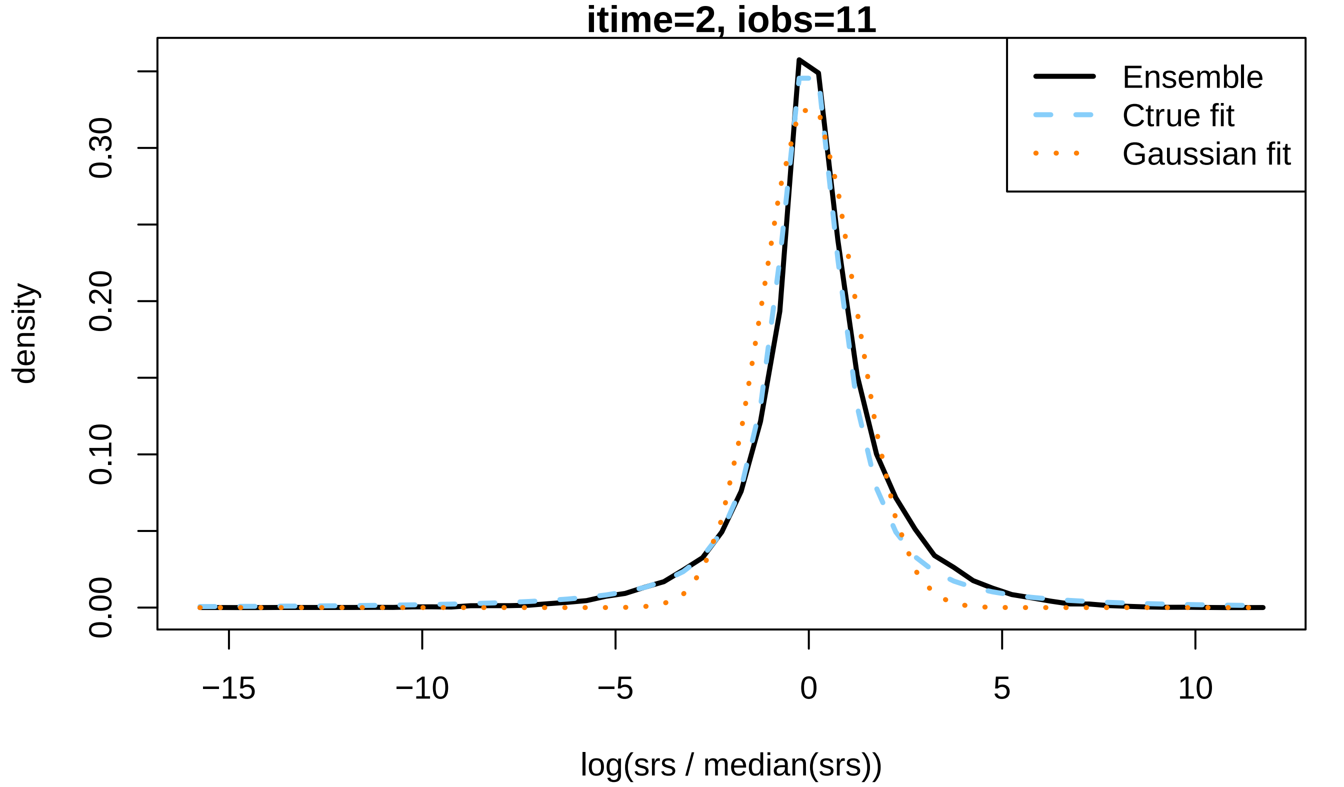

itime=24, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

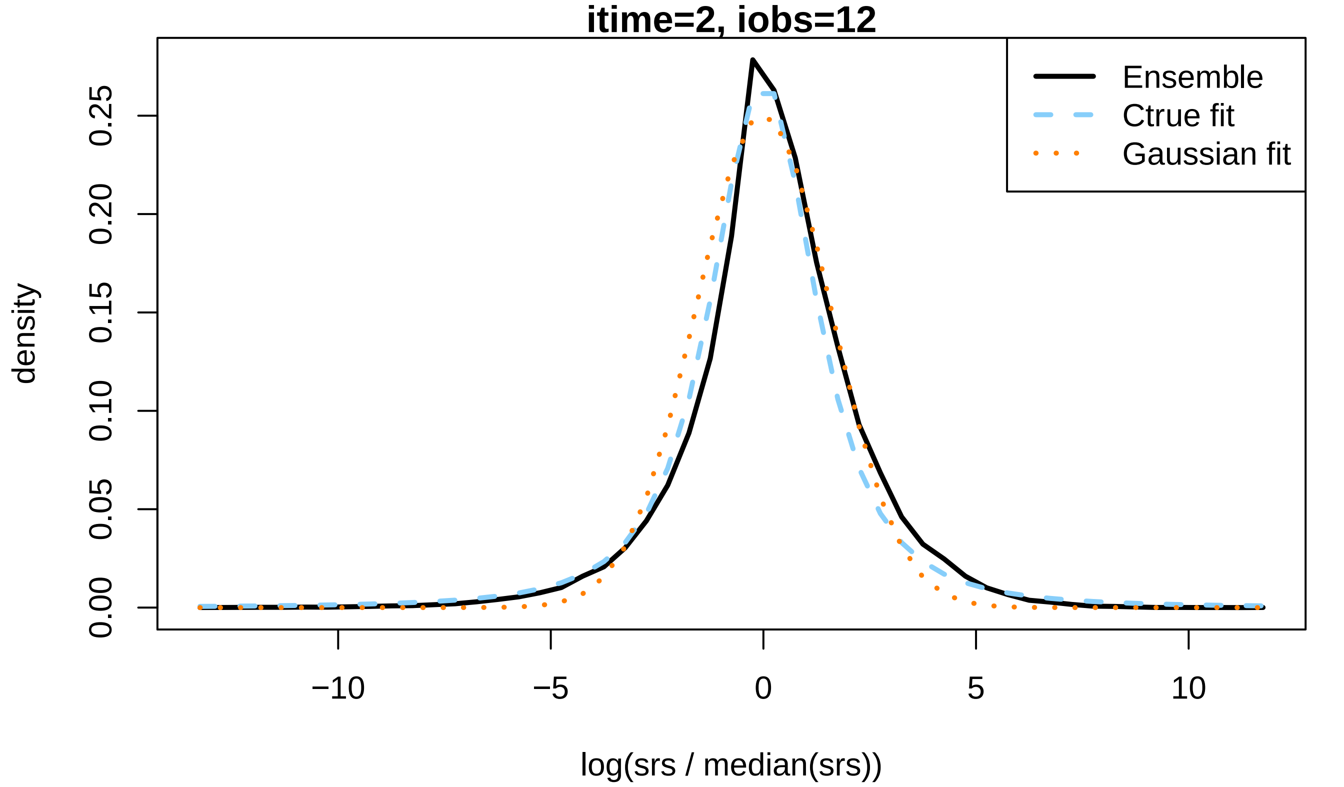


itime=24, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

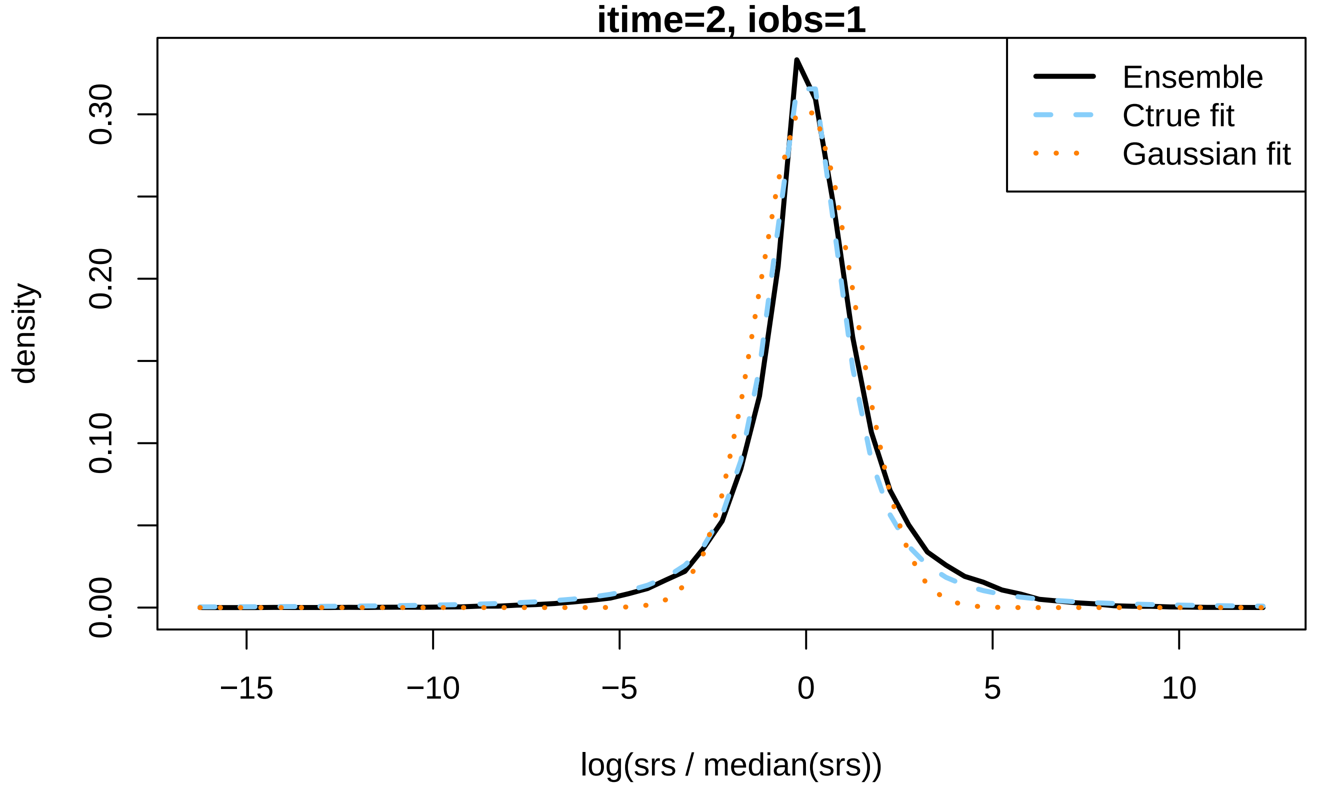


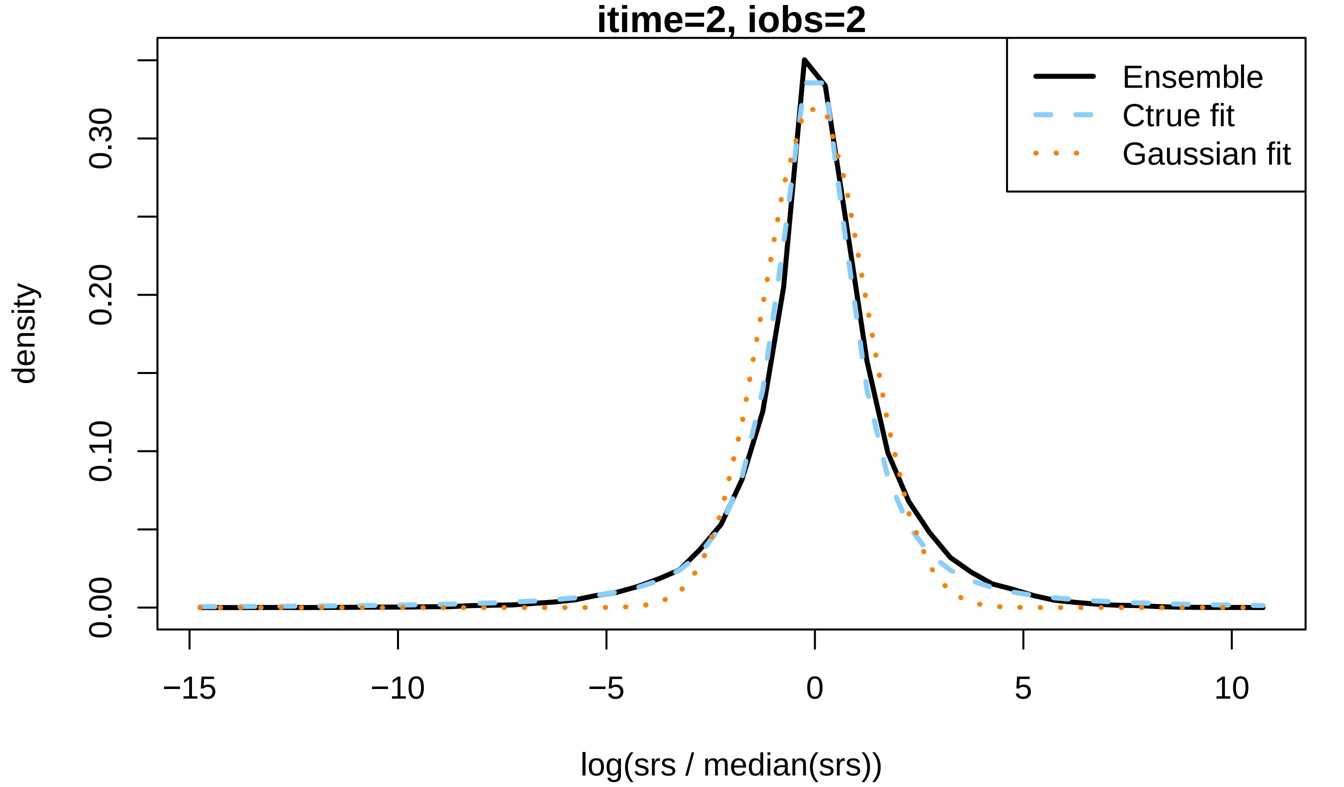
itime=2, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−10
−5
0
5
10
log(srs / median(srs))

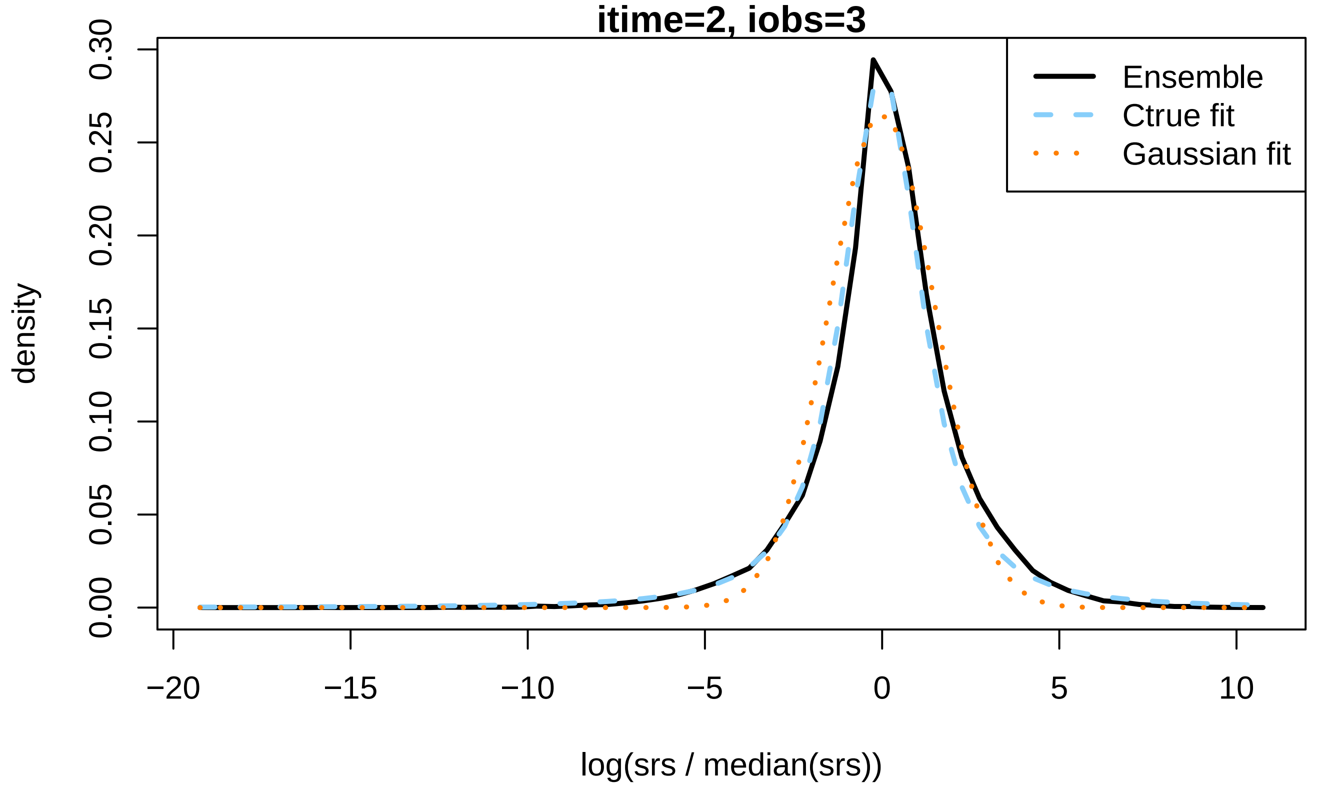

itime=2, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

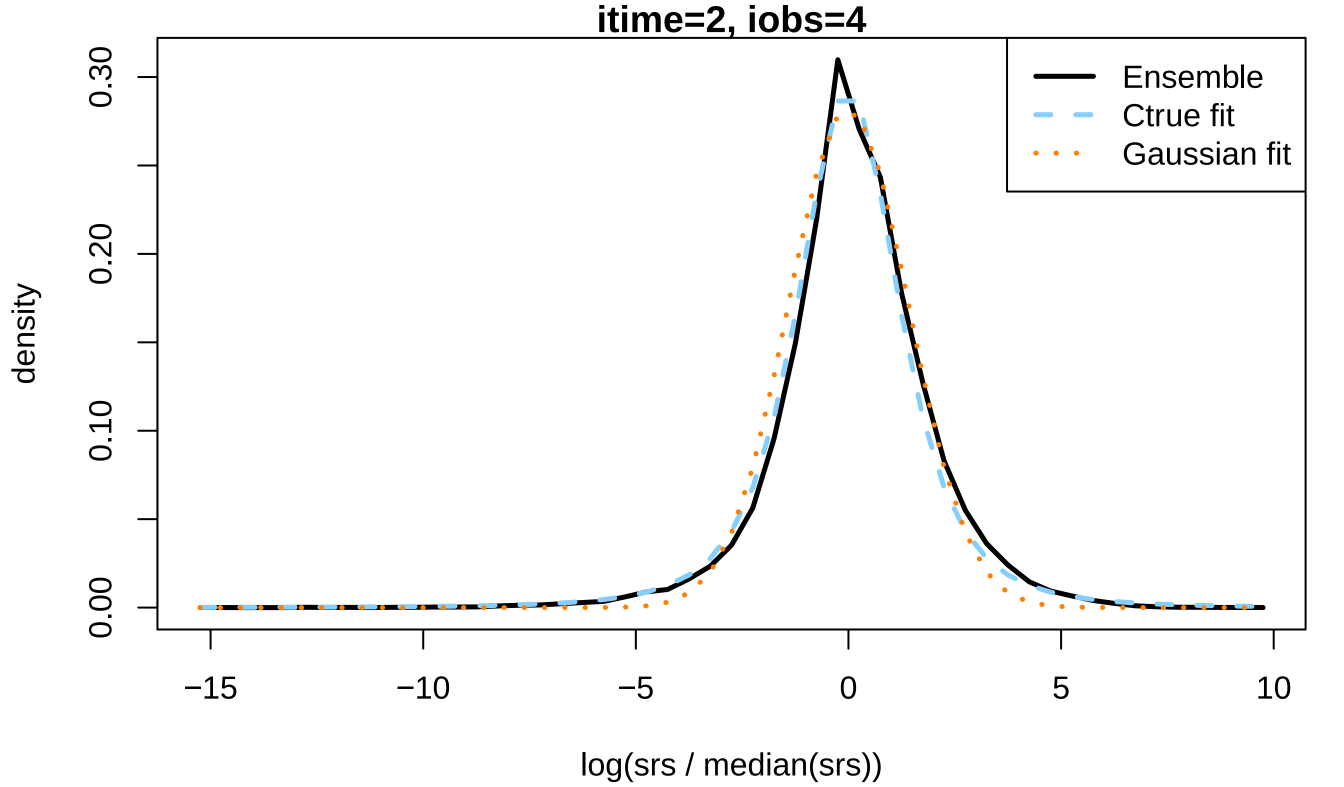


itime=2, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−10
−5
0
5
10
log(srs / median(srs))

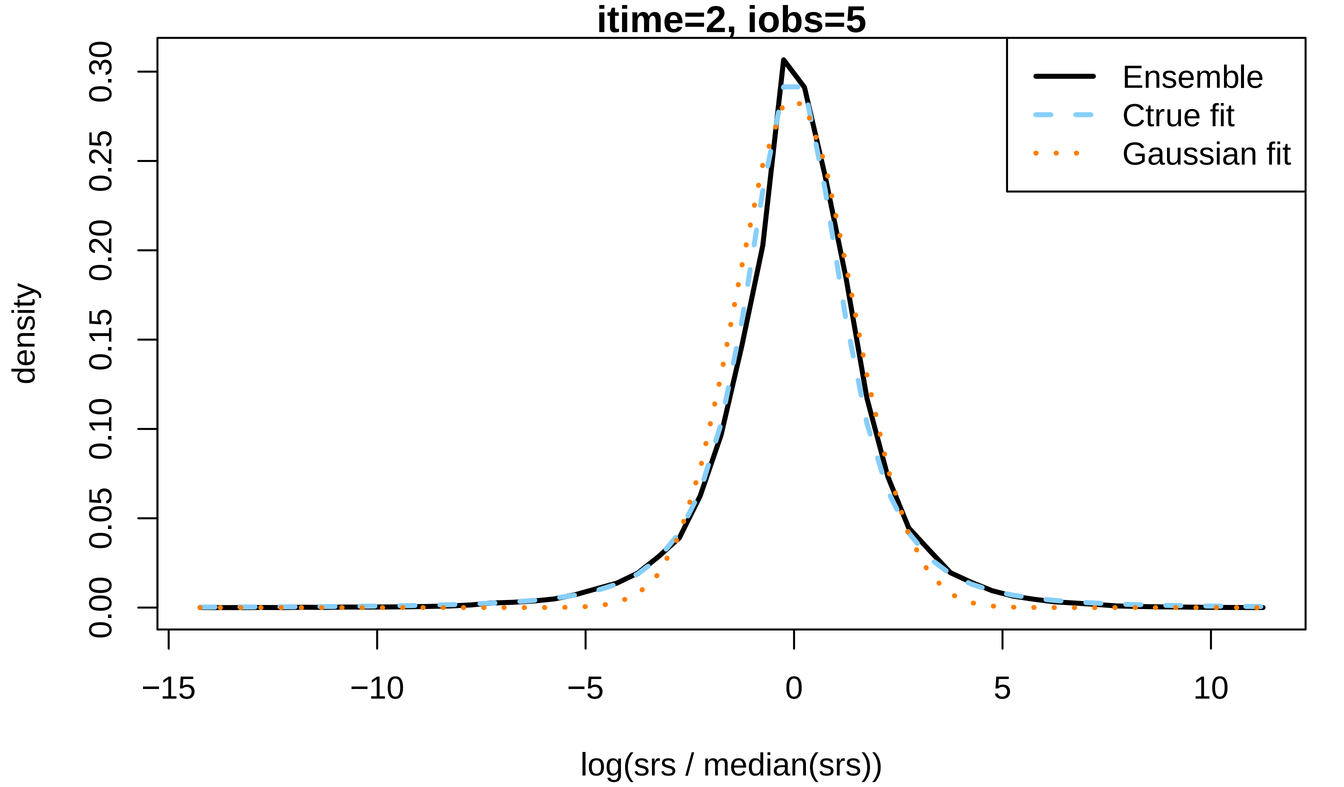

itime=2, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=2, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

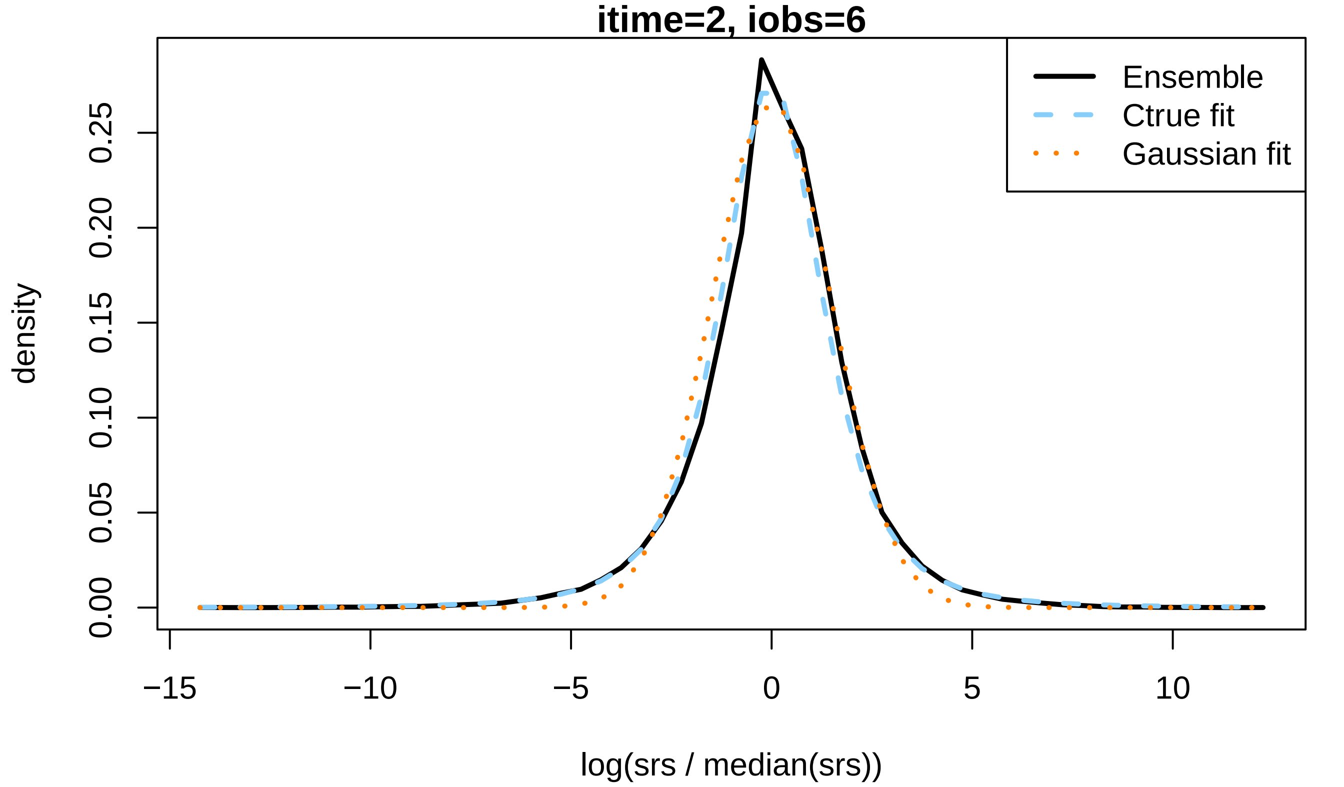

itime=2, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−20
−15
−10
−5
0
5
10
log(srs / median(srs))

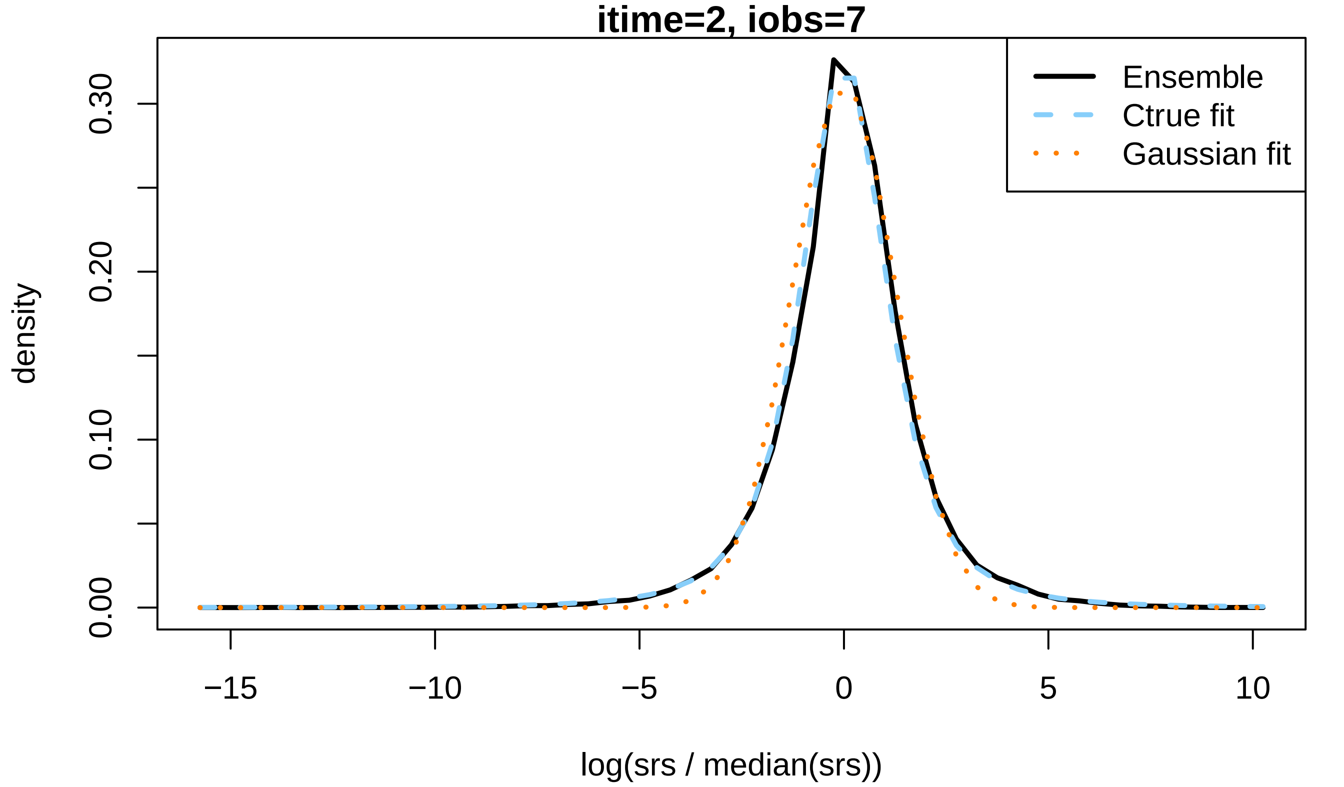


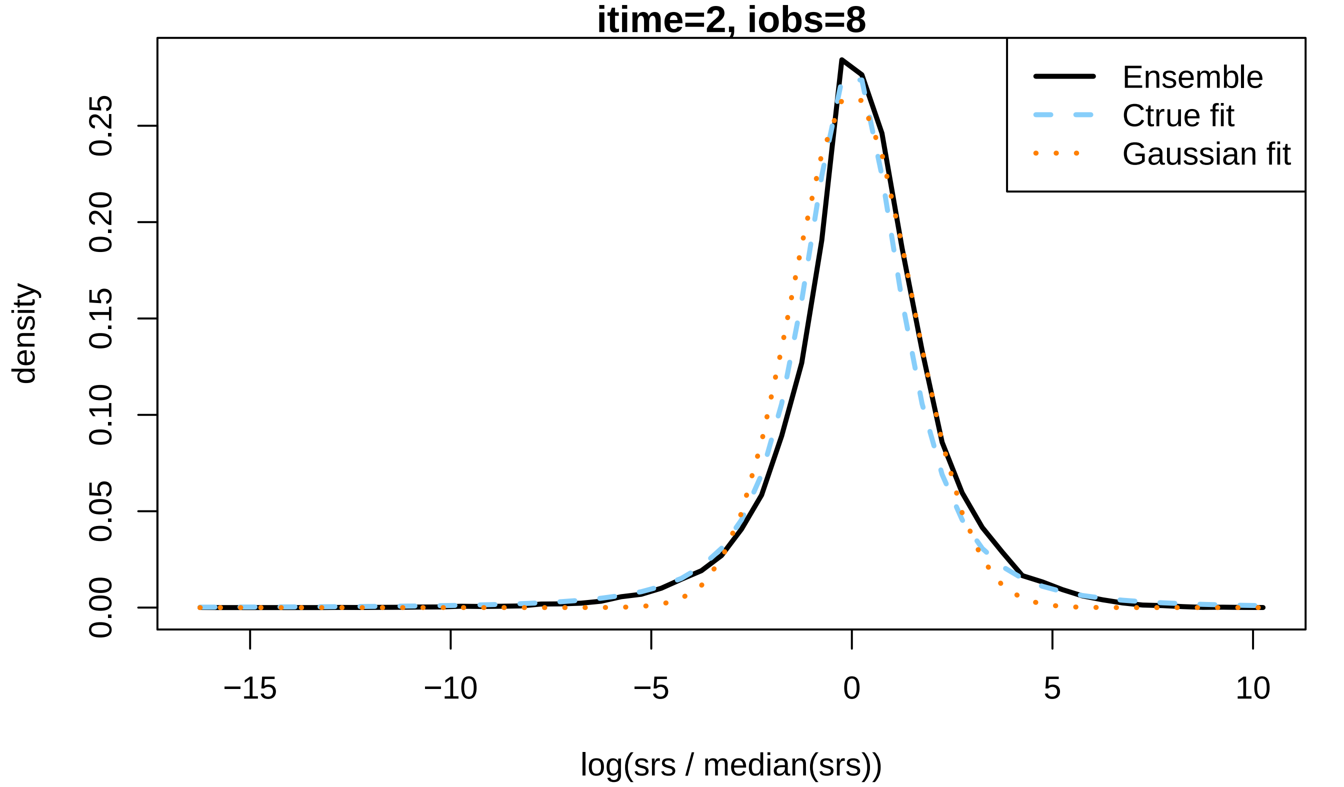
itime=2, iobs=4
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

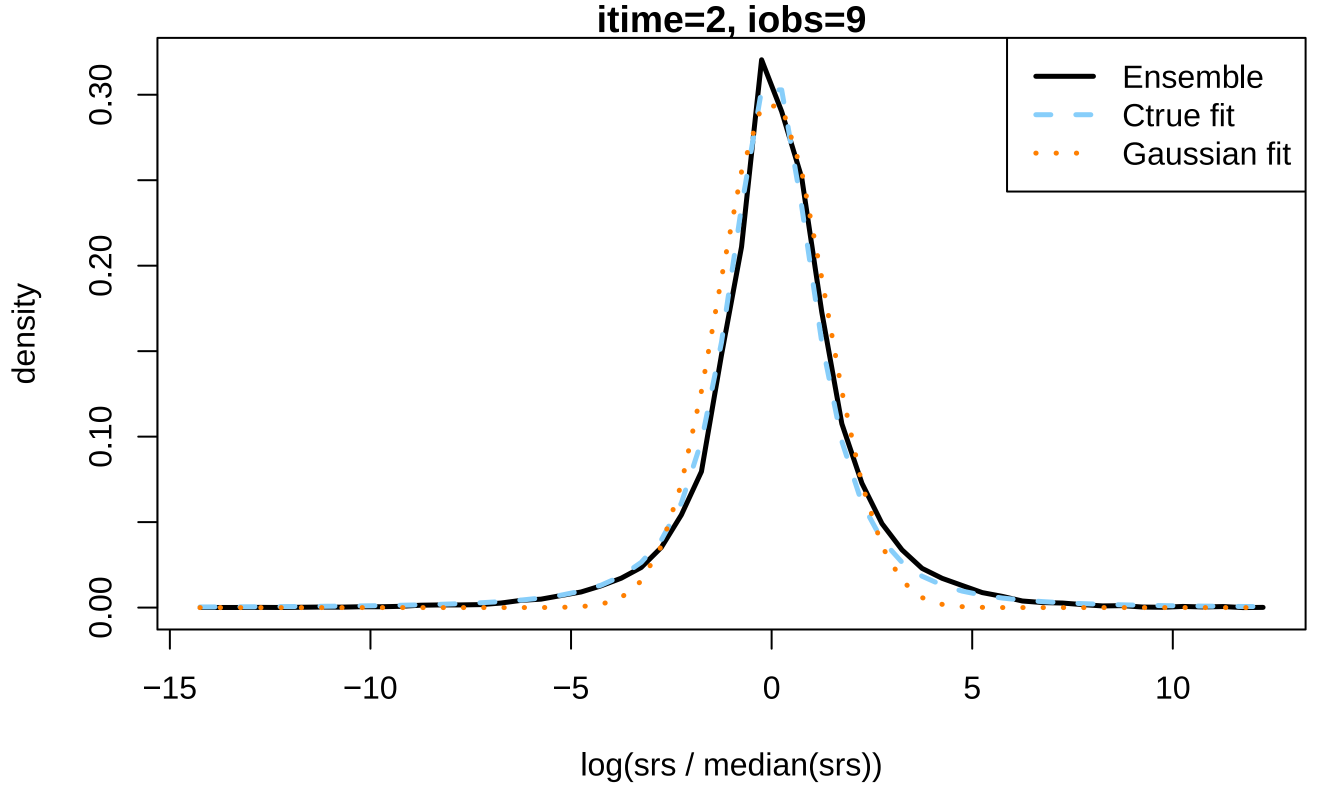


itime=2, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

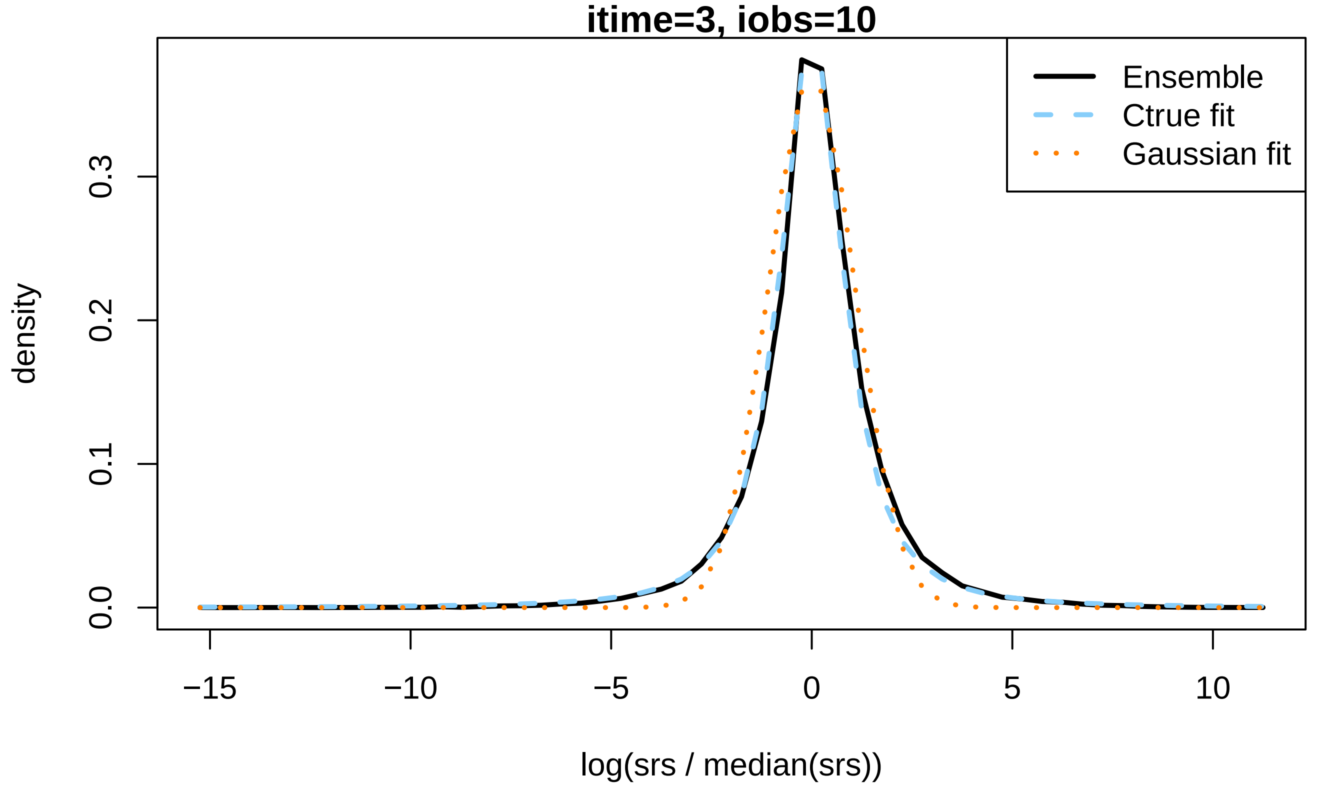

itime=2, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=2, iobs=7
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

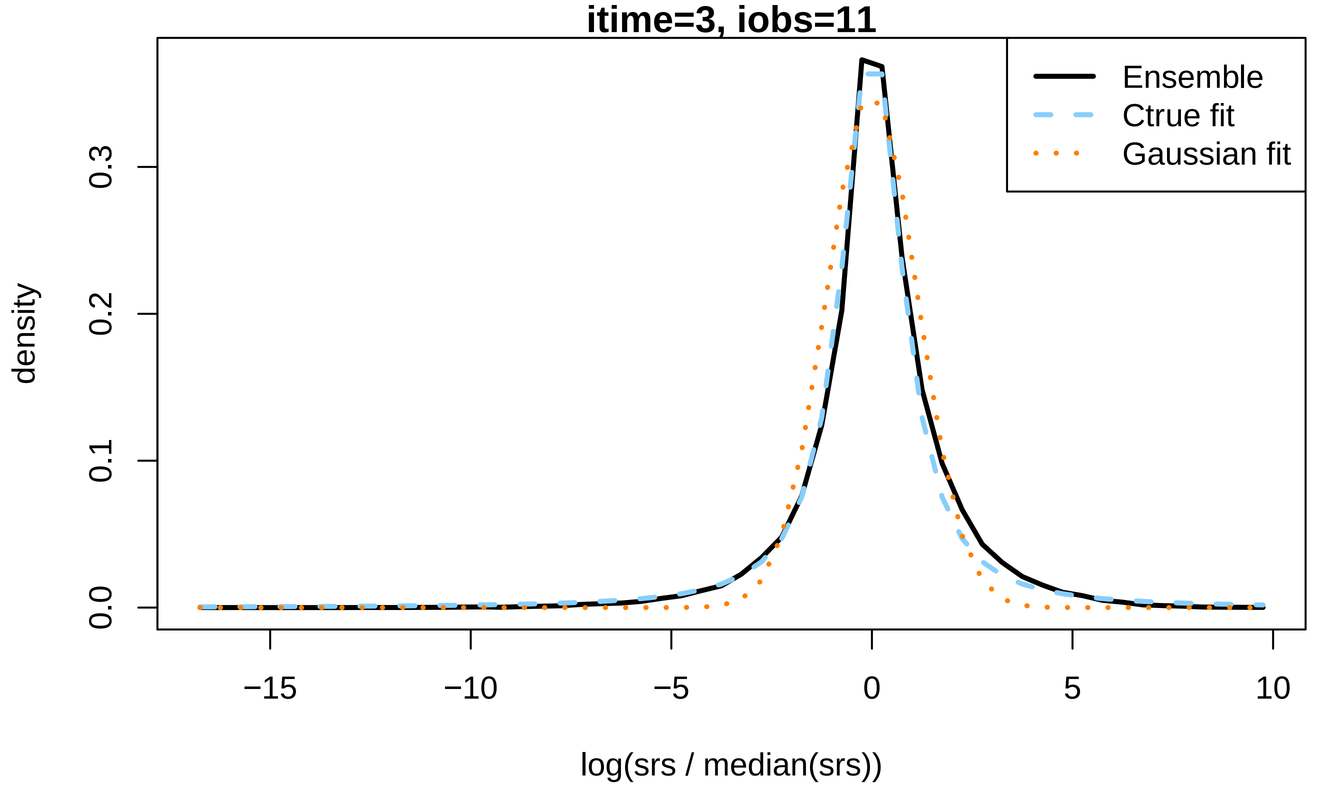

itime=2, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

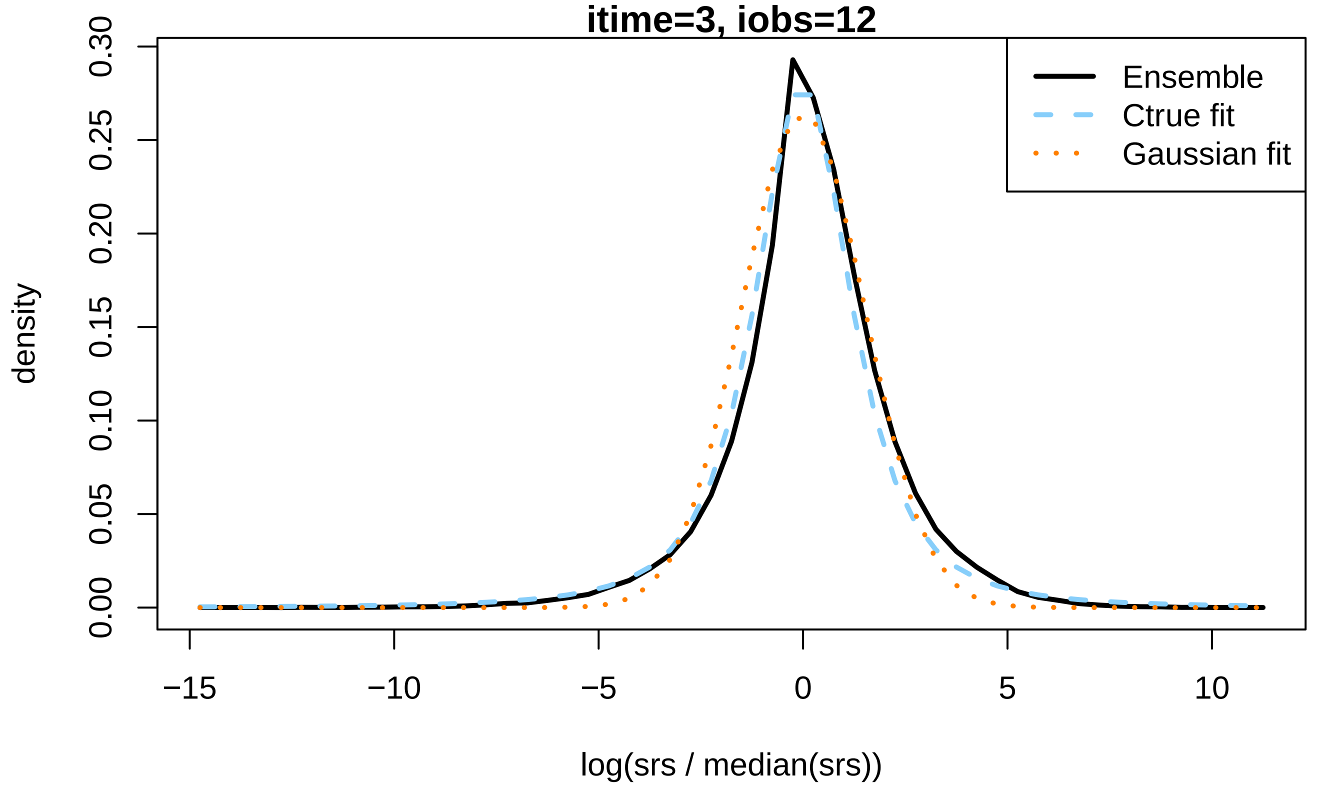


itime=2, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

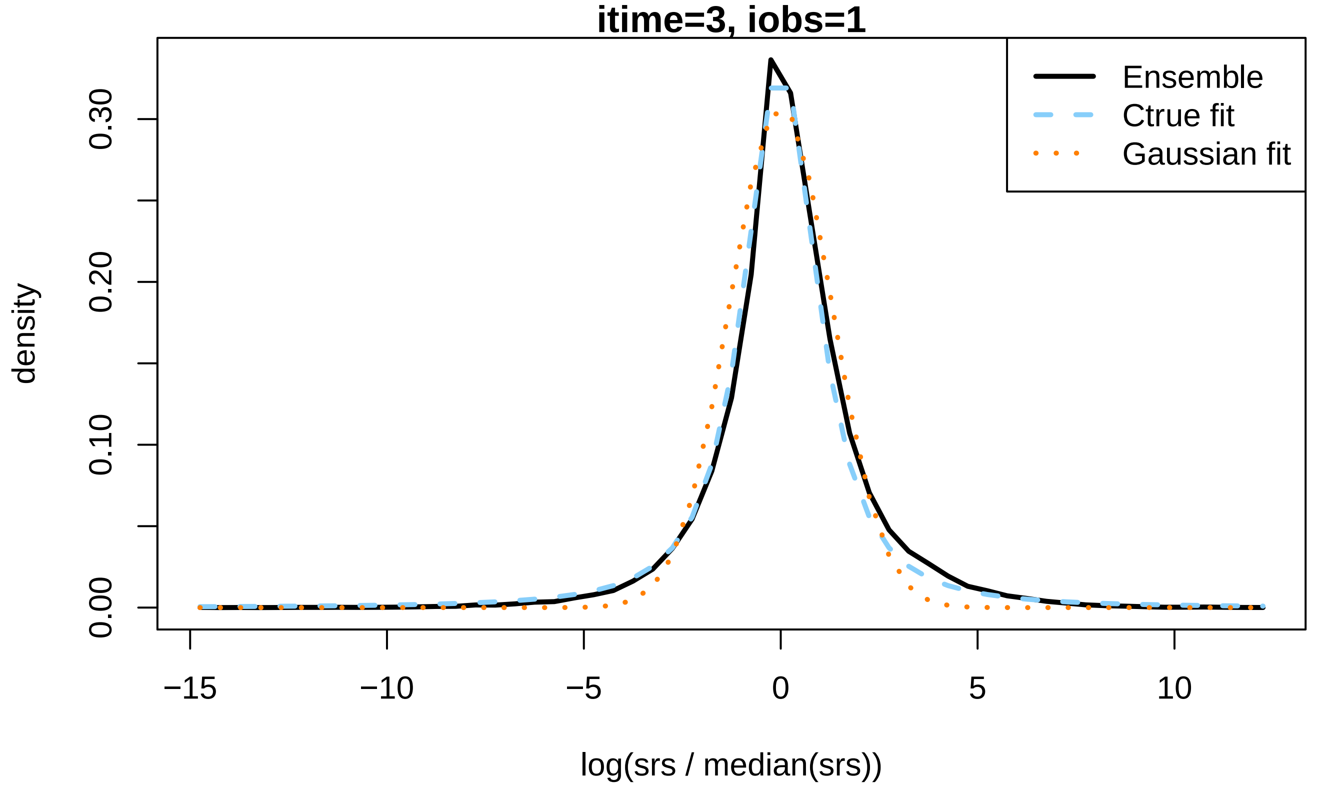


itime=3, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))

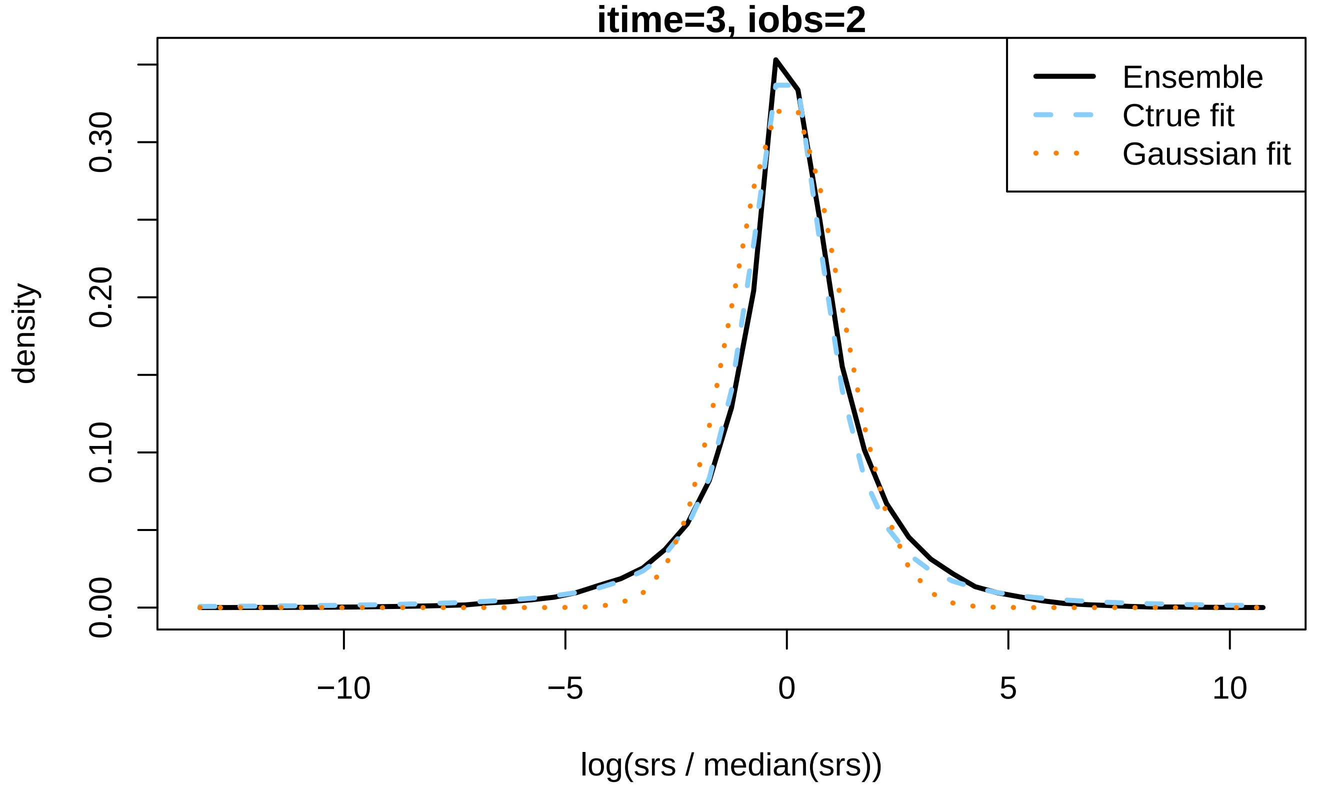

itime=3, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
−15
−10
−5
0
5
10
log(srs / median(srs))



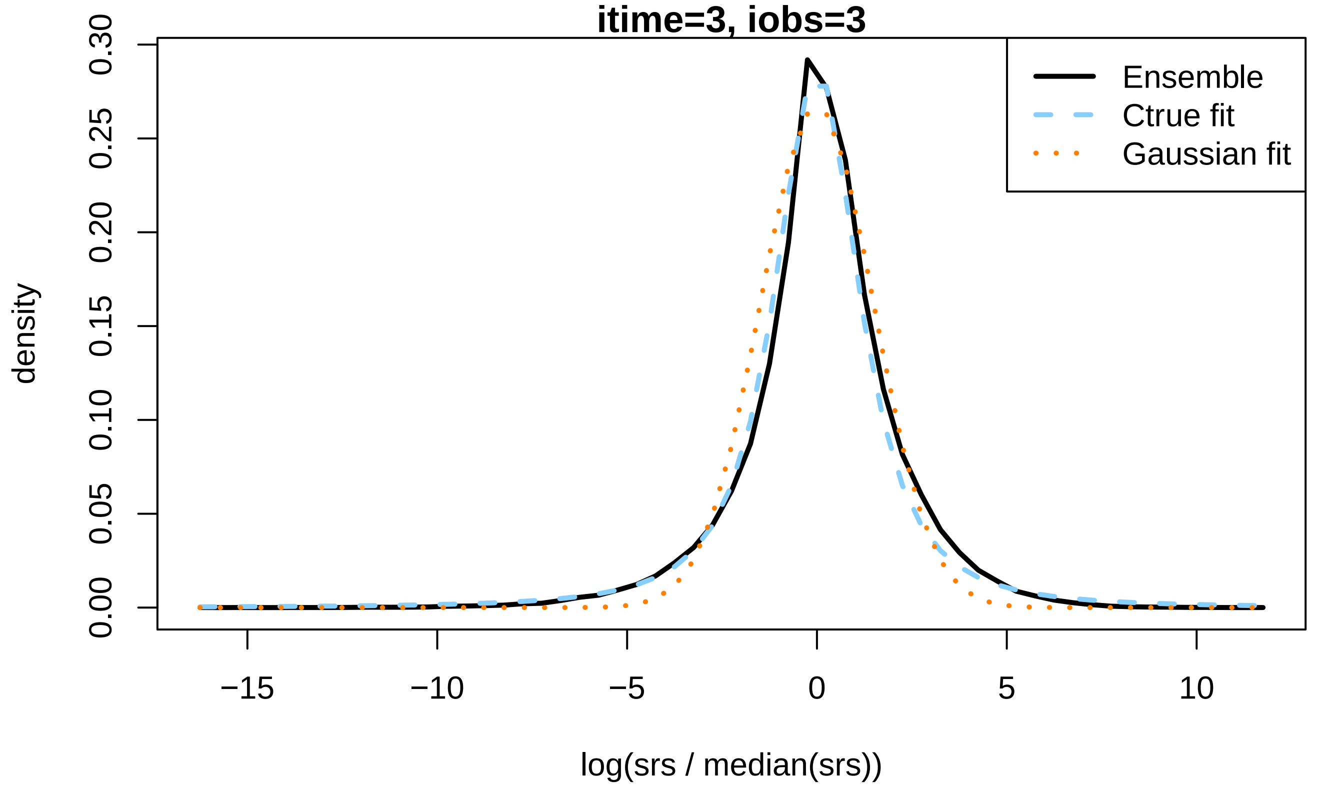
itime=3, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



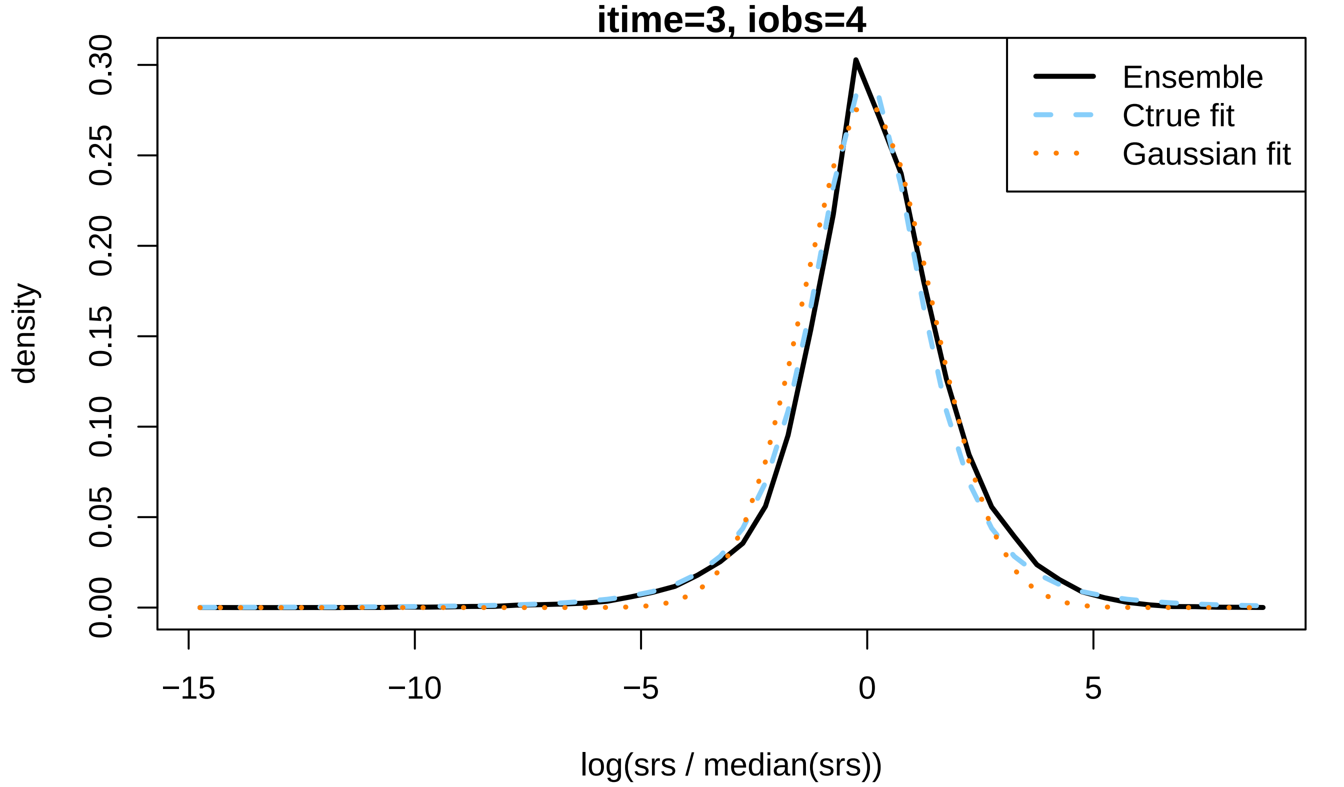
itime=3, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

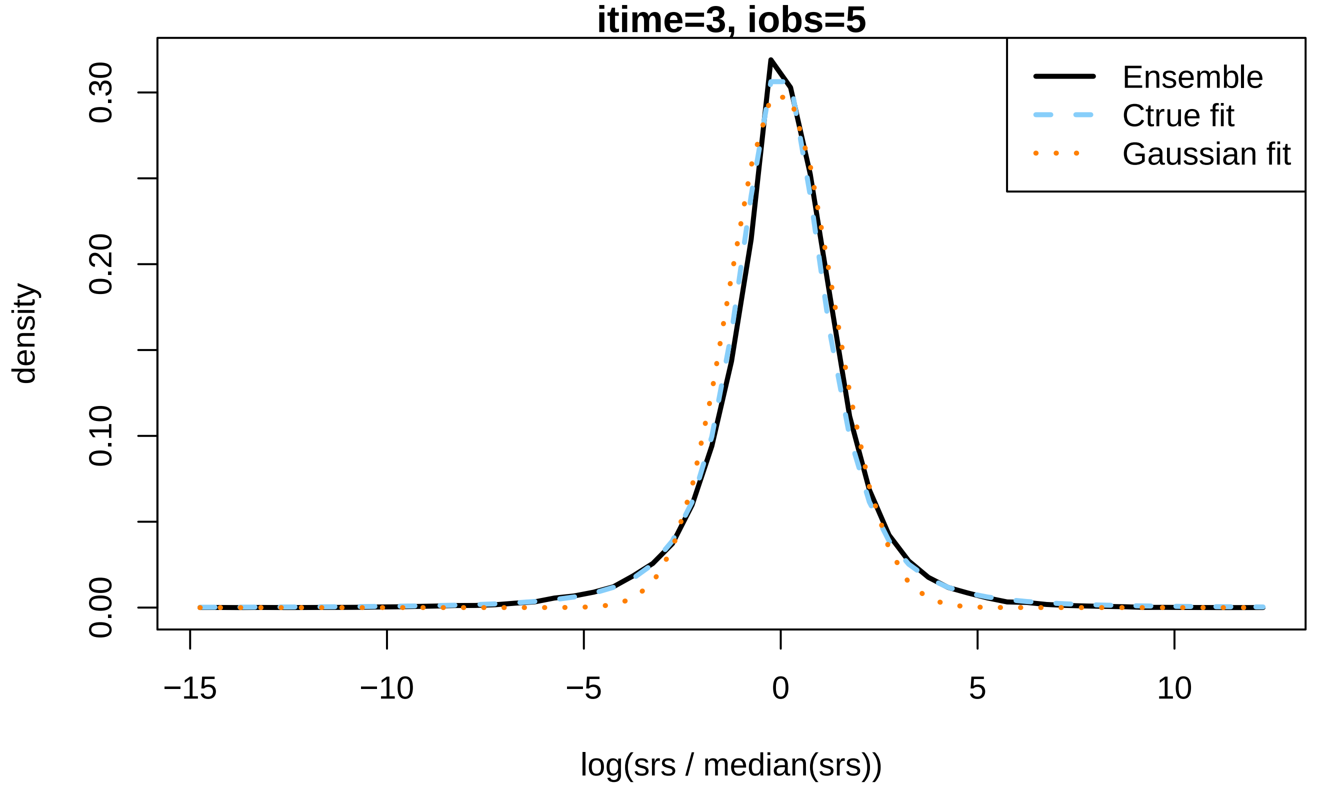


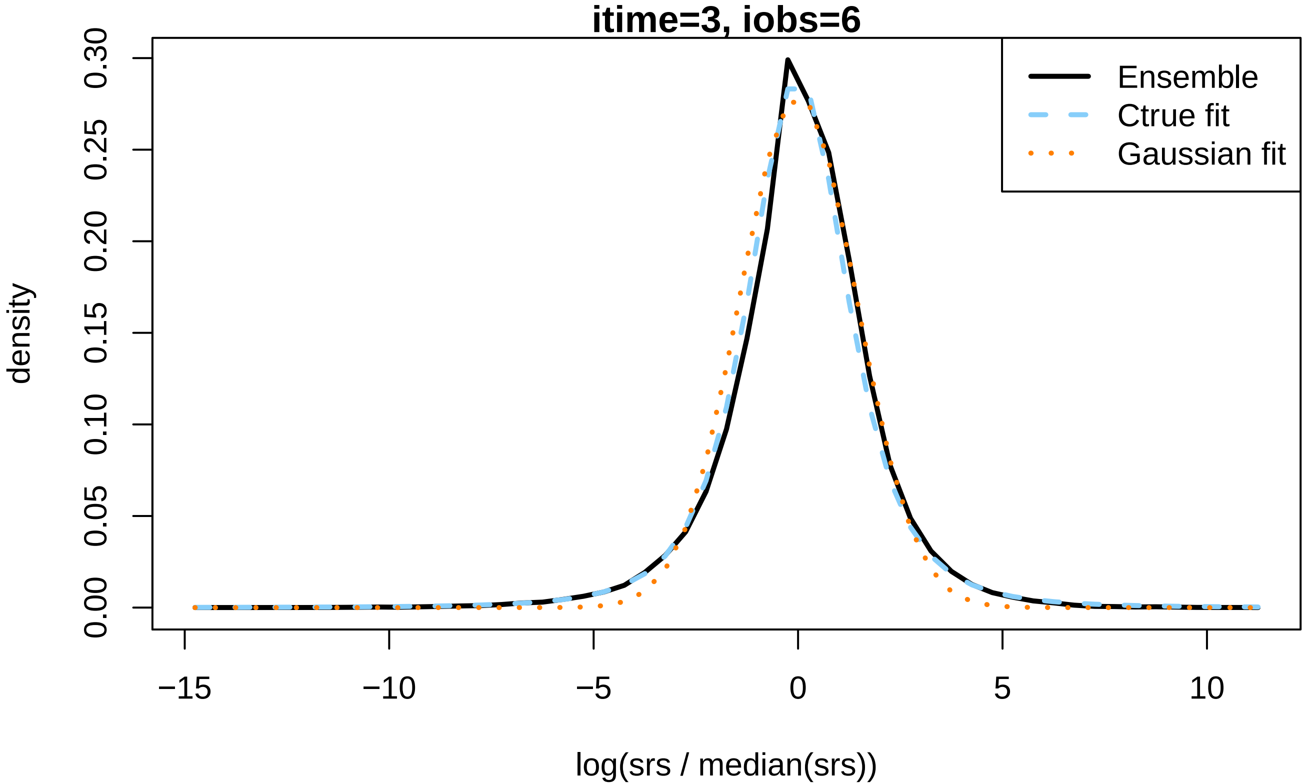
itime=3, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))

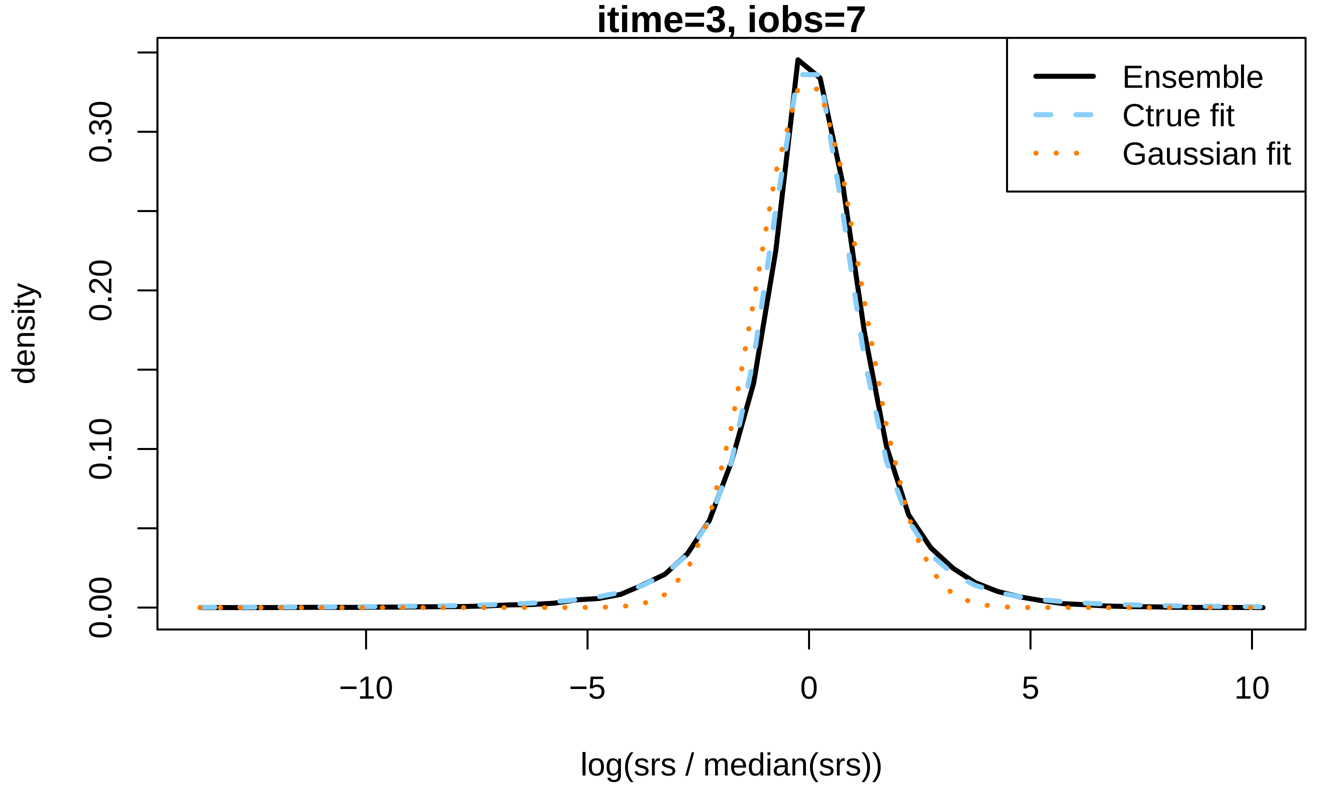


itime=3, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

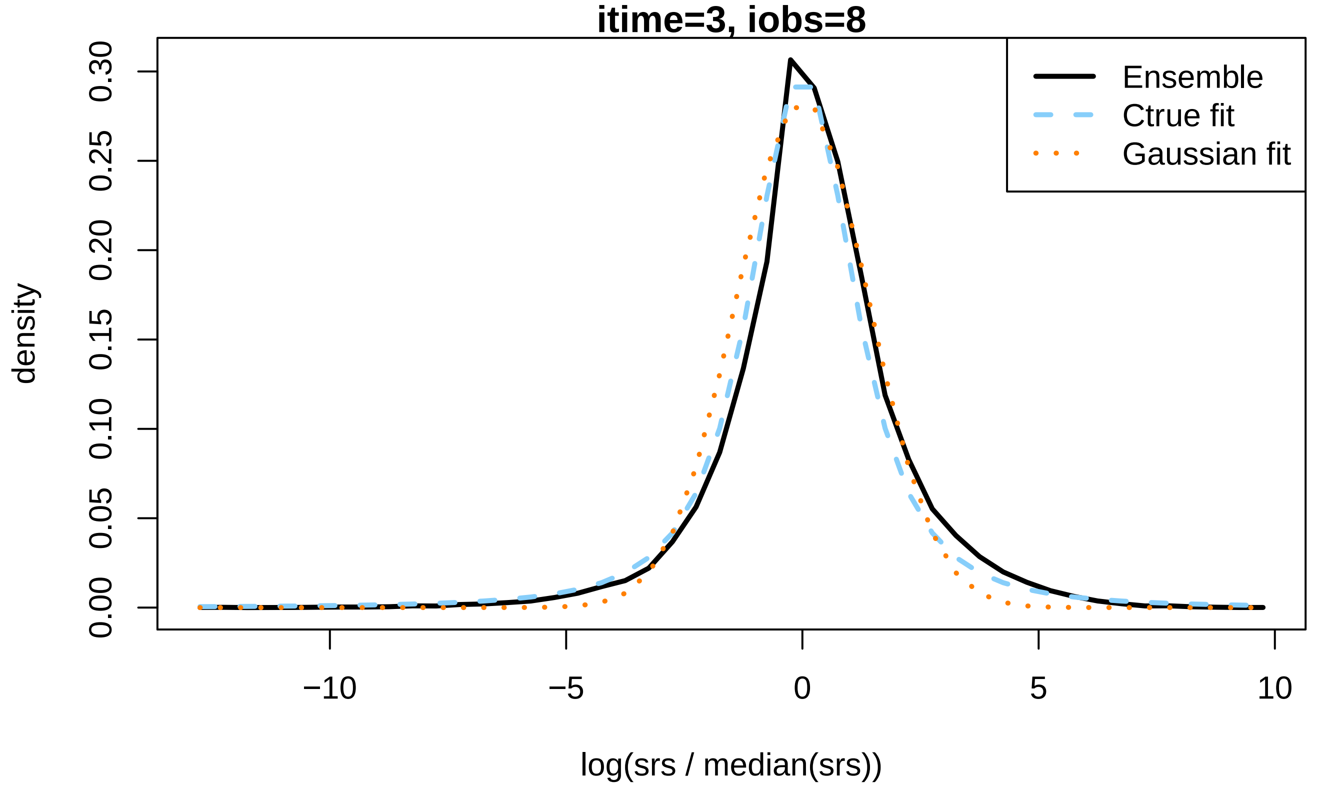

itime=3, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
log(srs / median(srs))


itime=3, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

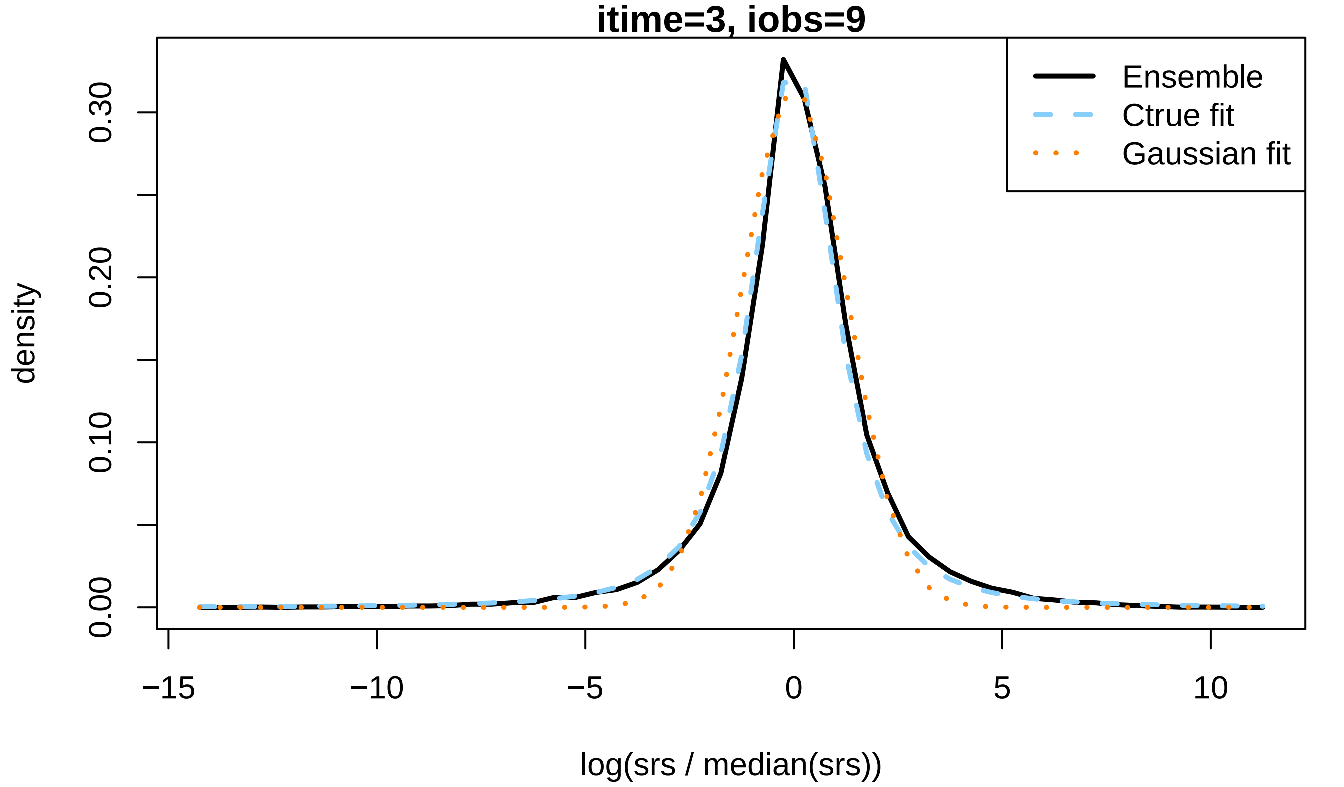

itime=3, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



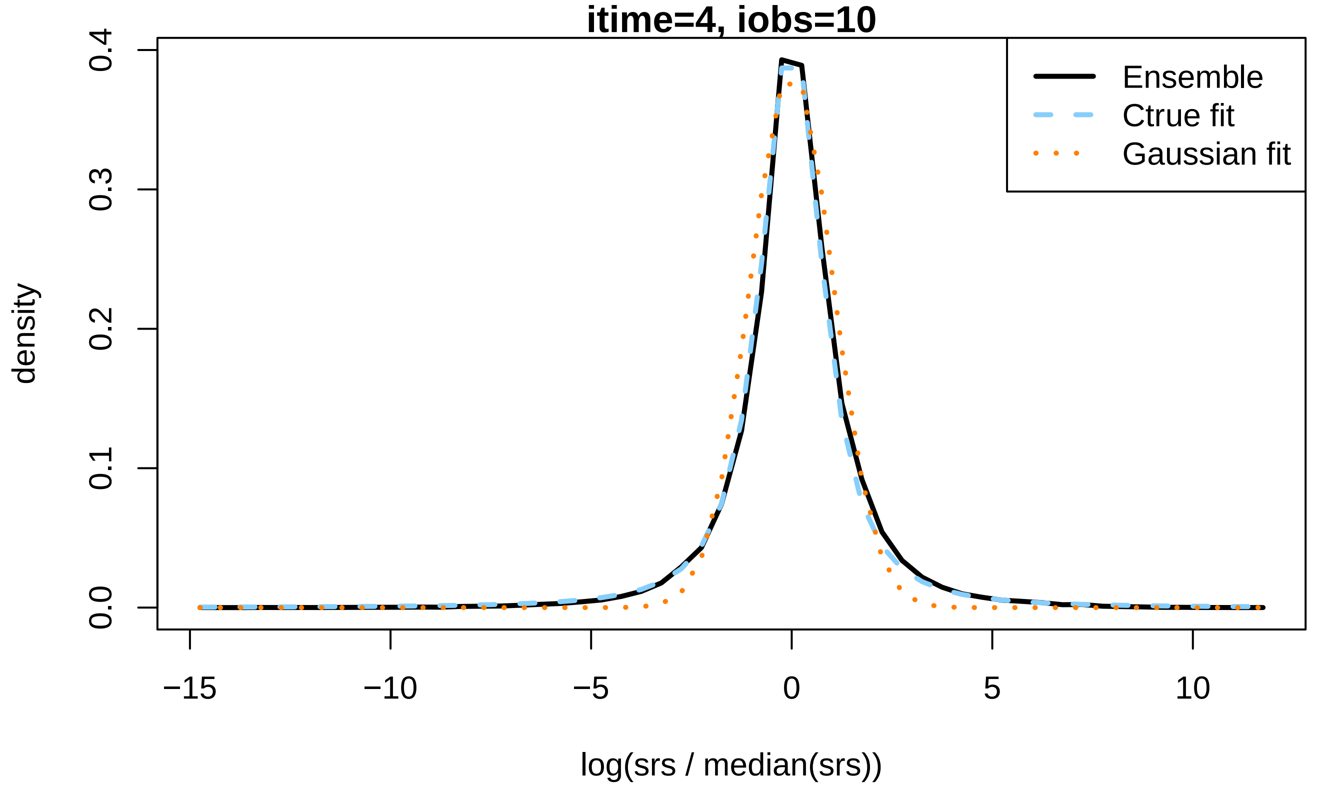
itime=3, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))



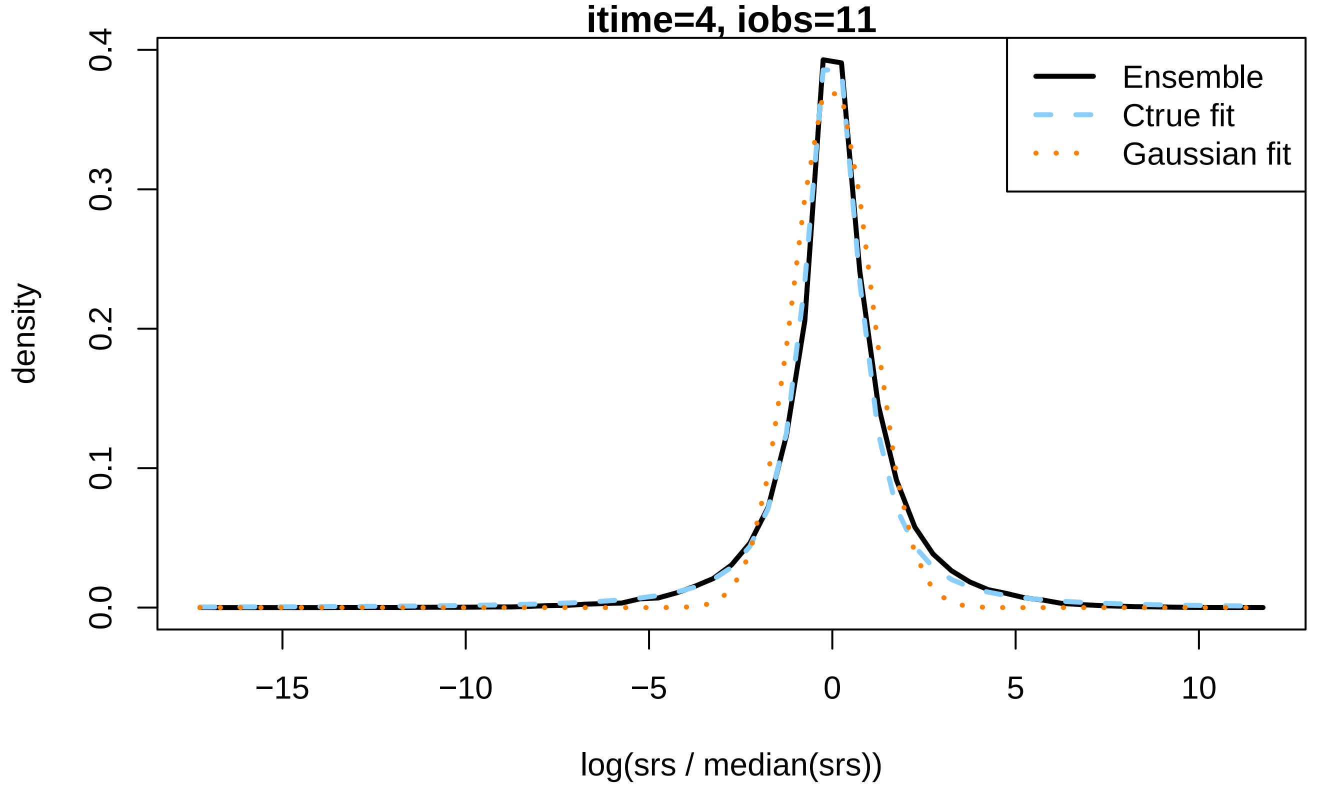
itime=3, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−10
−5
0
5
10
log(srs / median(srs))

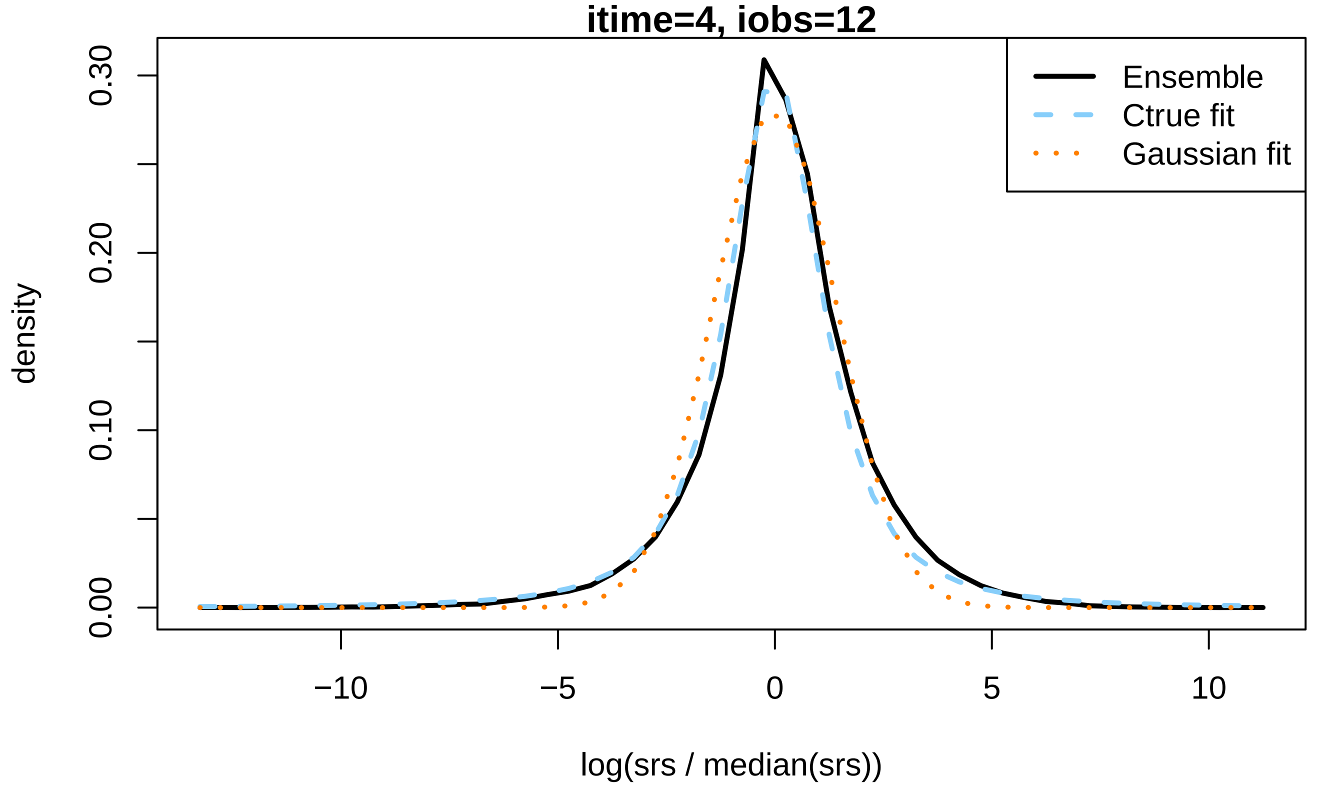


itime=3, iobs=9
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

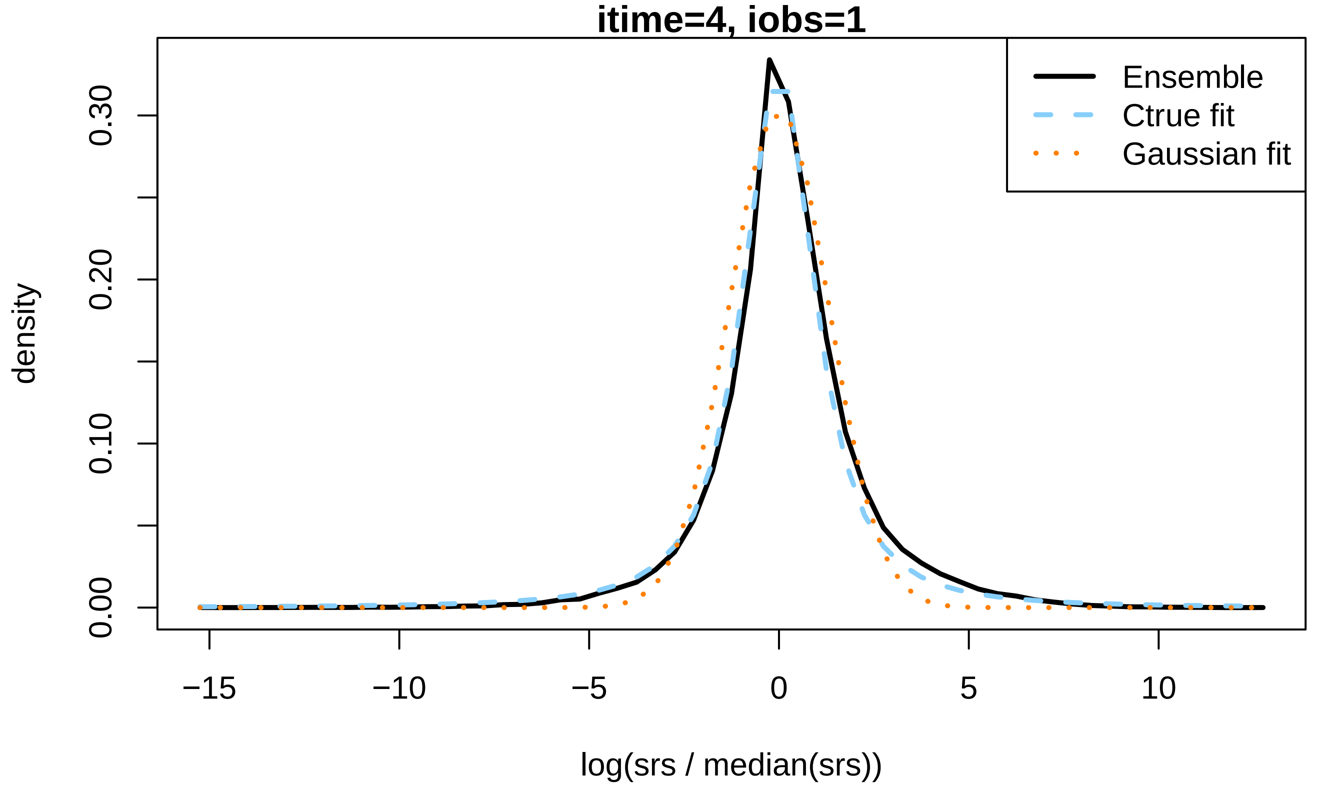


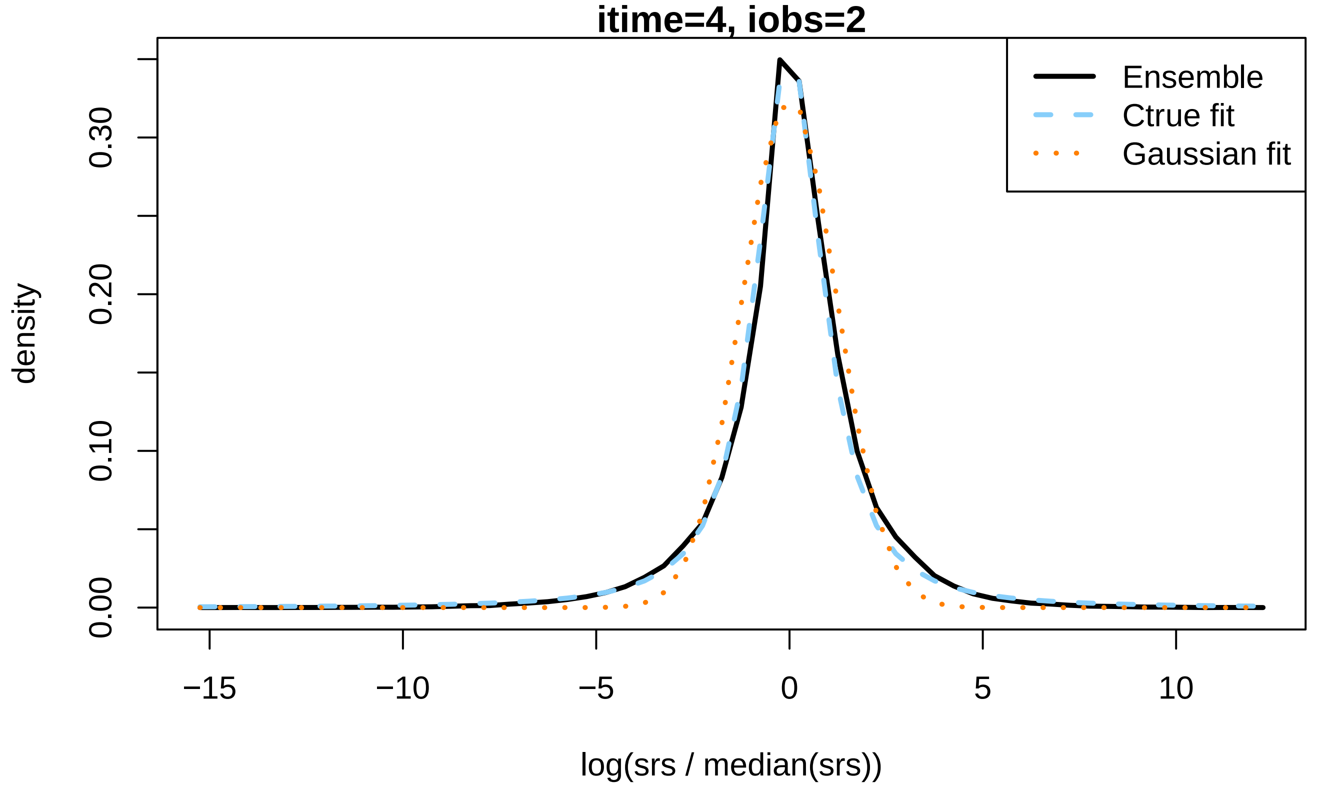
itime=4, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))

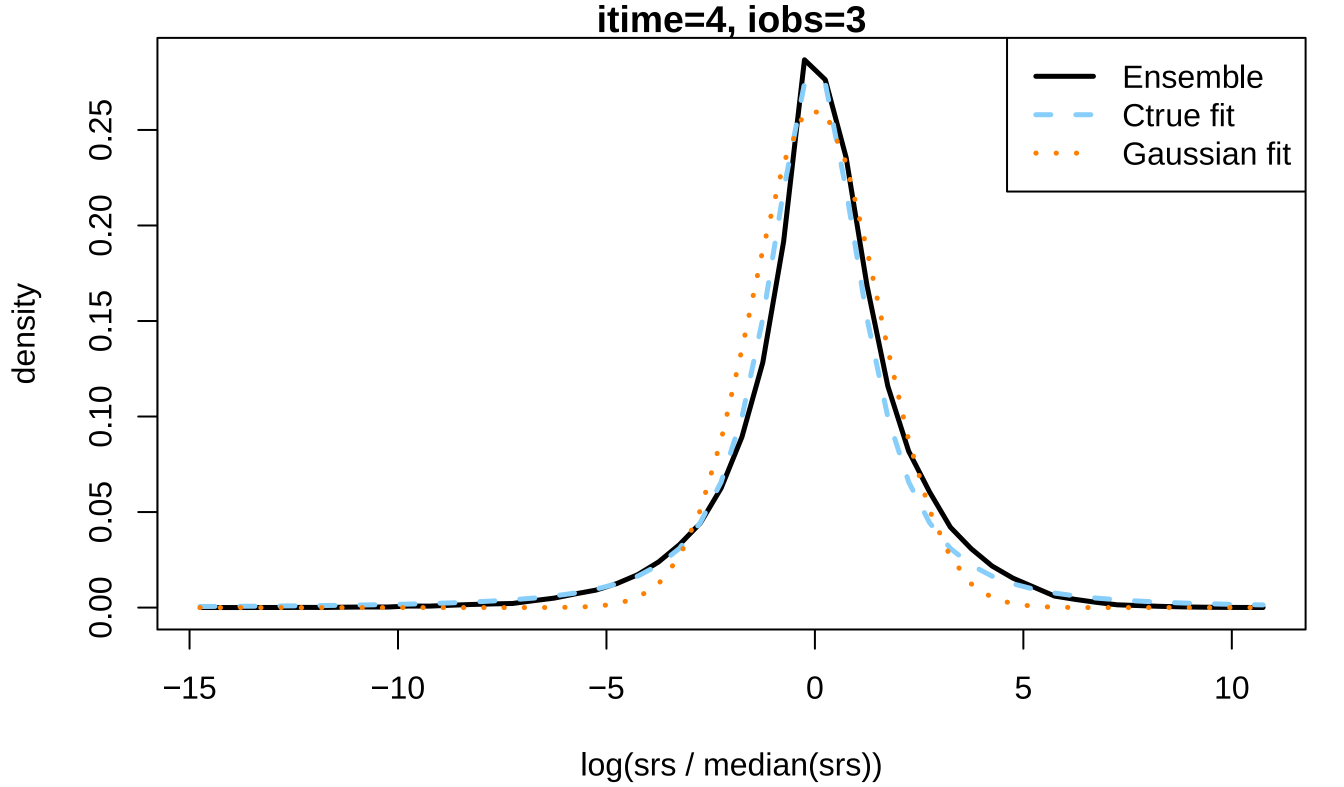

itime=4, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))



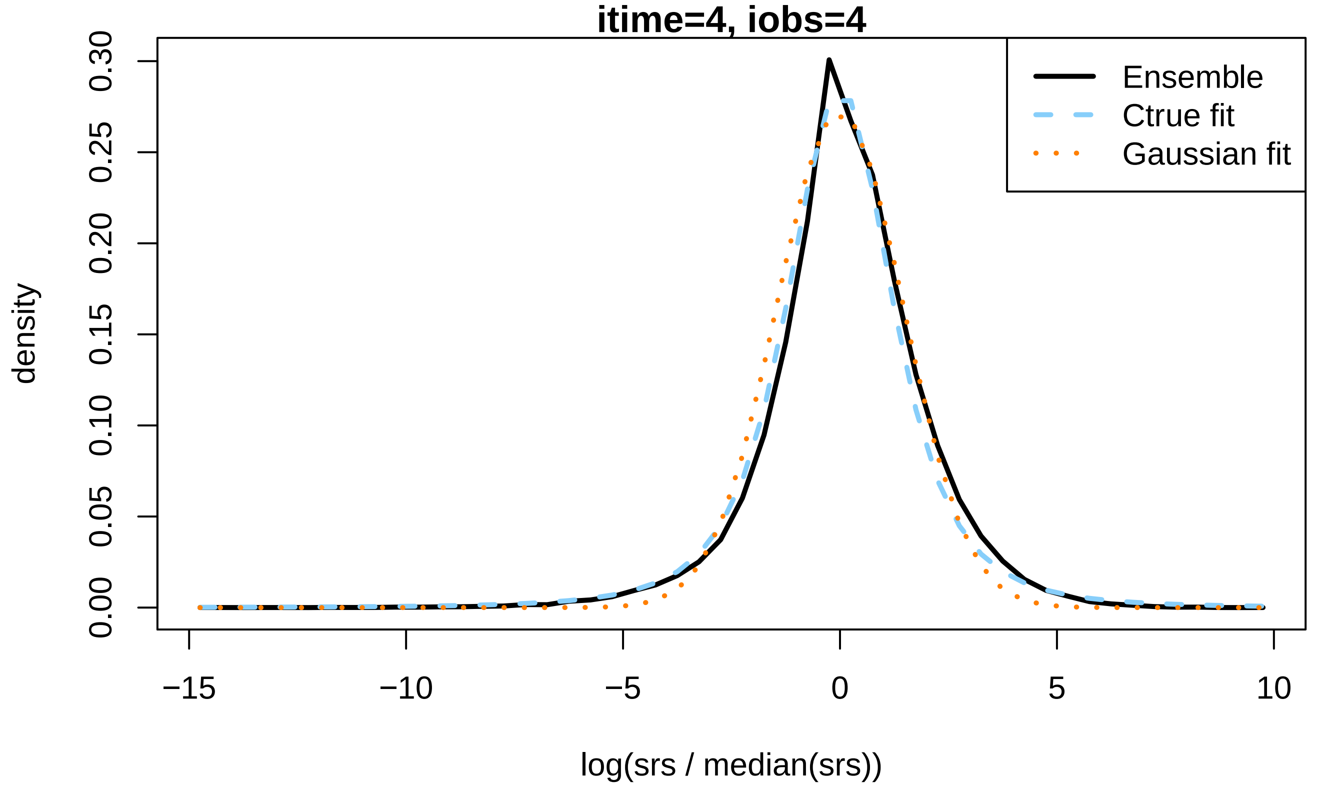
itime=4, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))

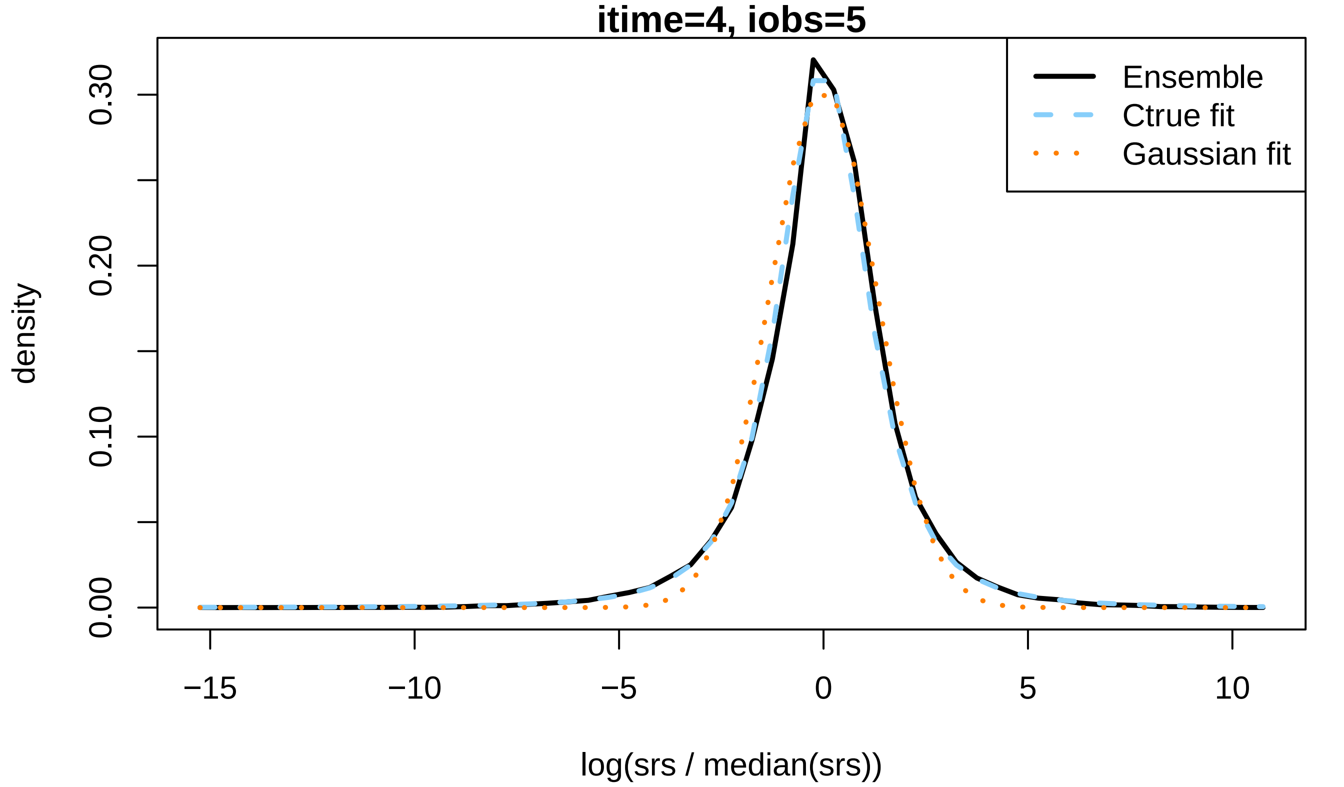

itime=4, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

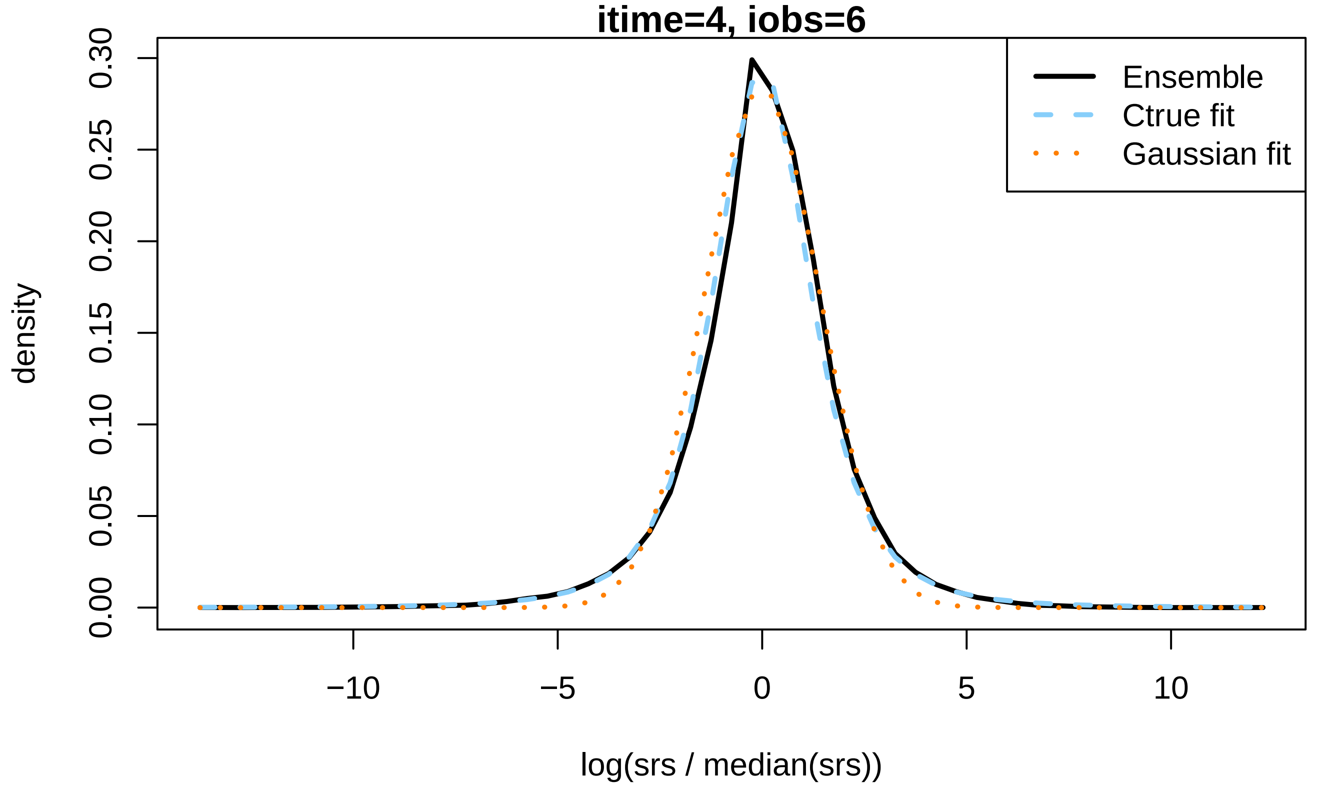

itime=4, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



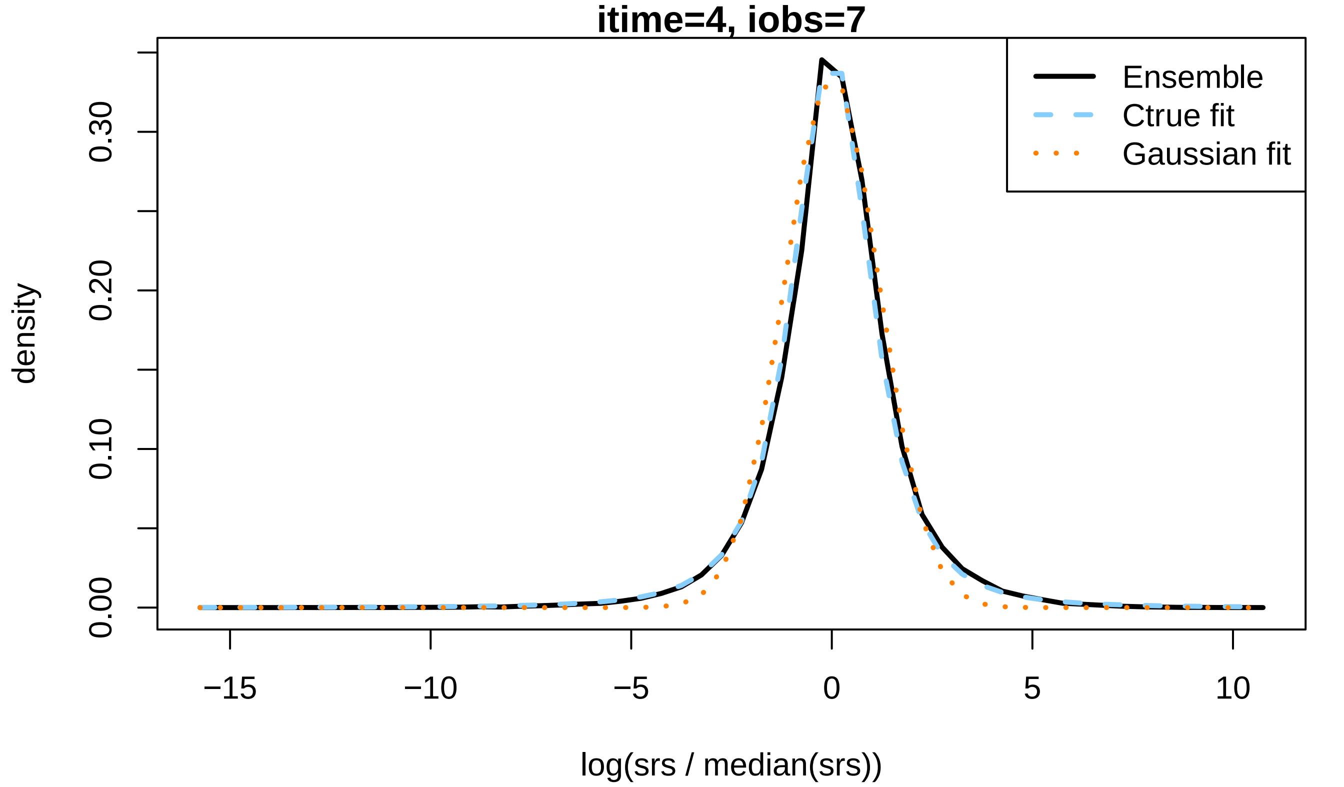
itime=4, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=4, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

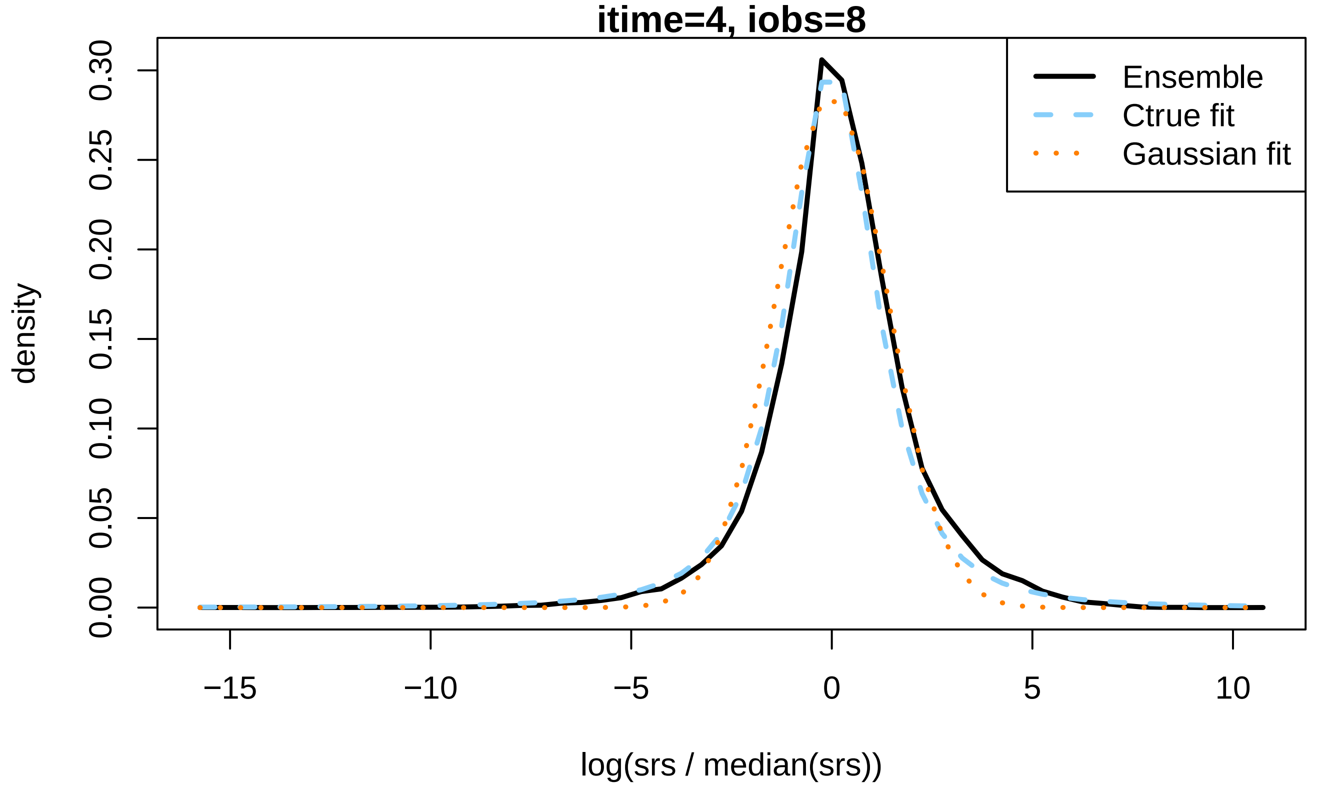

itime=4, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

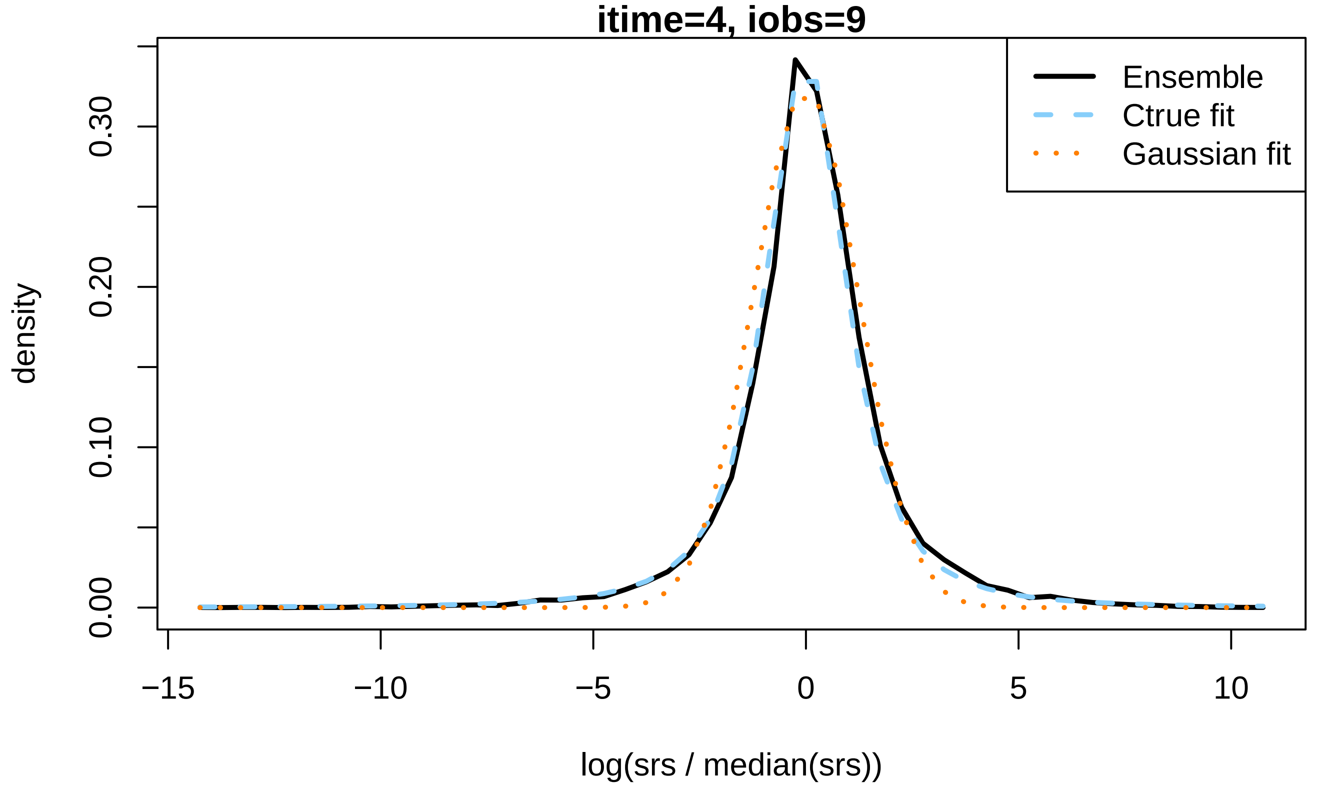


itime=4, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−10
−5
0
5
10
log(srs / median(srs))

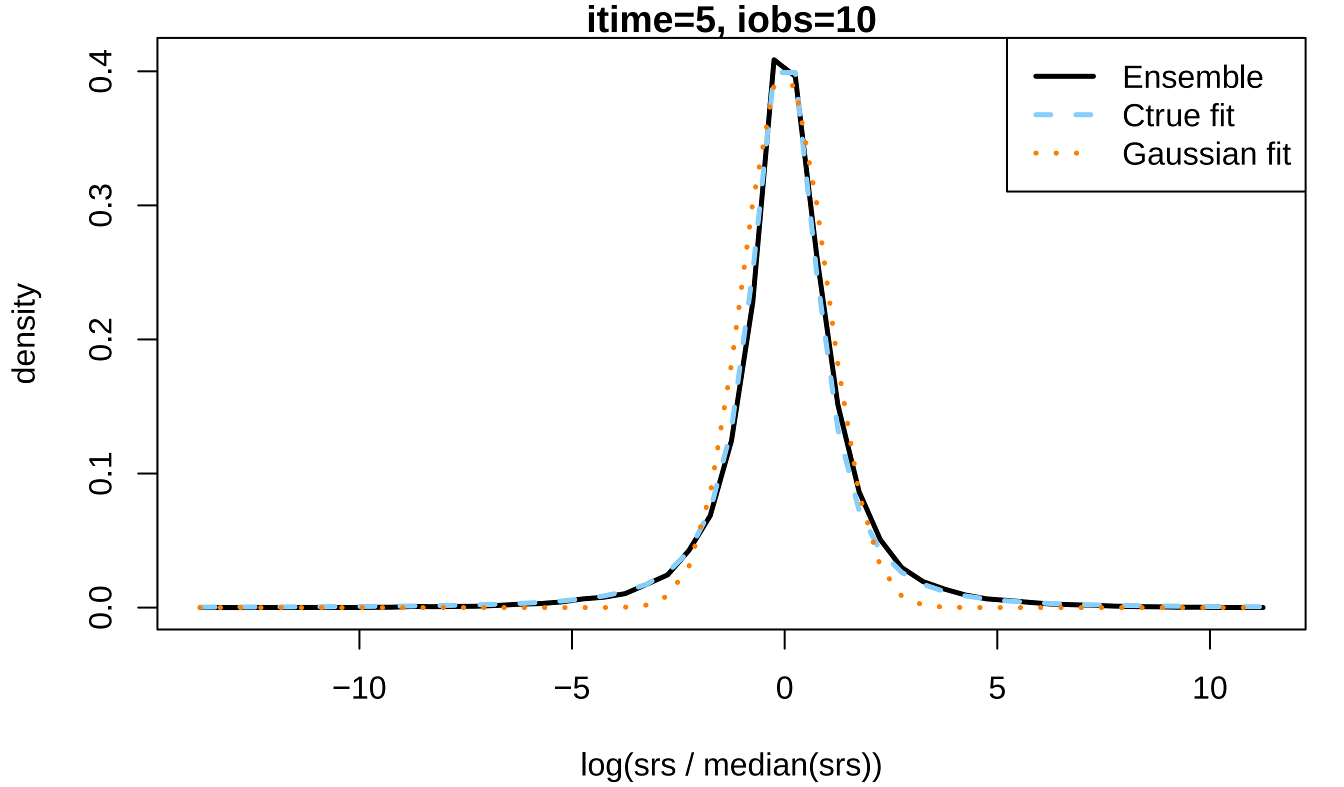


itime=4, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

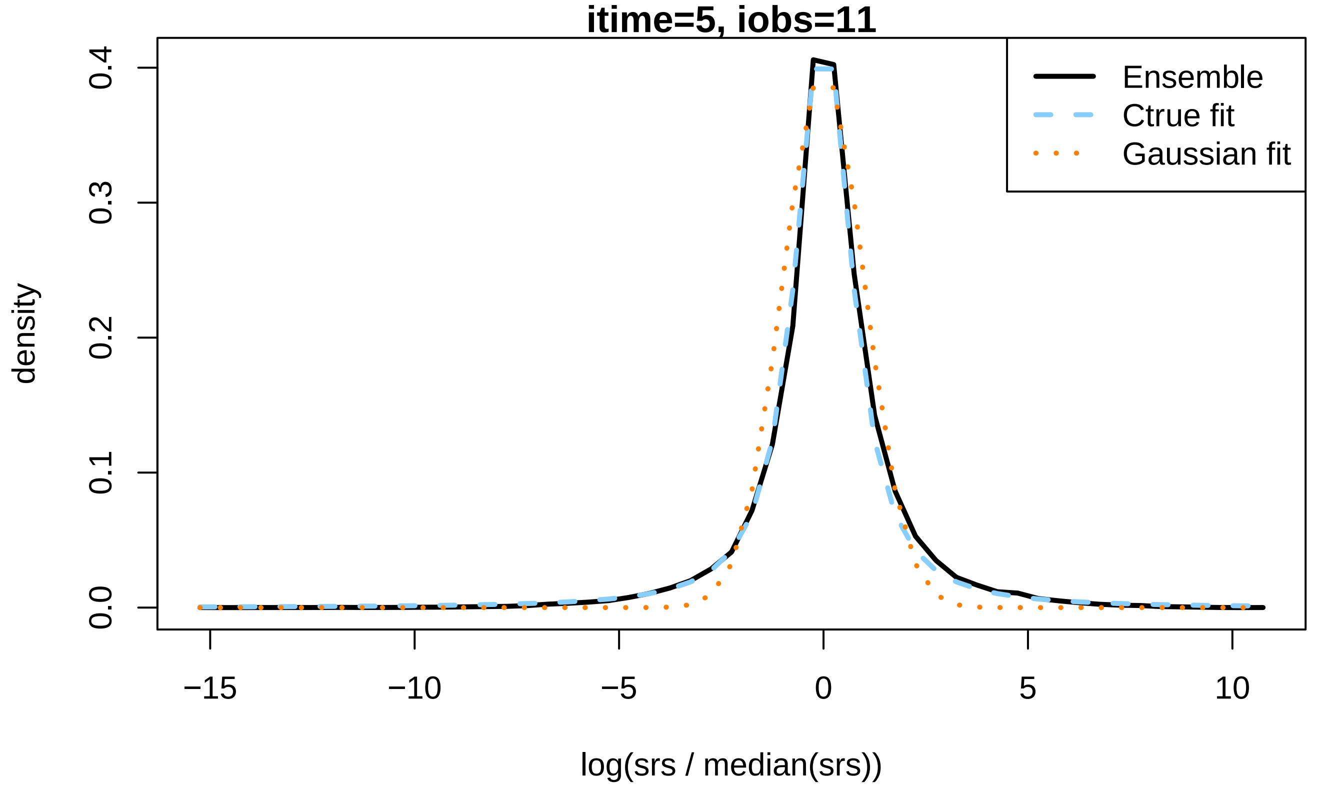

itime=4, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

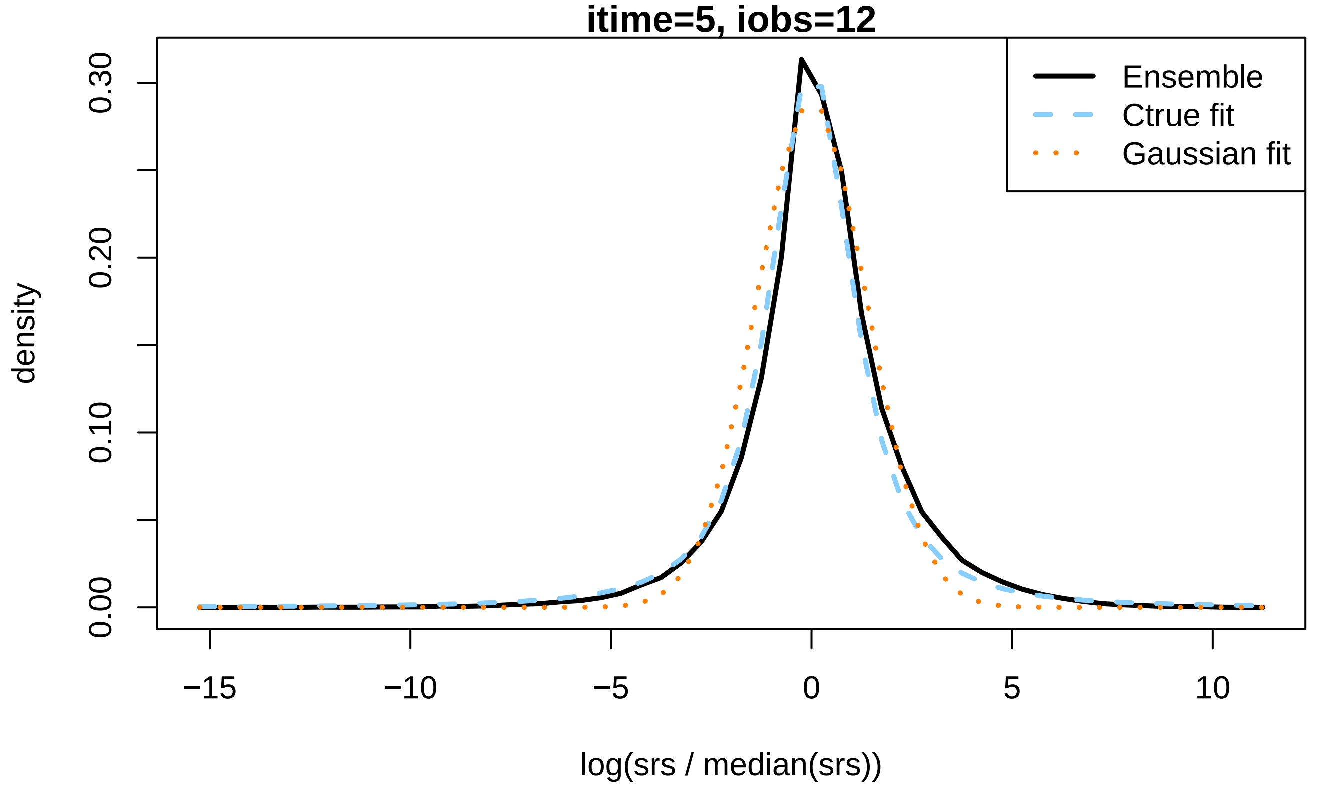


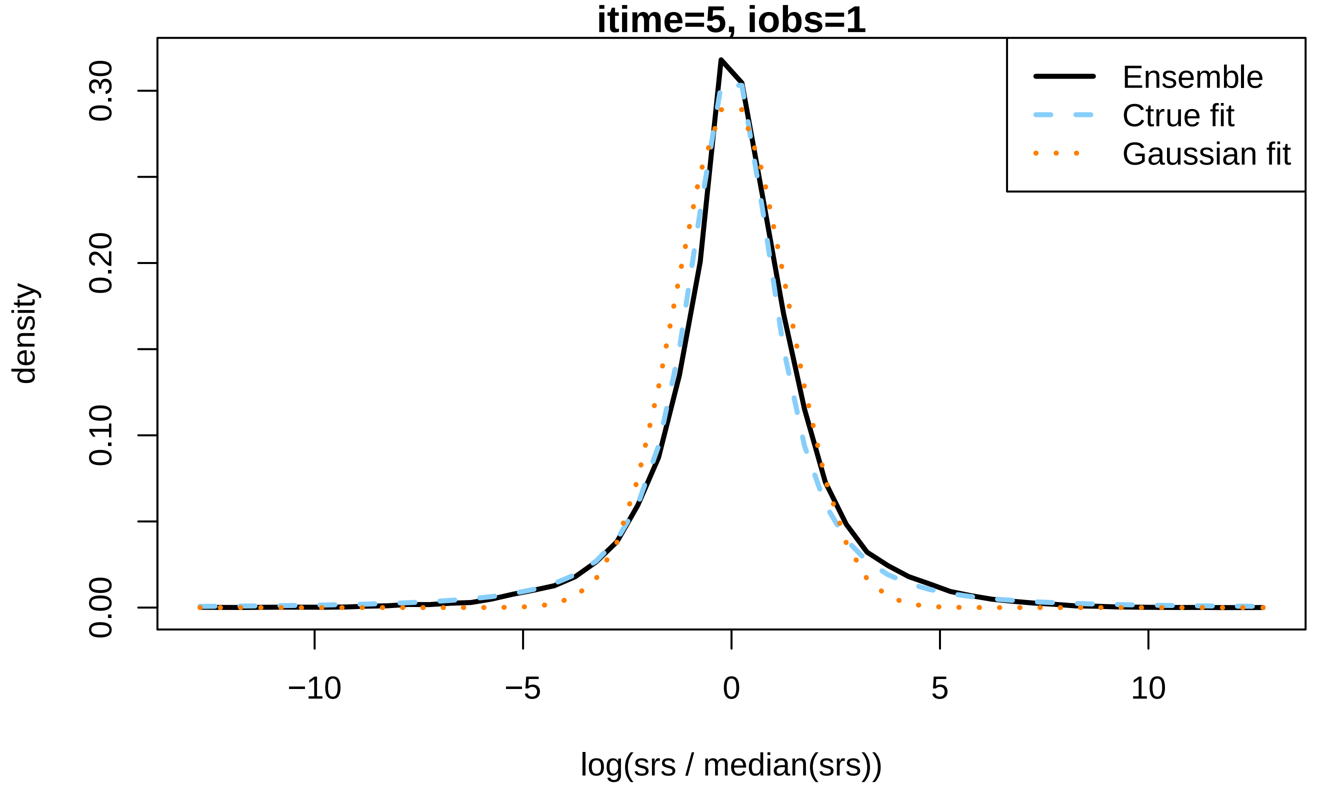
itime=4, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



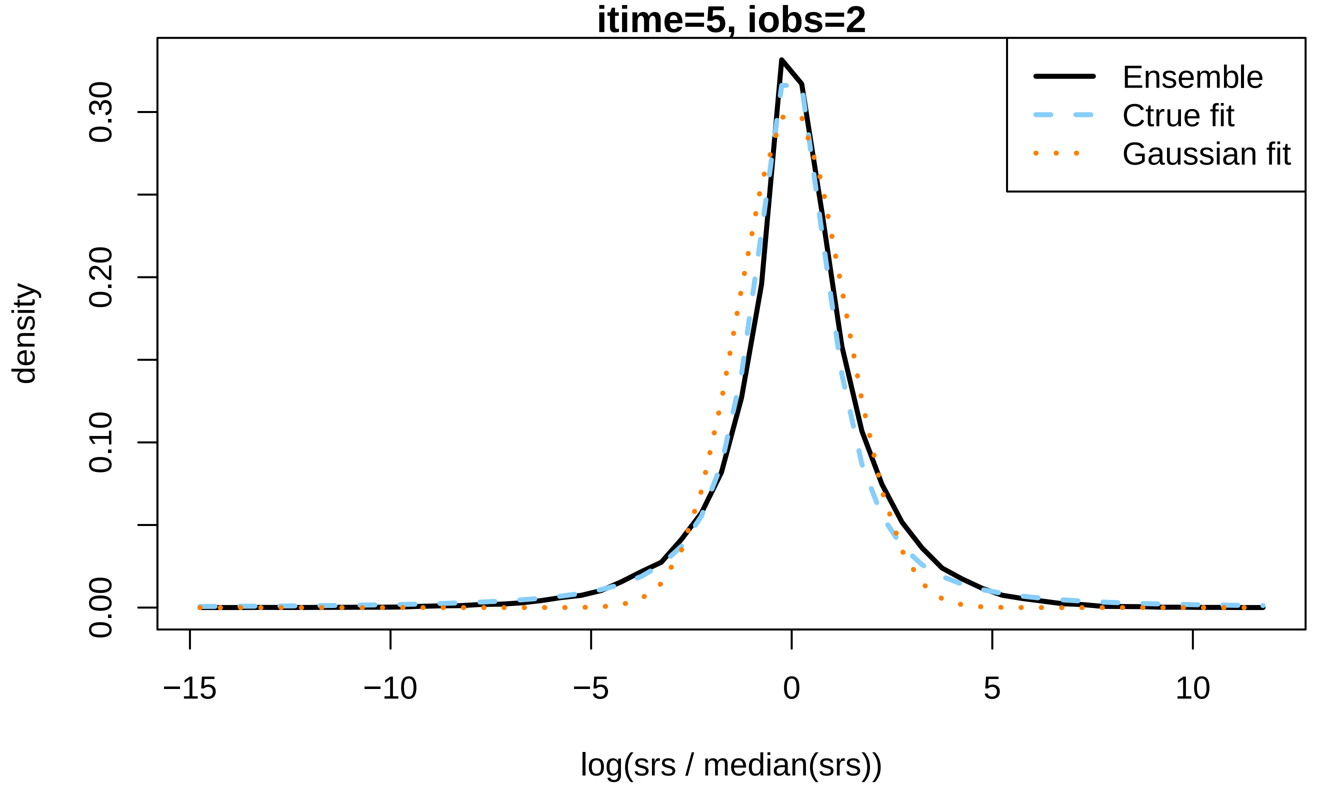
itime=5, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−10
−5
0
5
10
log(srs / median(srs))


itime=5, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))

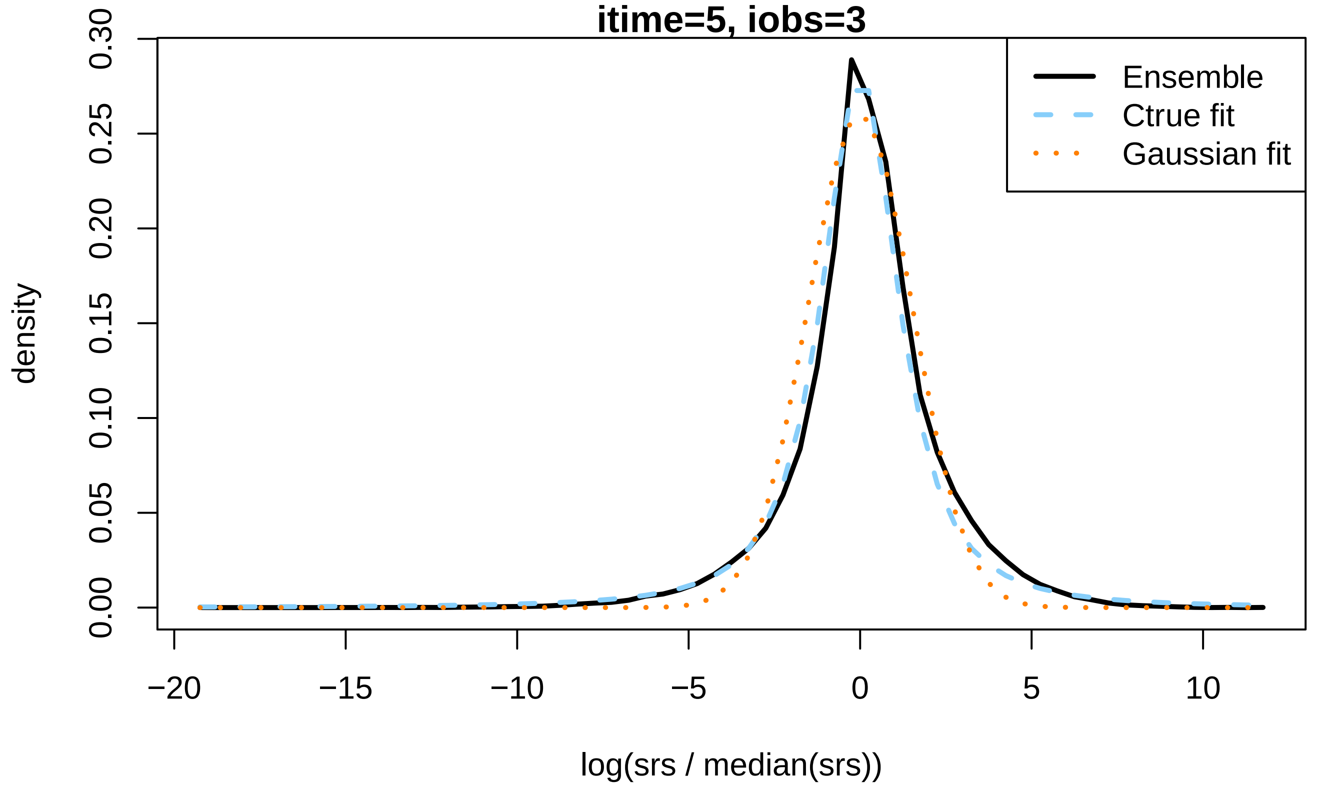

itime=5, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

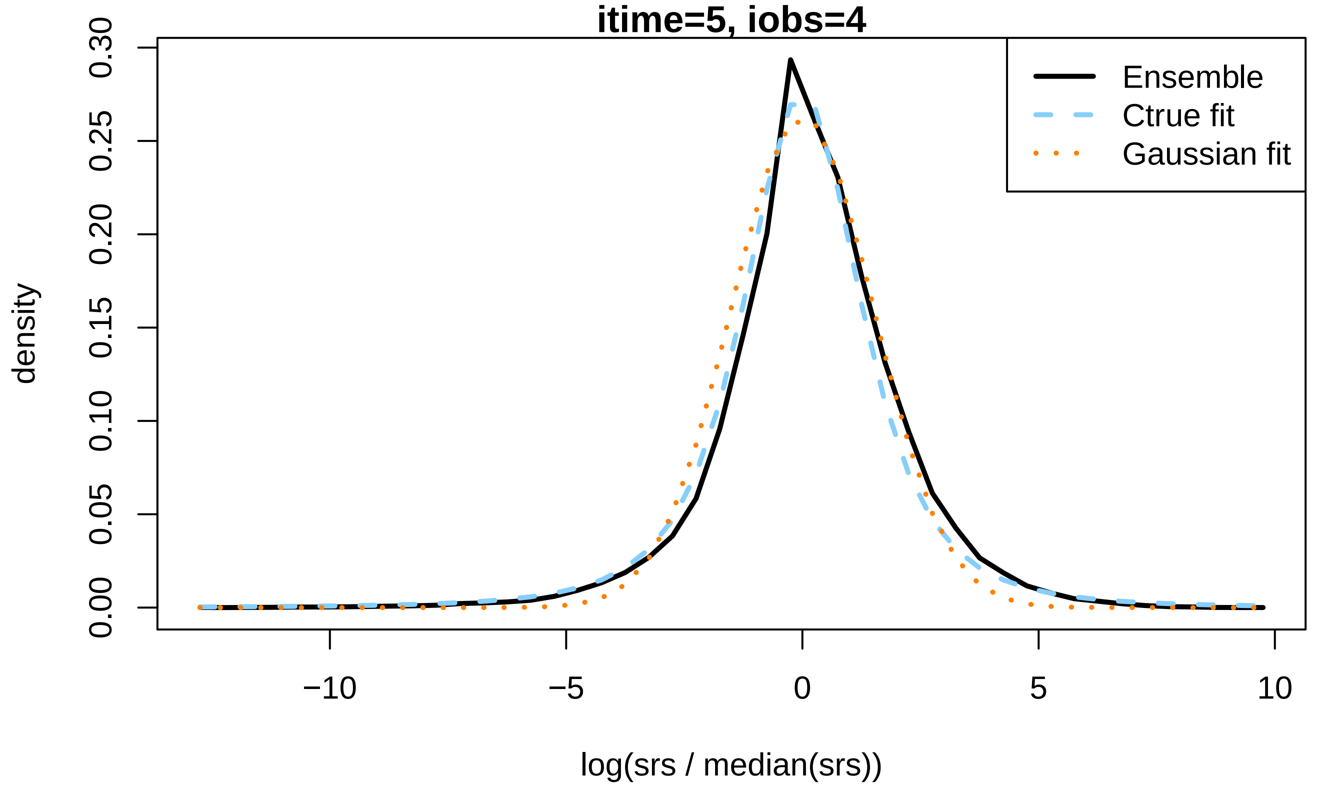


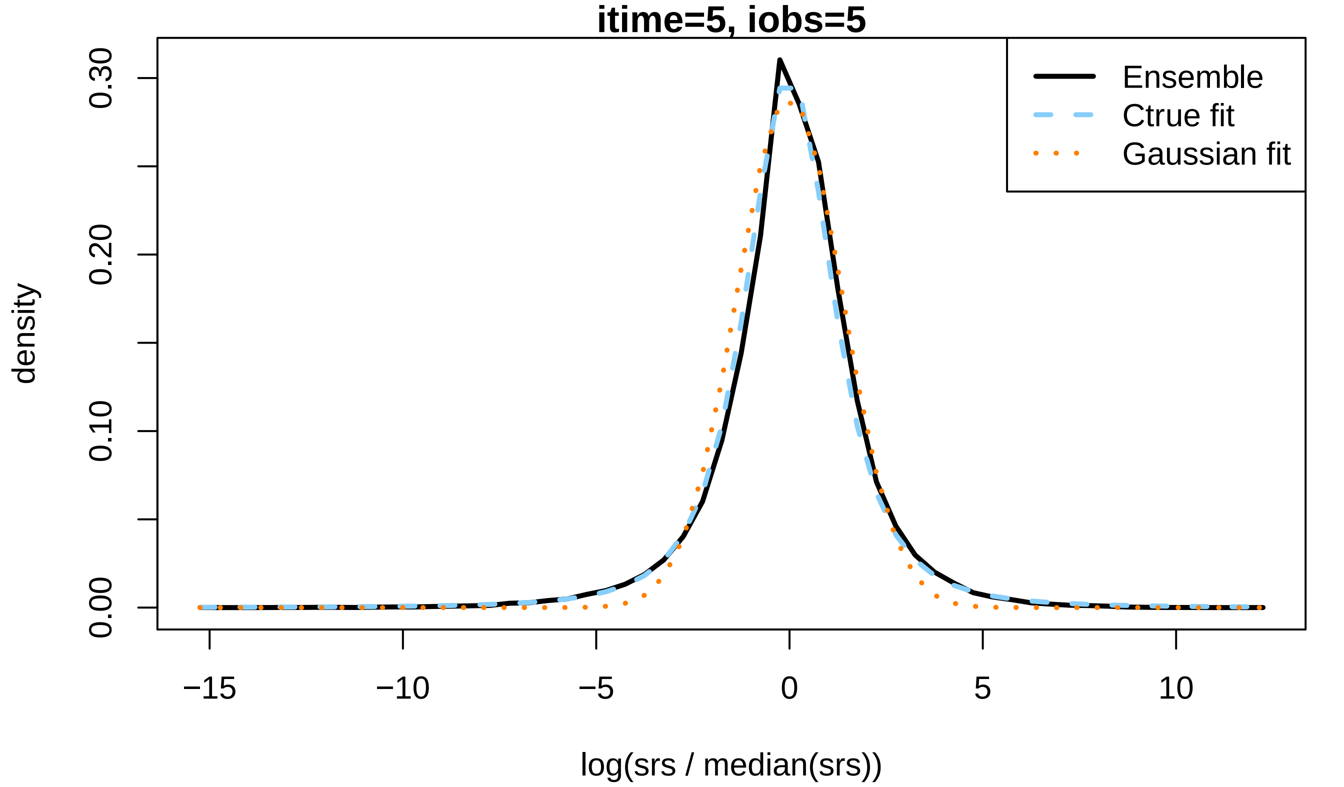
itime=5, iobs=1
density
0.00
0.10
0.20
0.30
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

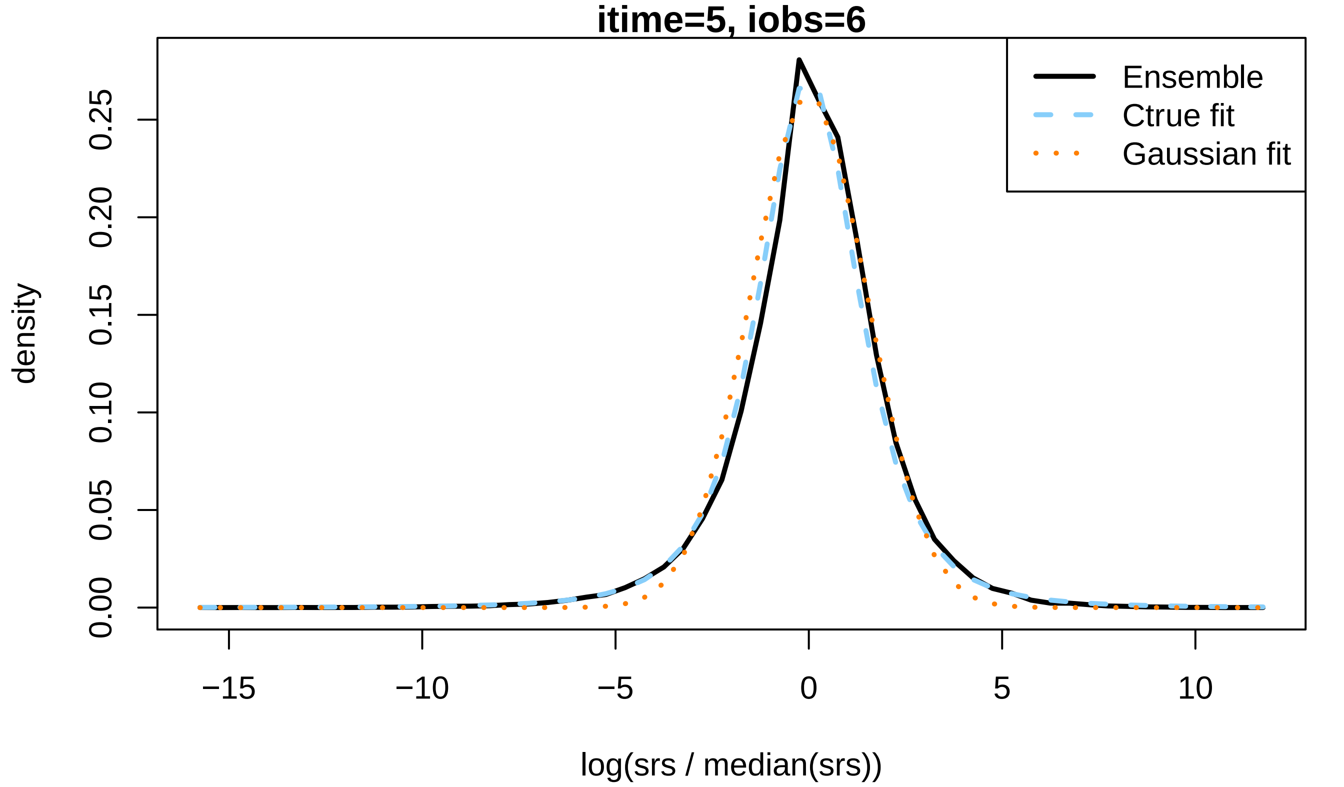


itime=5, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

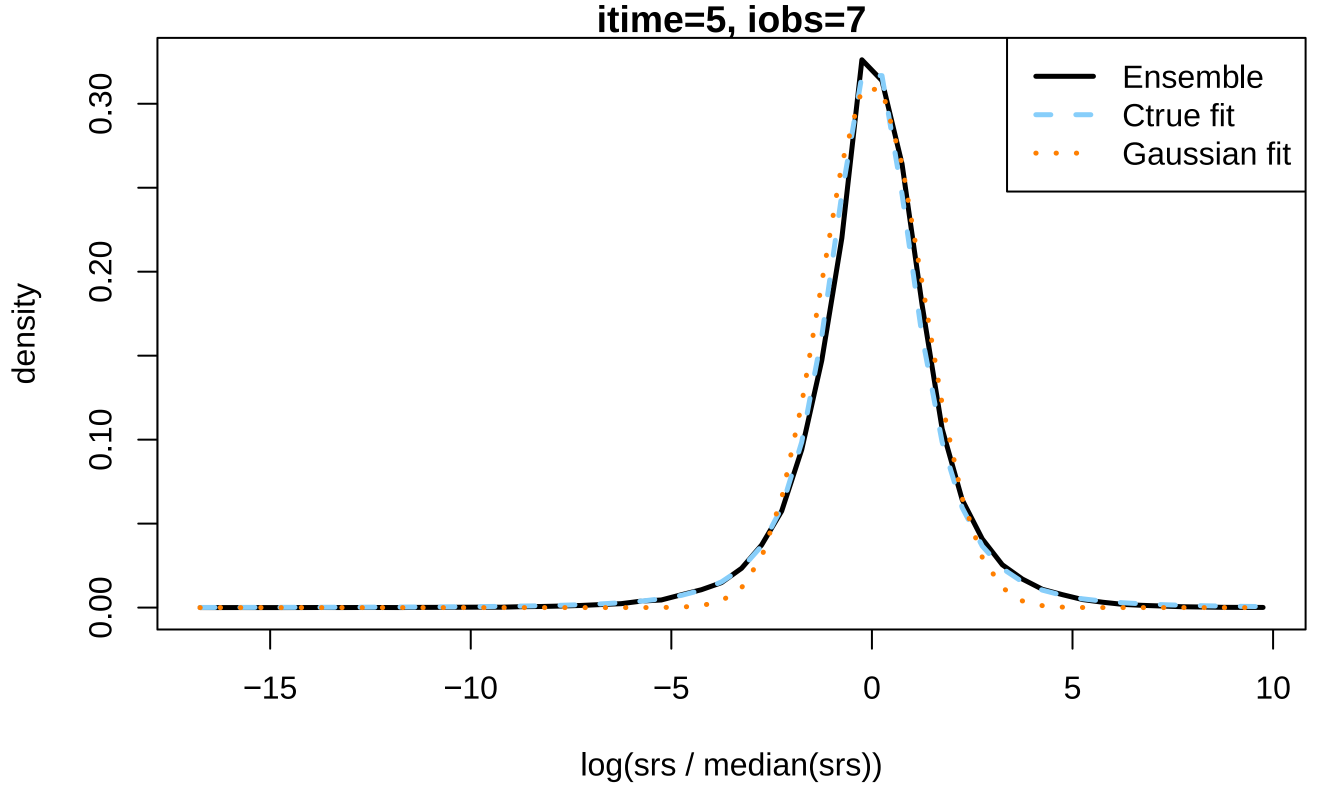

itime=5, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−20
−15
−10
−5
0
5
10
log(srs / median(srs))

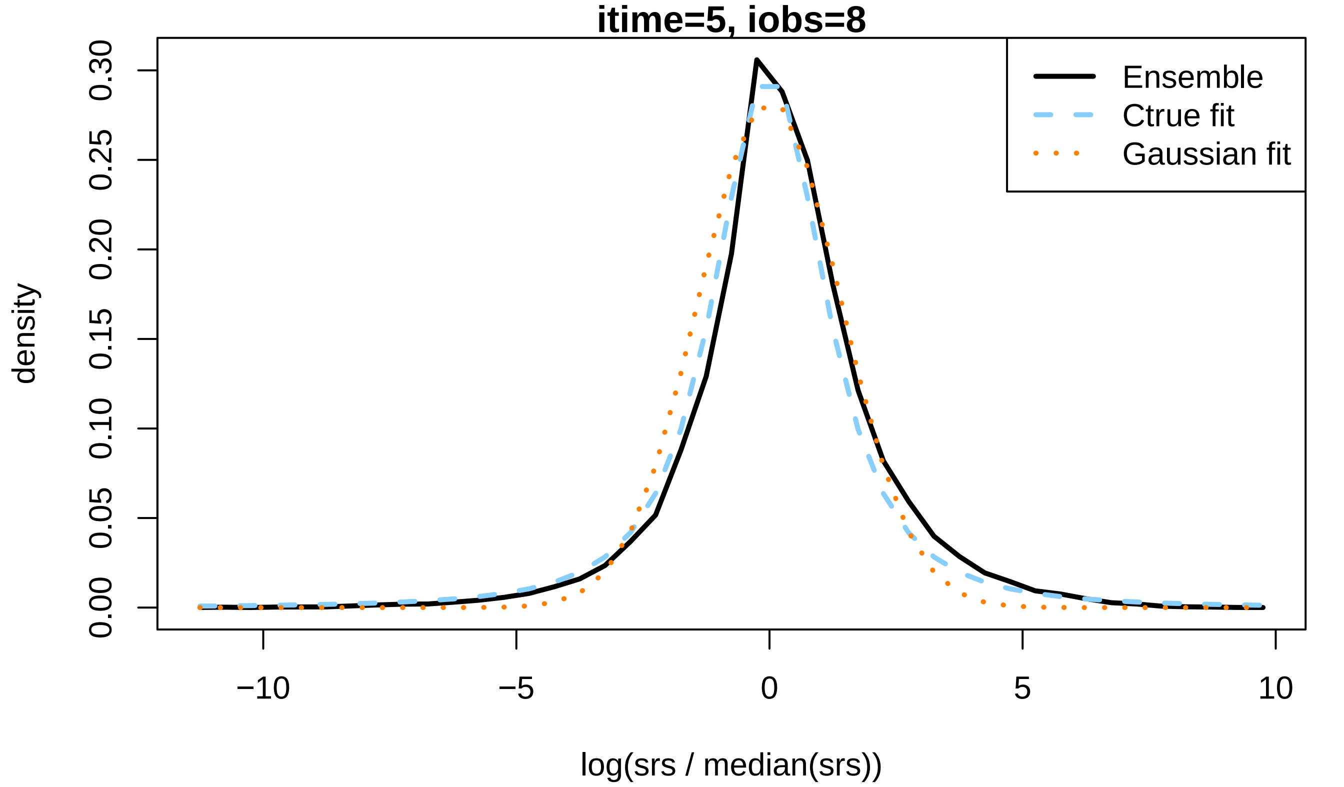

itime=5, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−10
−5
0
5
10
log(srs / median(srs))


itime=5, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

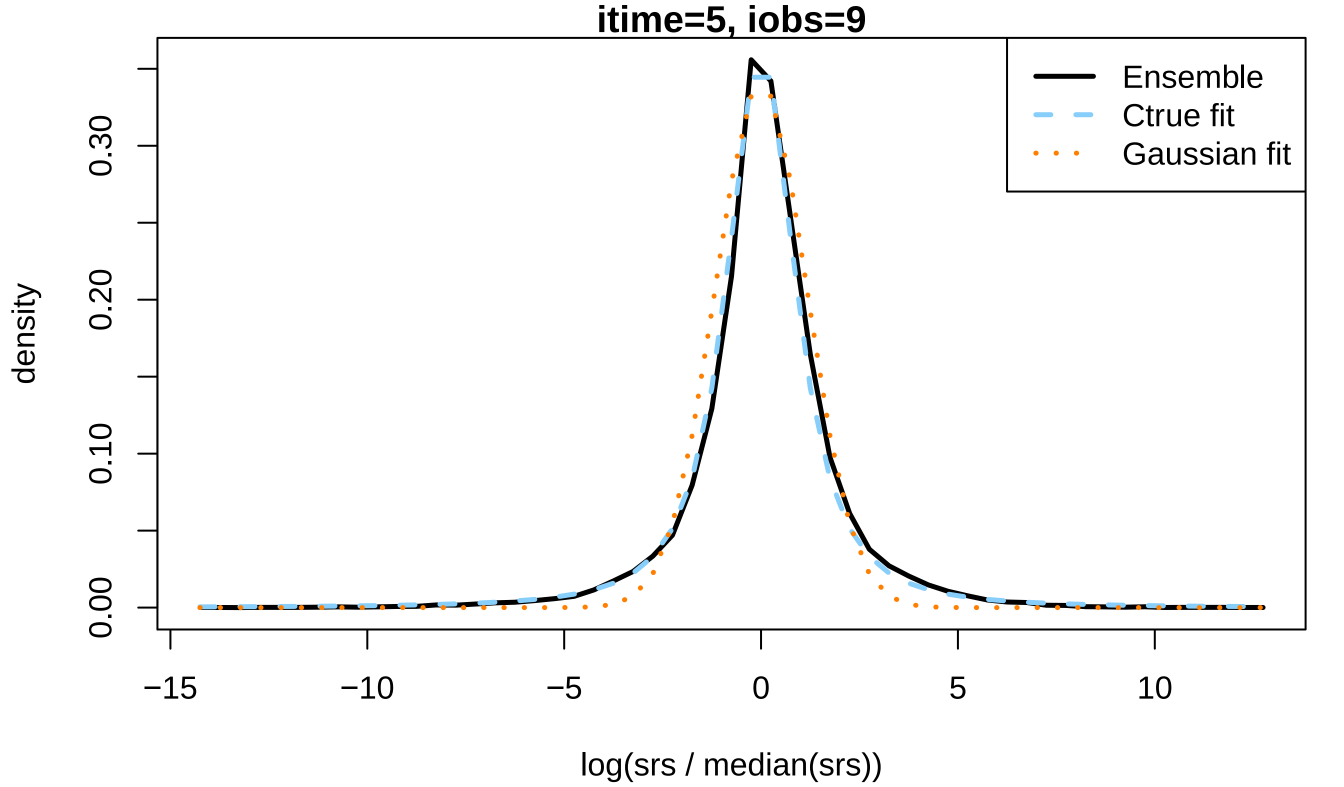


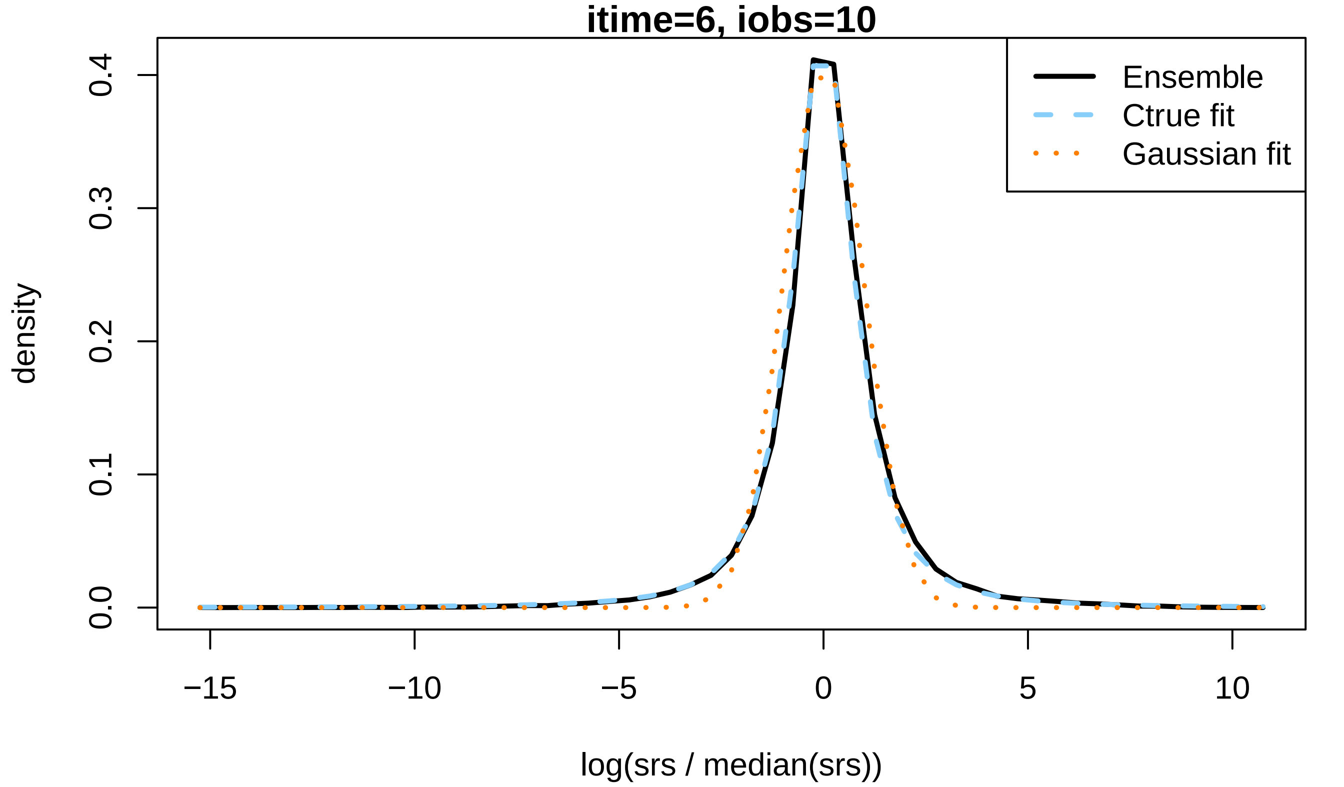
itime=5, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

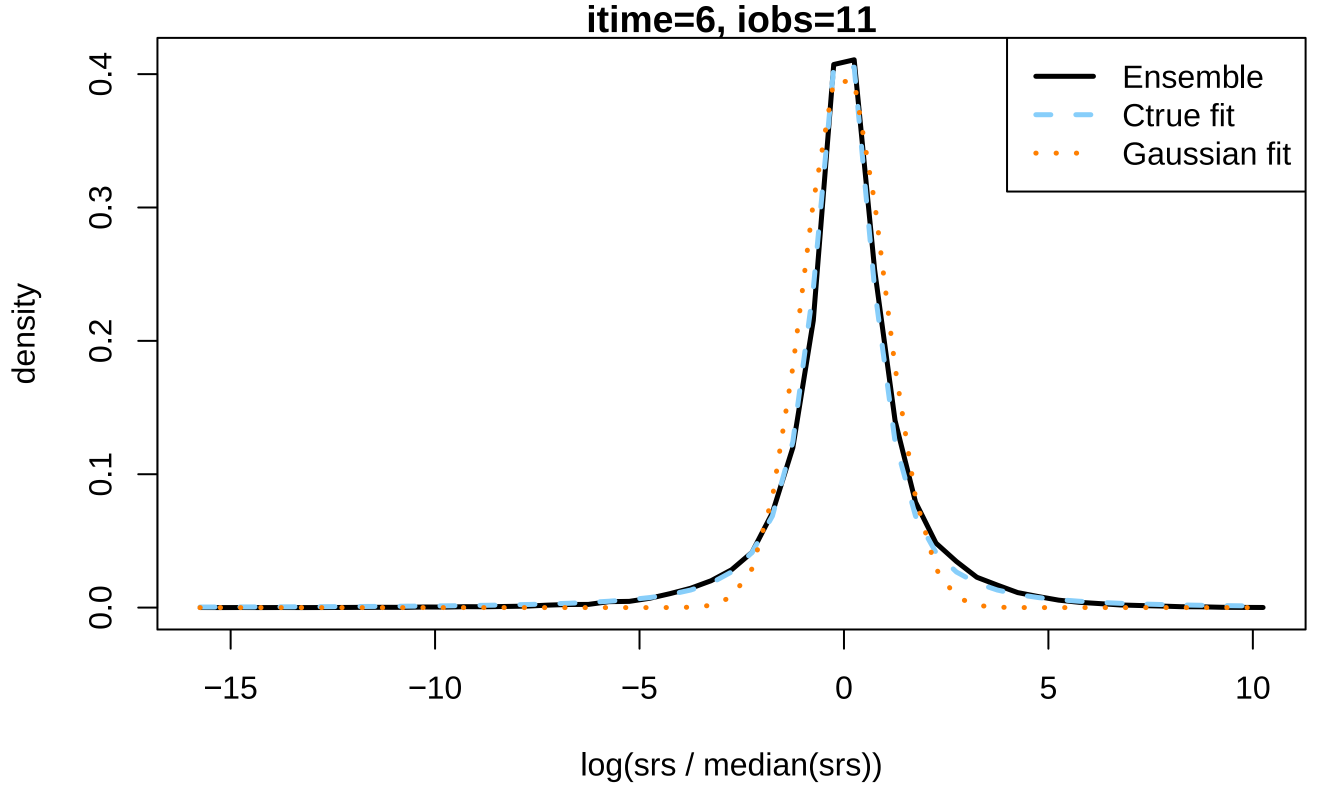

itime=5, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

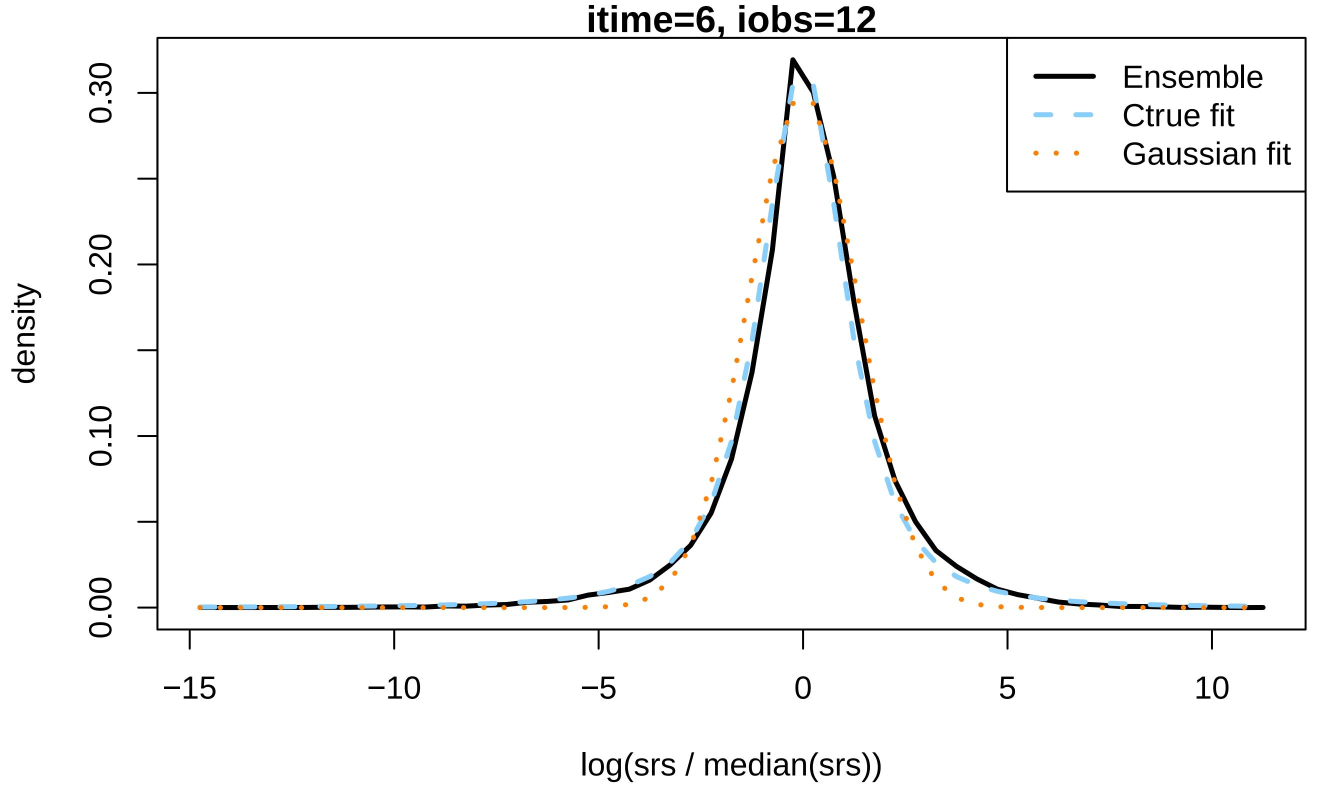


itime=5, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−10
−5
0
5
10
log(srs / median(srs))

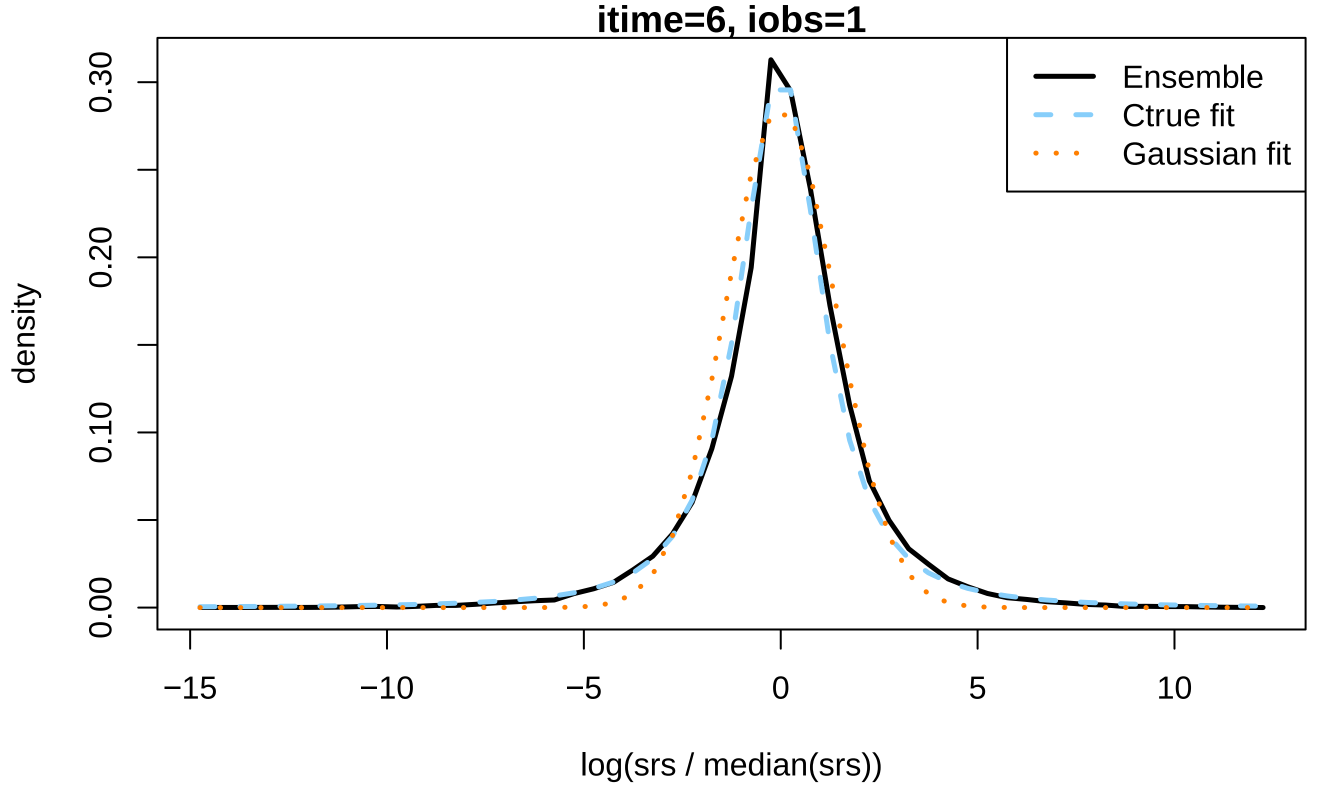

itime=5, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=6, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))

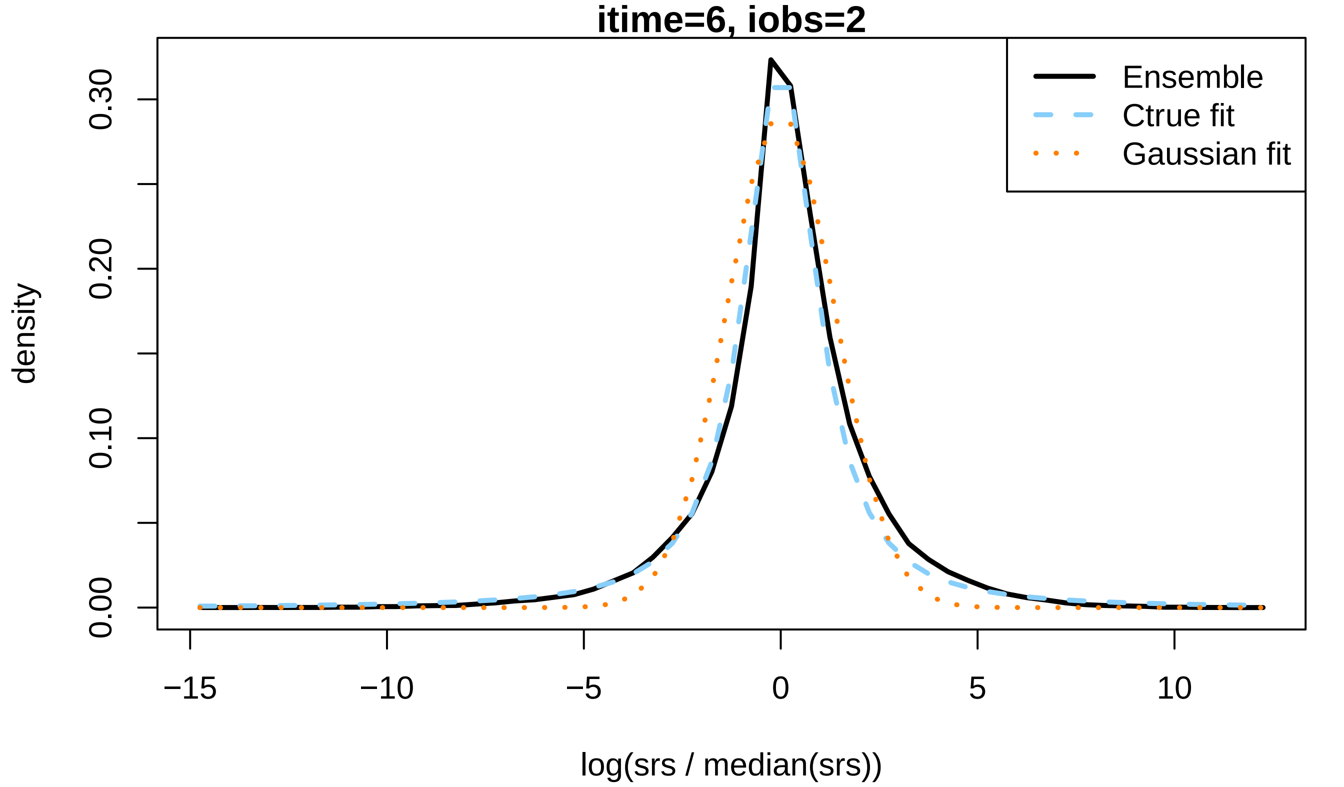

itime=6, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))



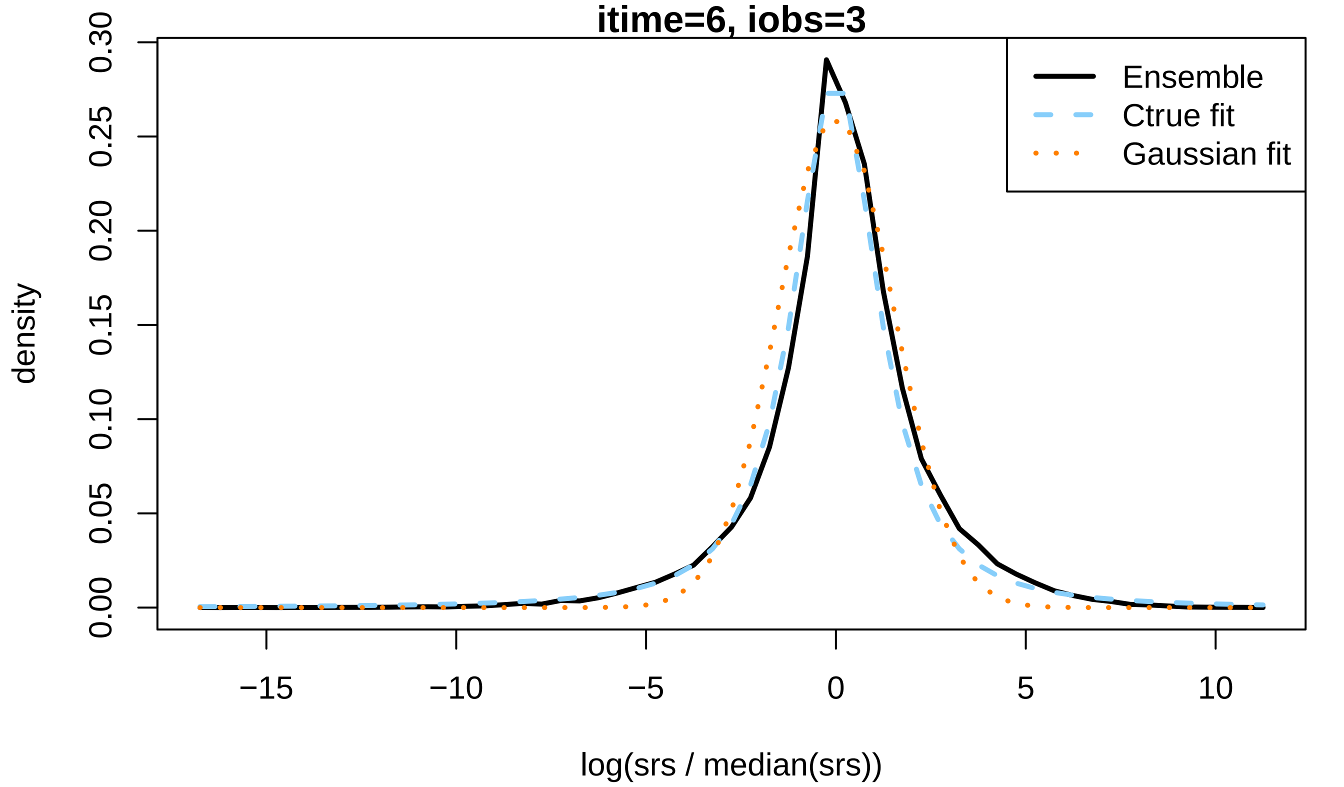
itime=6, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

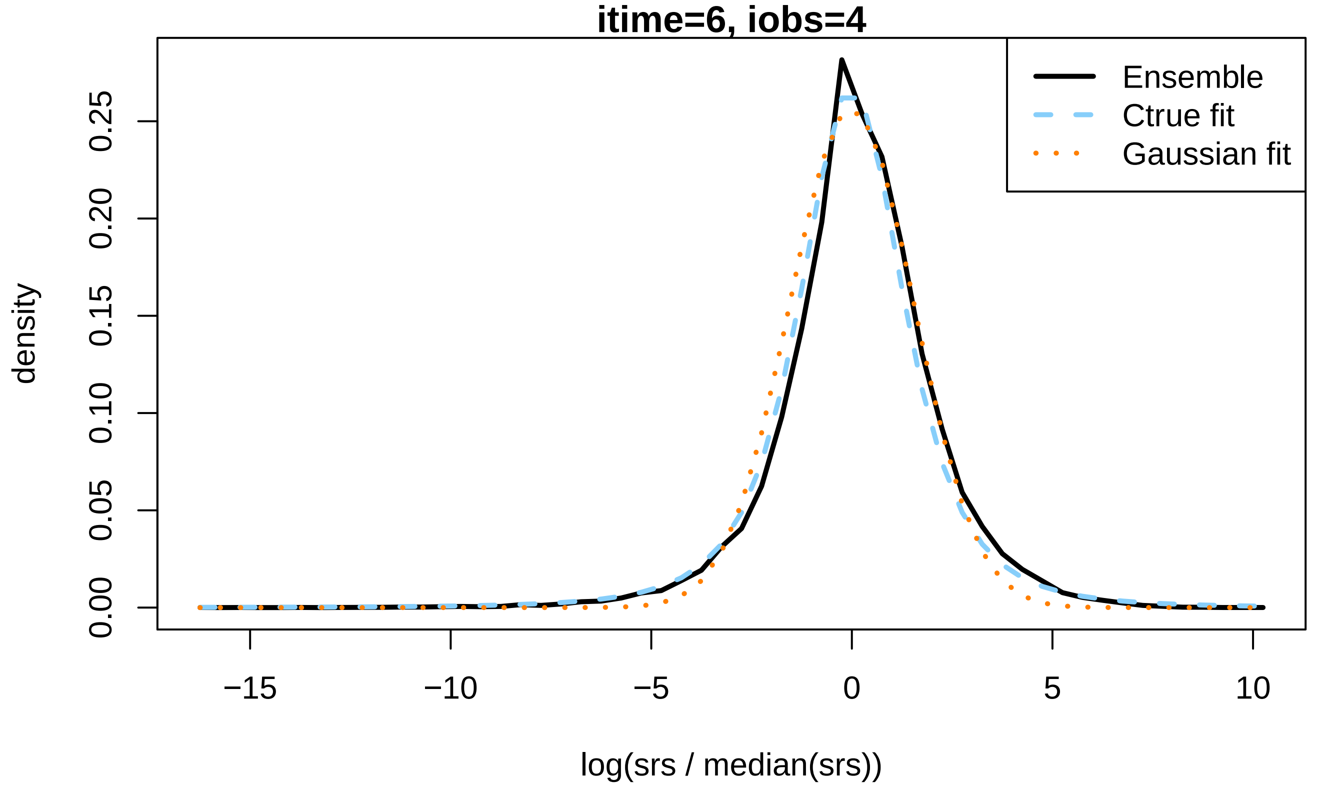


itime=6, iobs=1
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

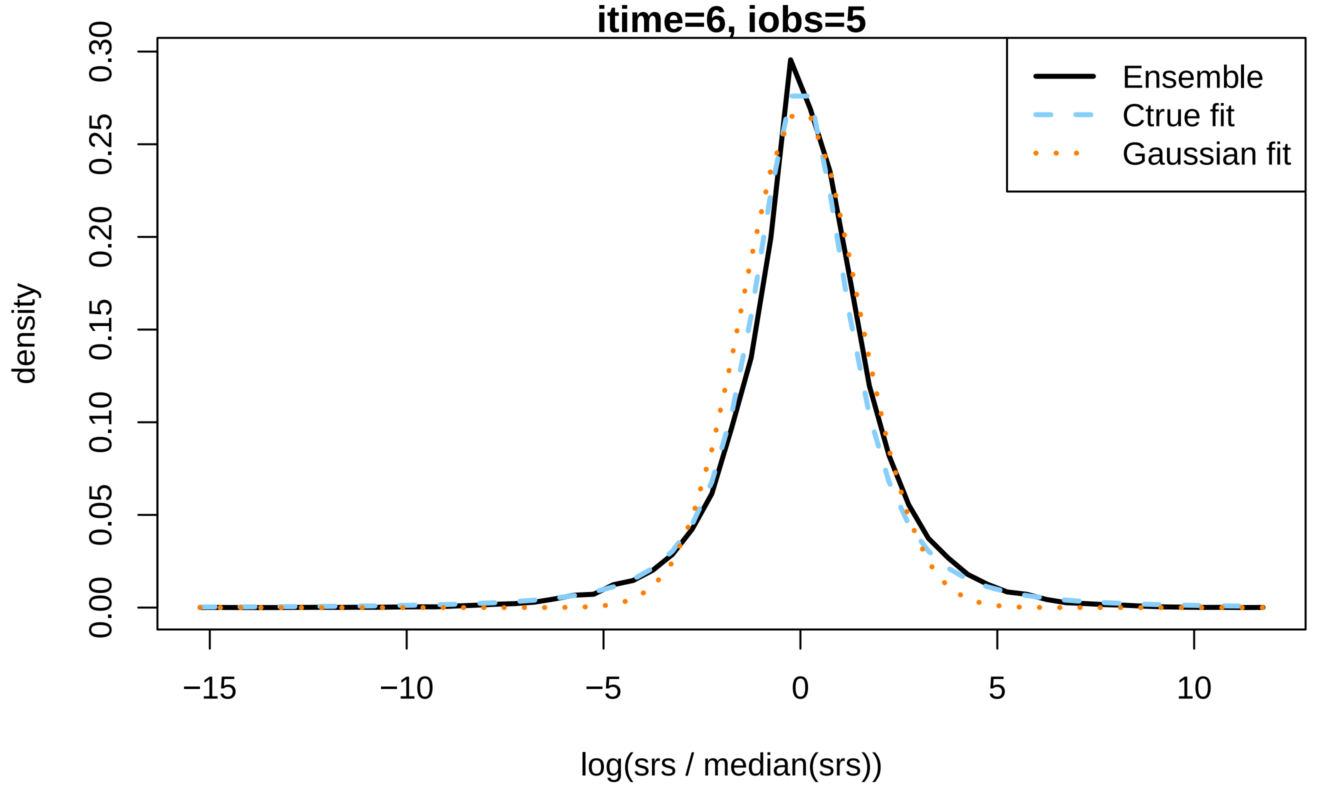


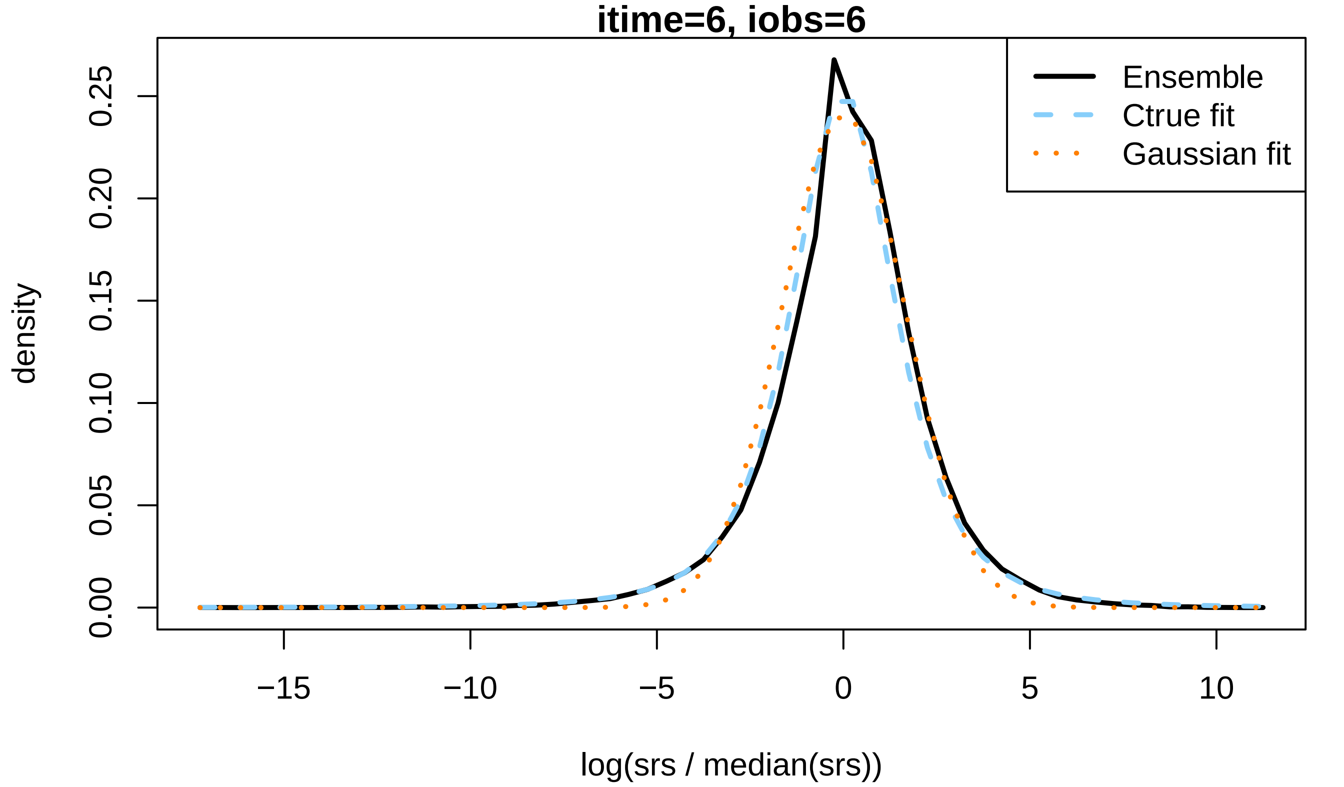
itime=6, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

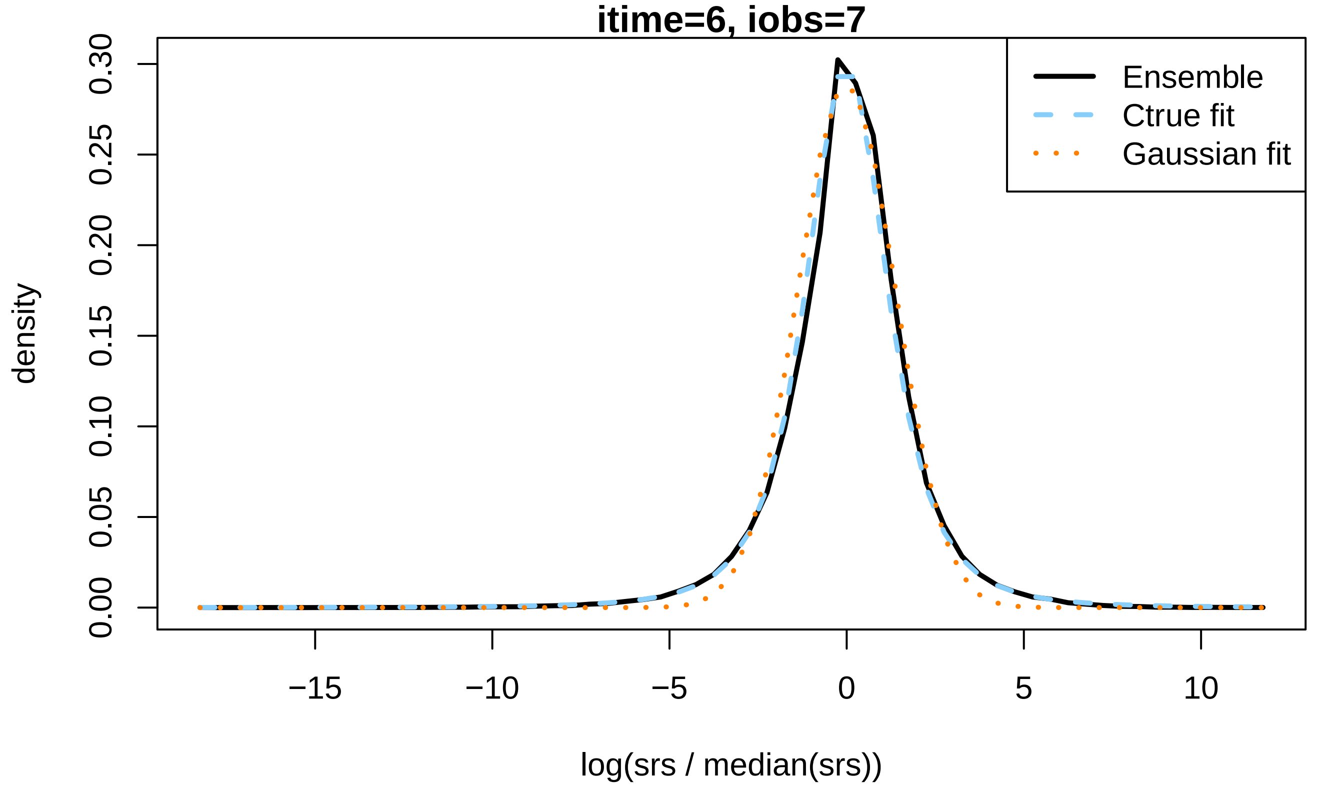

itime=6, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



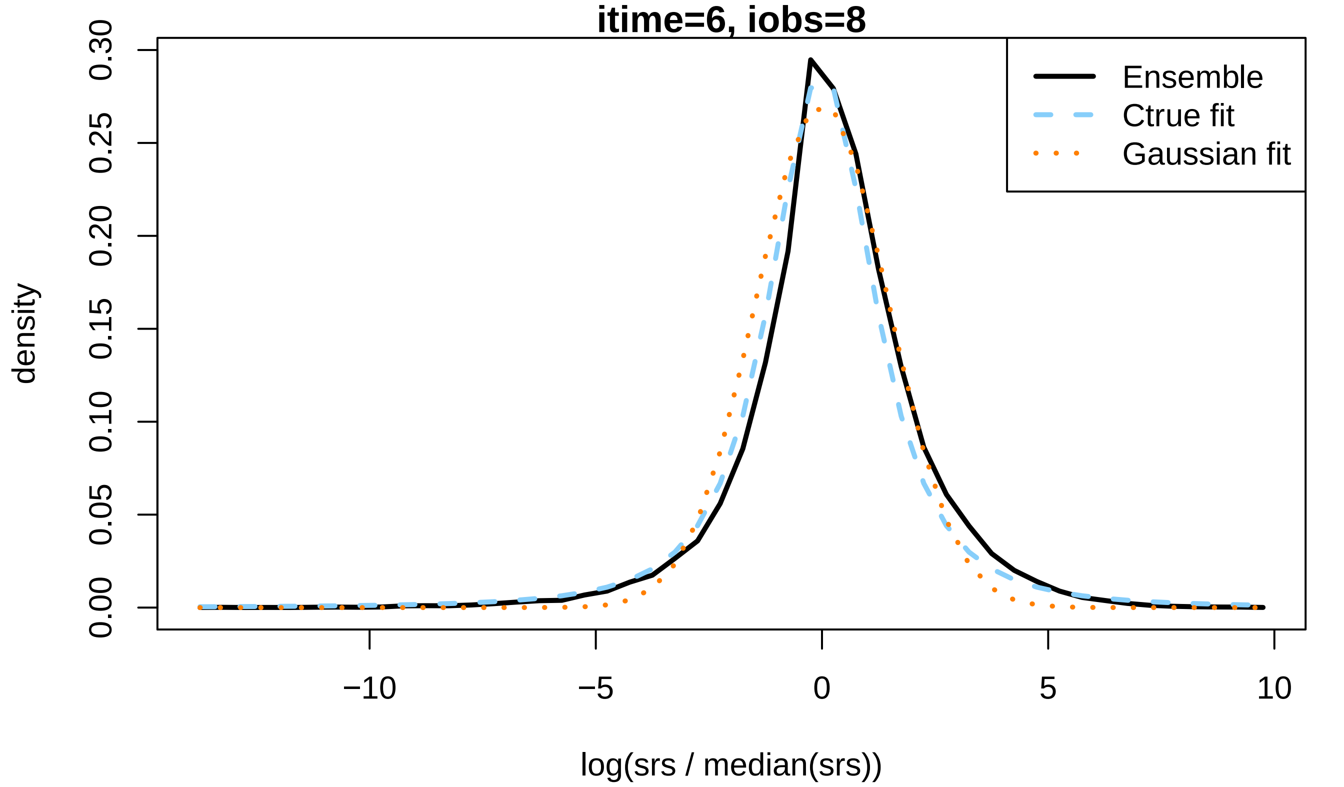
itime=6, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

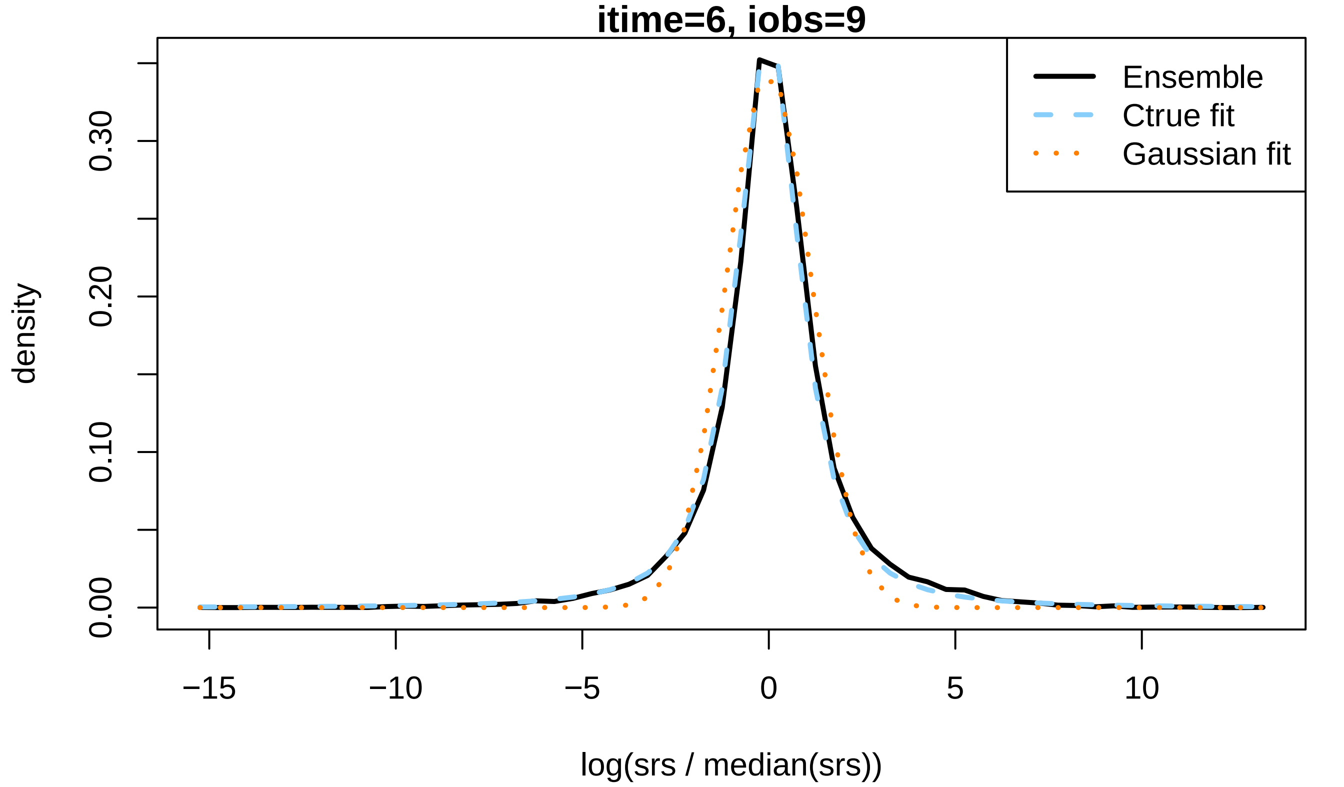

itime=6, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

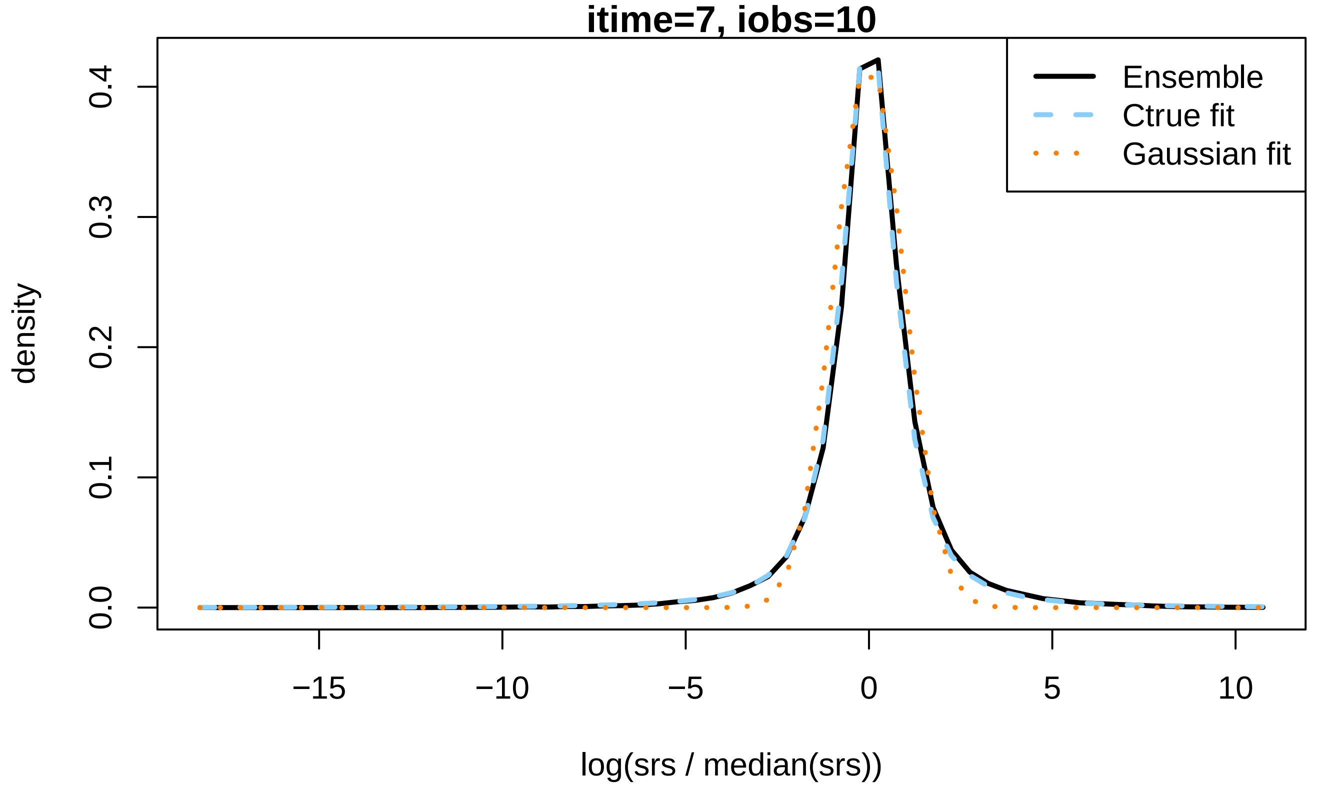

itime=6, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

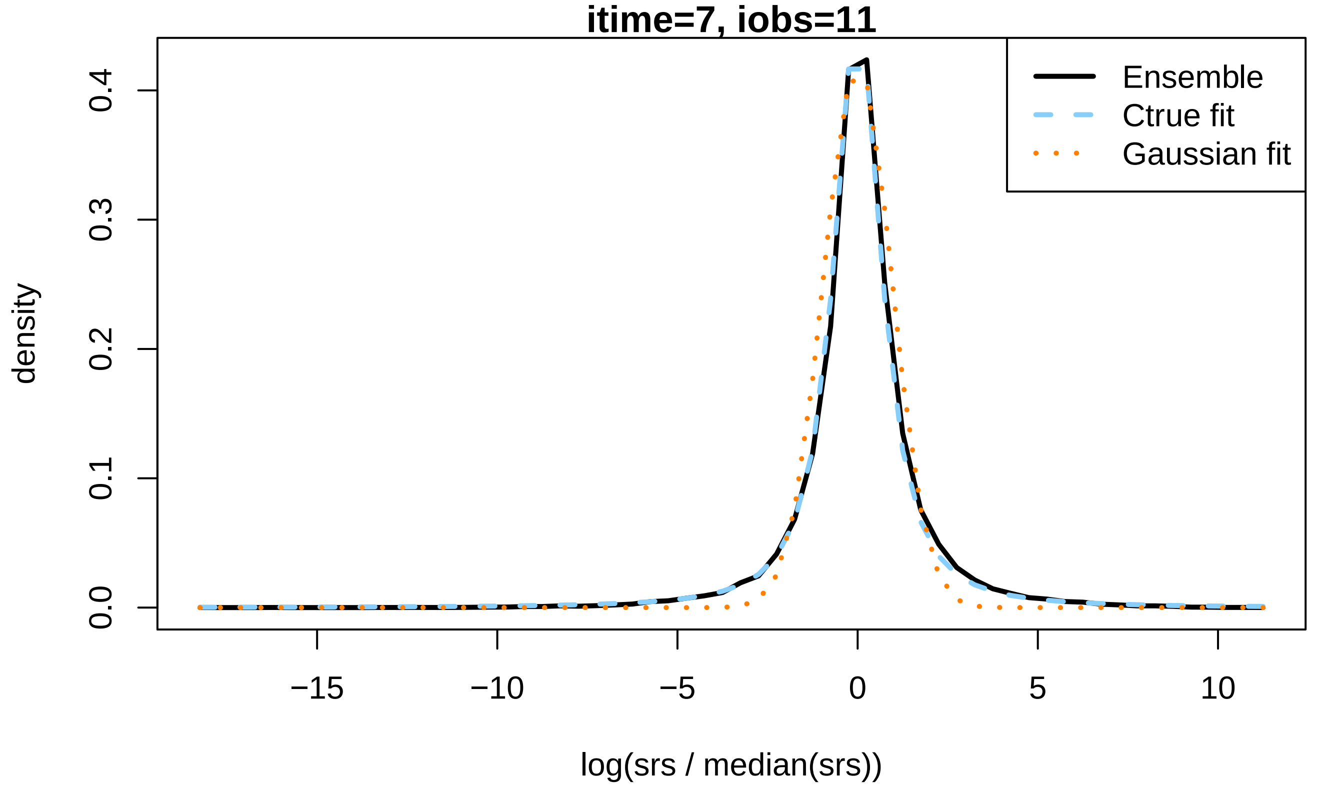


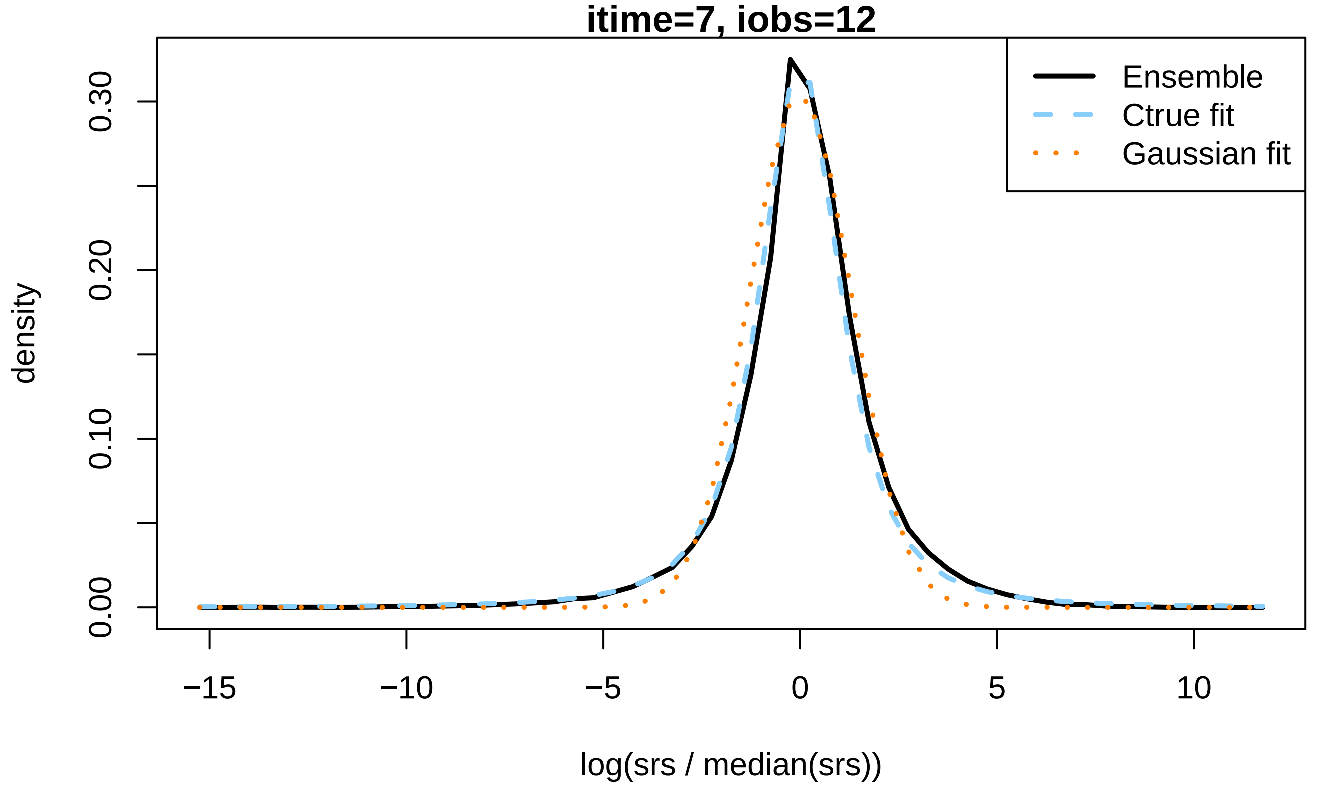
itime=6, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

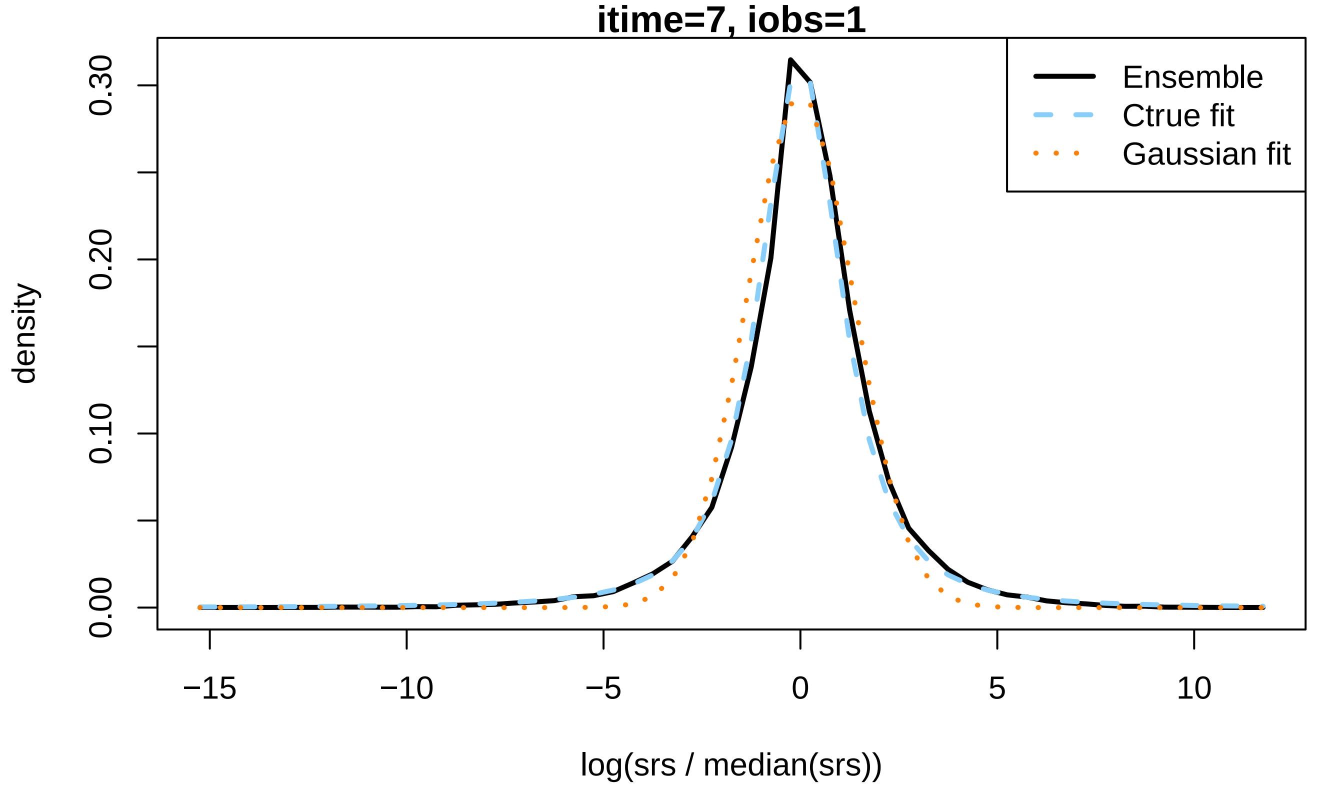

itime=6, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−10
−5
0
5
10
log(srs / median(srs))


itime=6, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

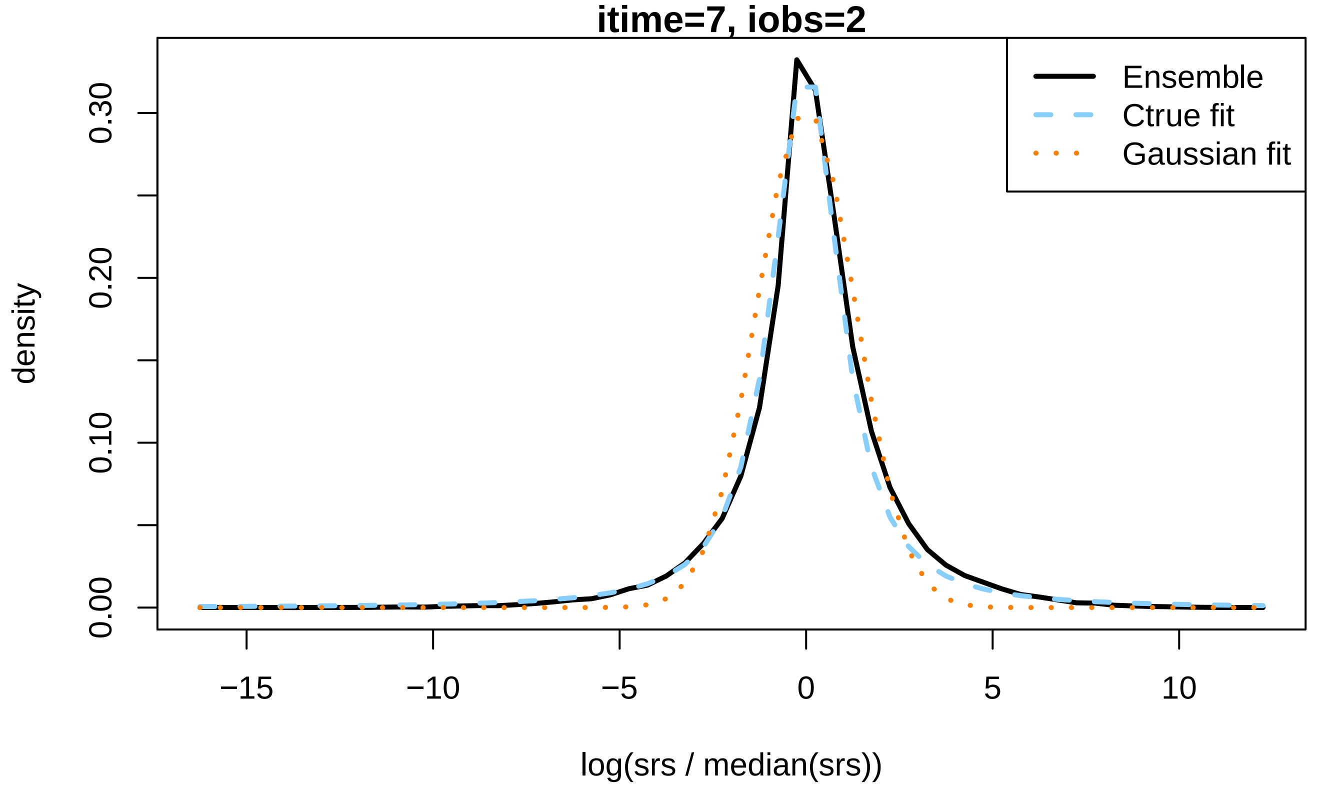

itime=7, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))

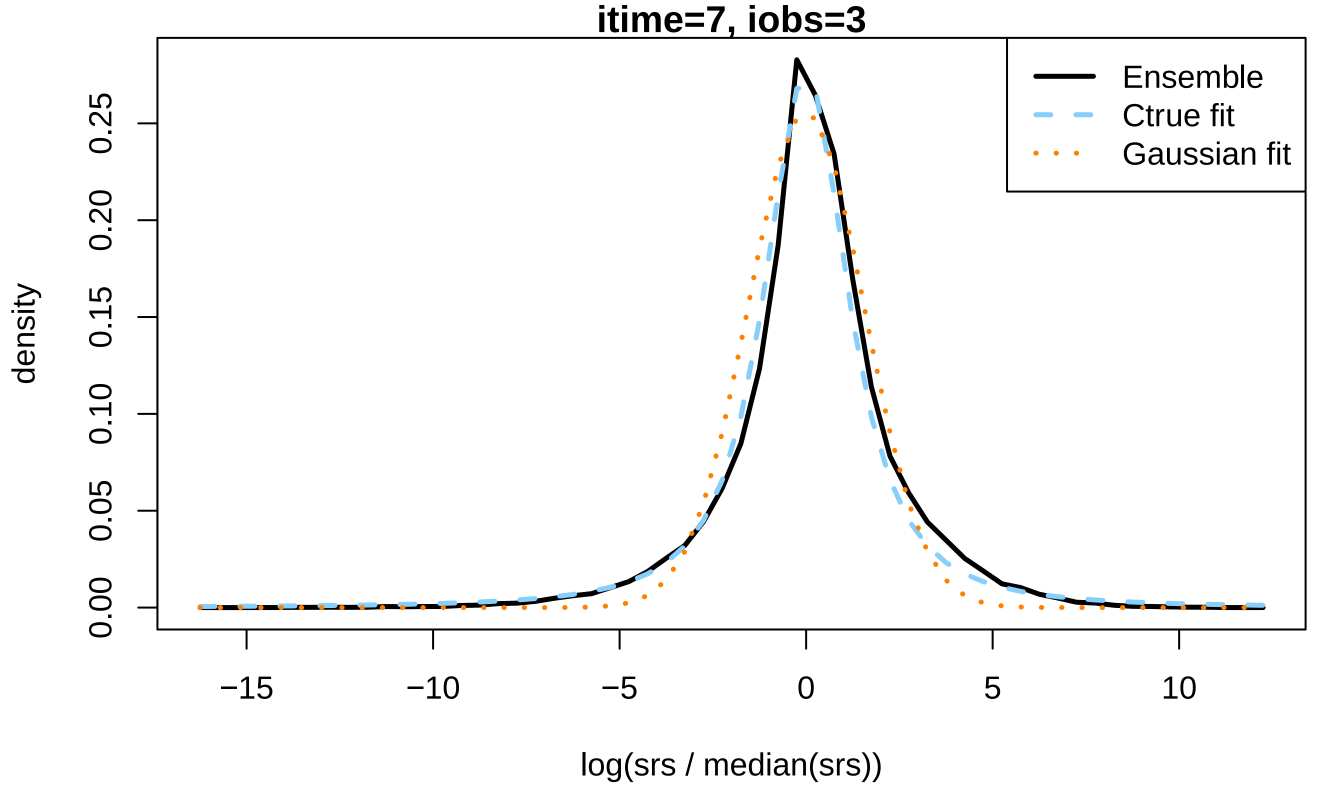

itime=7, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=7, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

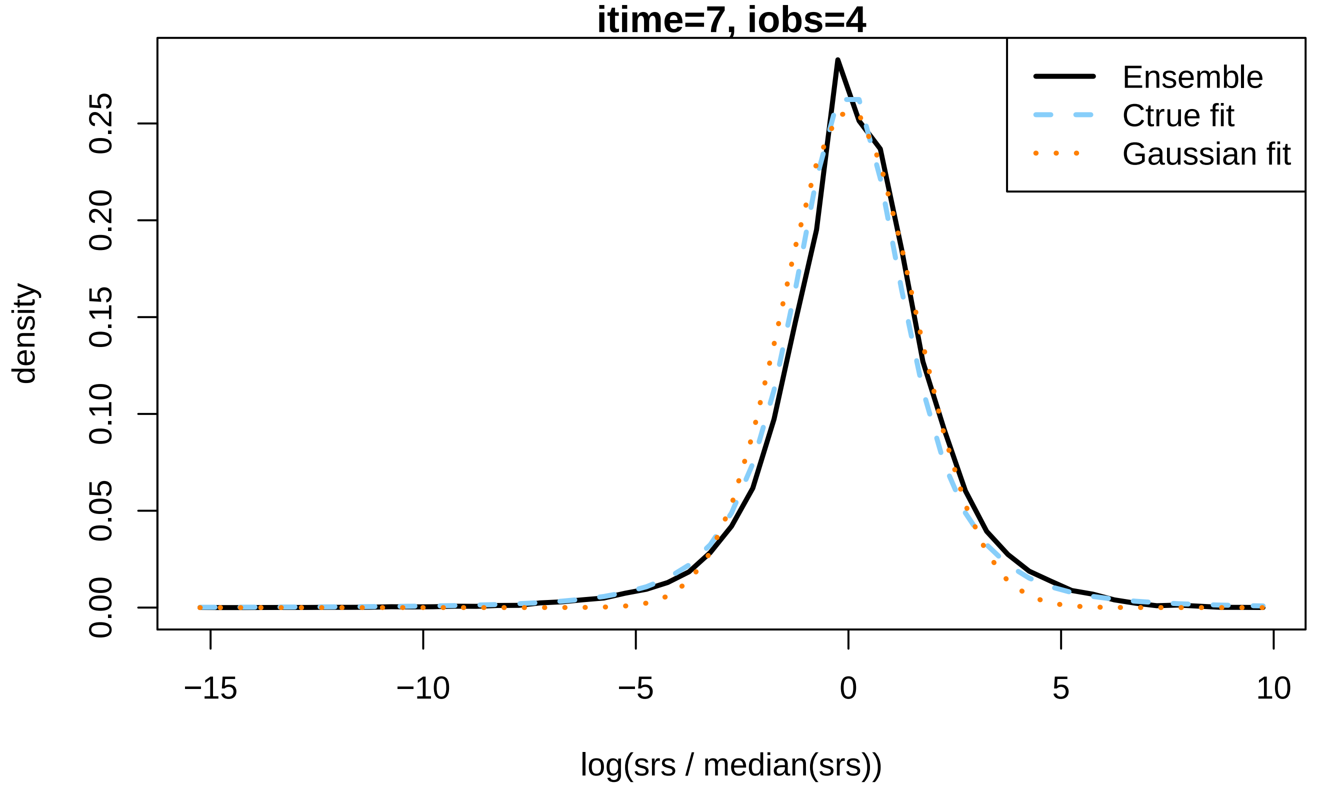


itime=7, iobs=1
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit

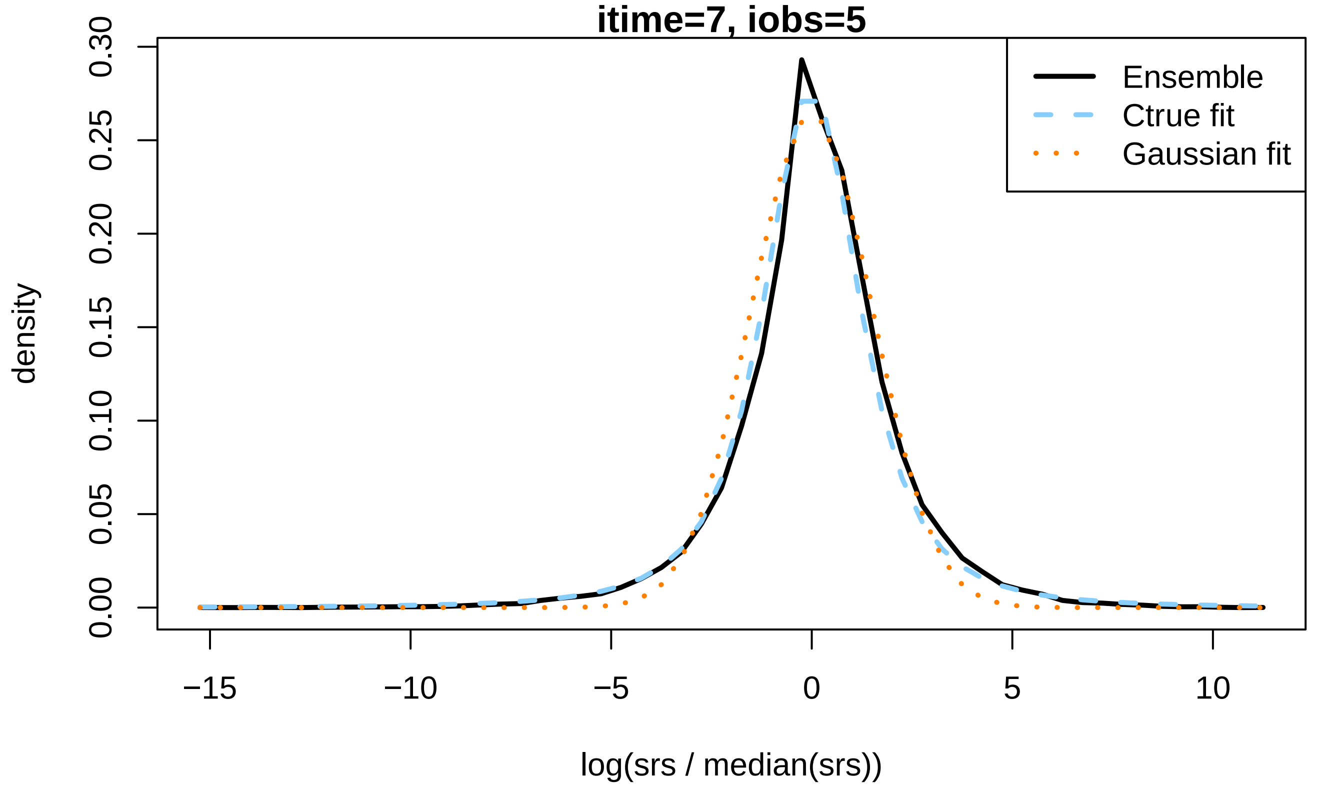

itime=7, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))



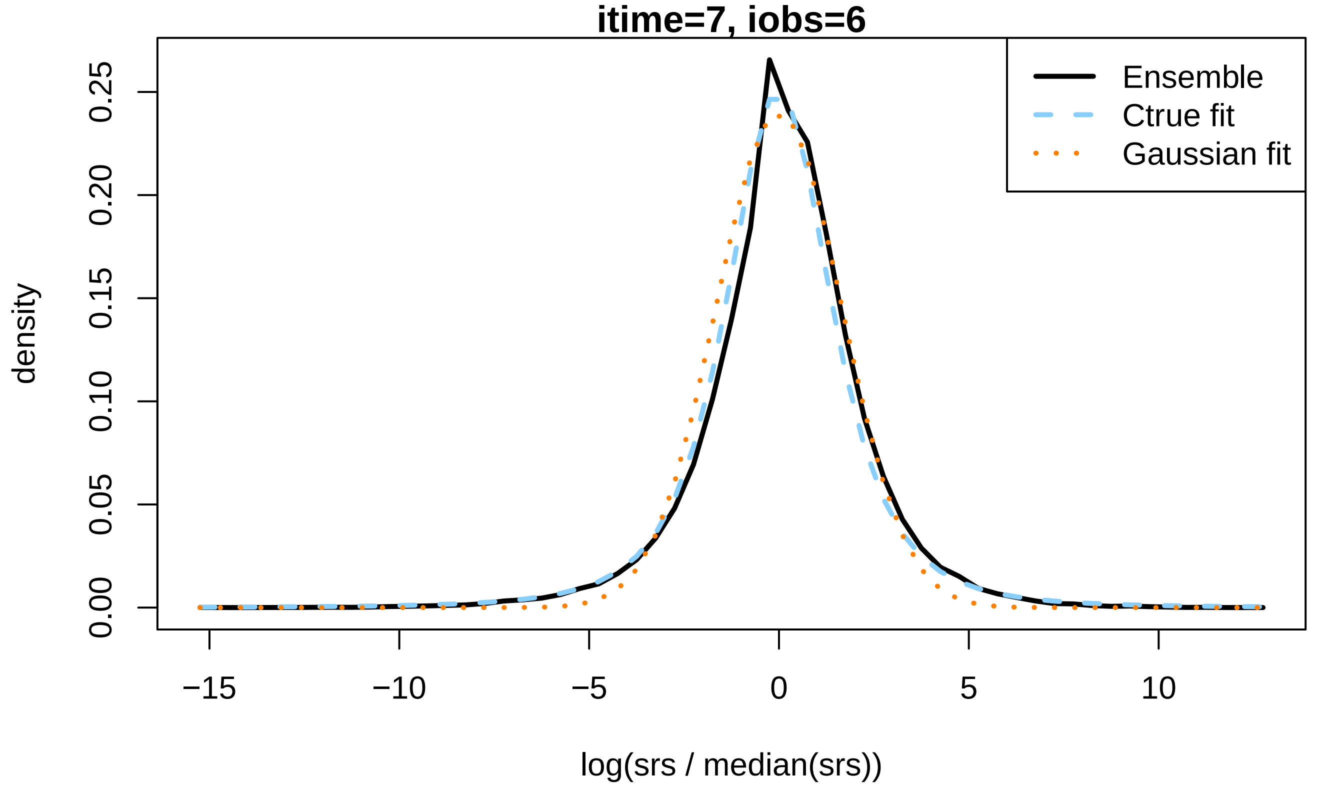
itime=7, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))



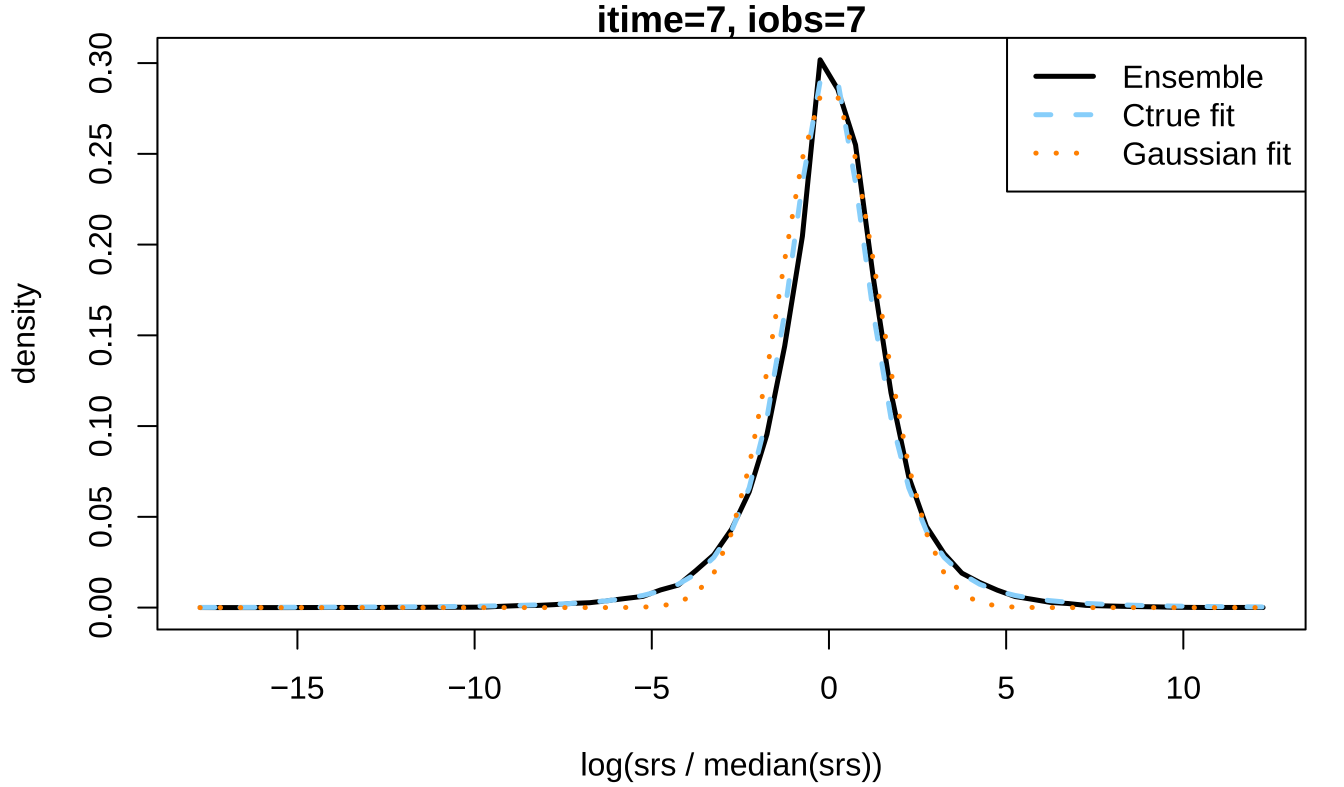
itime=7, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

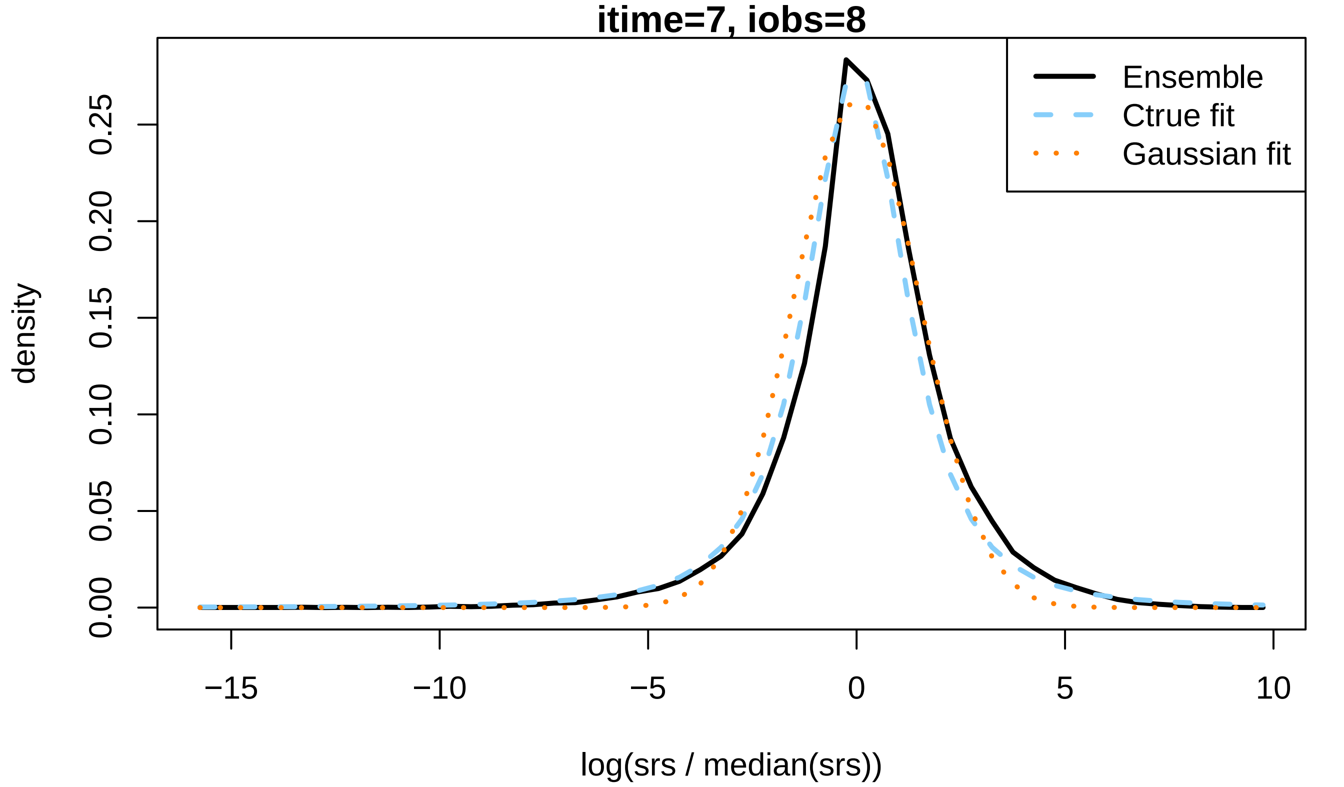


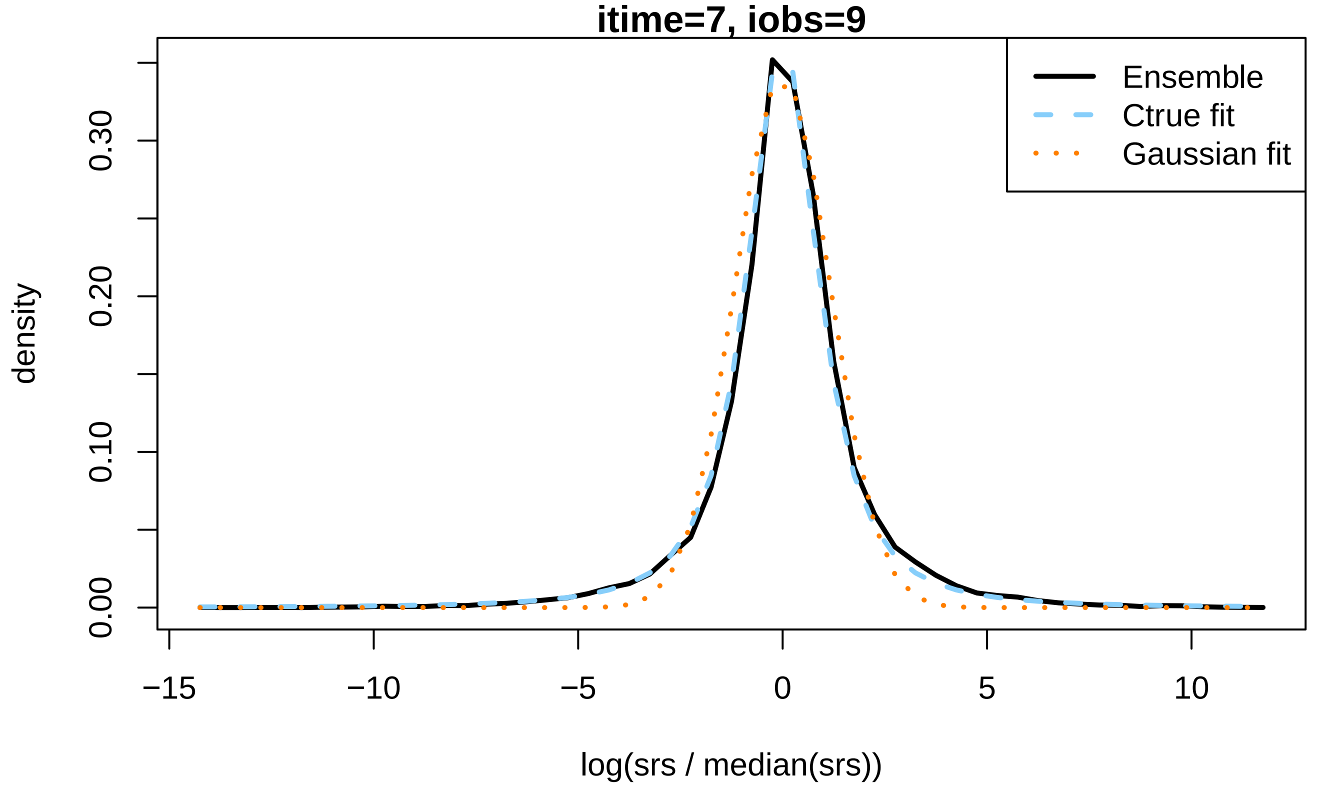
itime=7, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

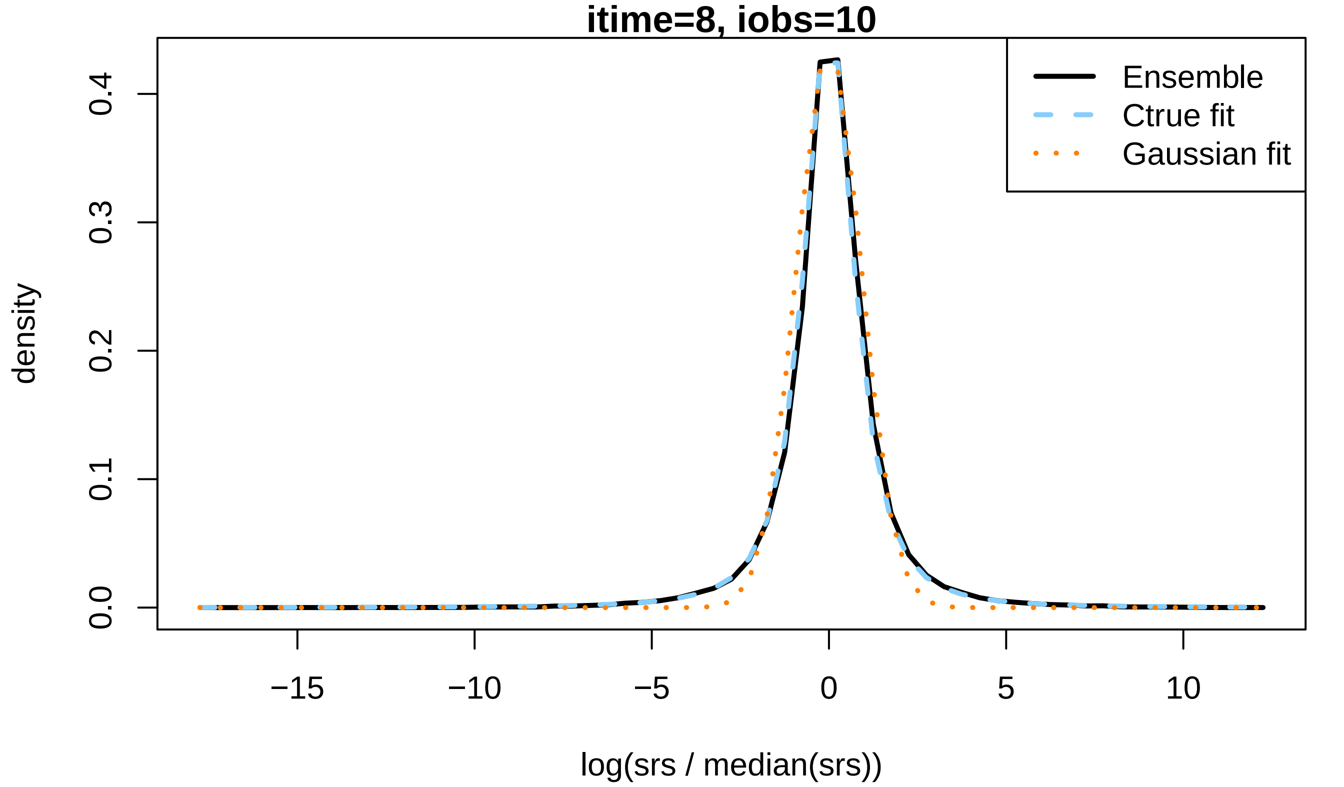


itime=7, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

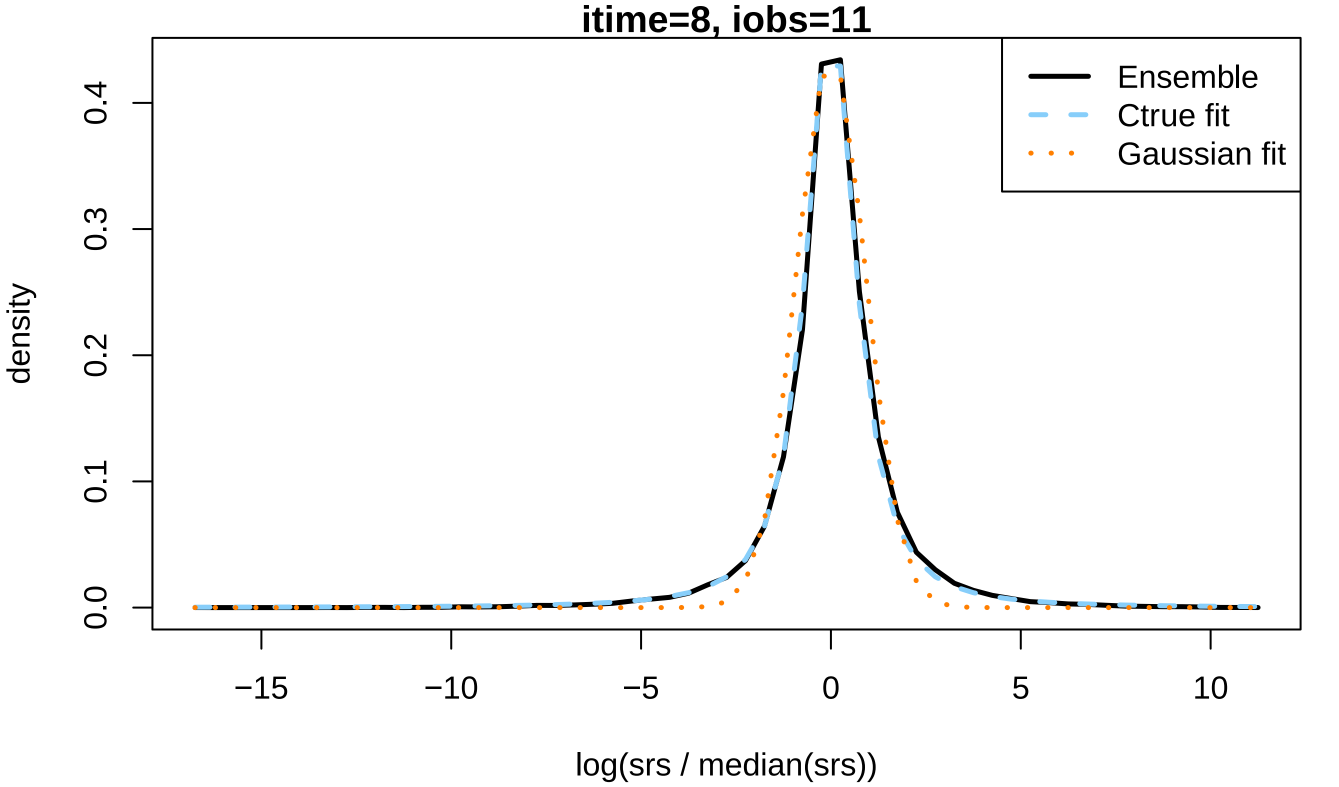

itime=7, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

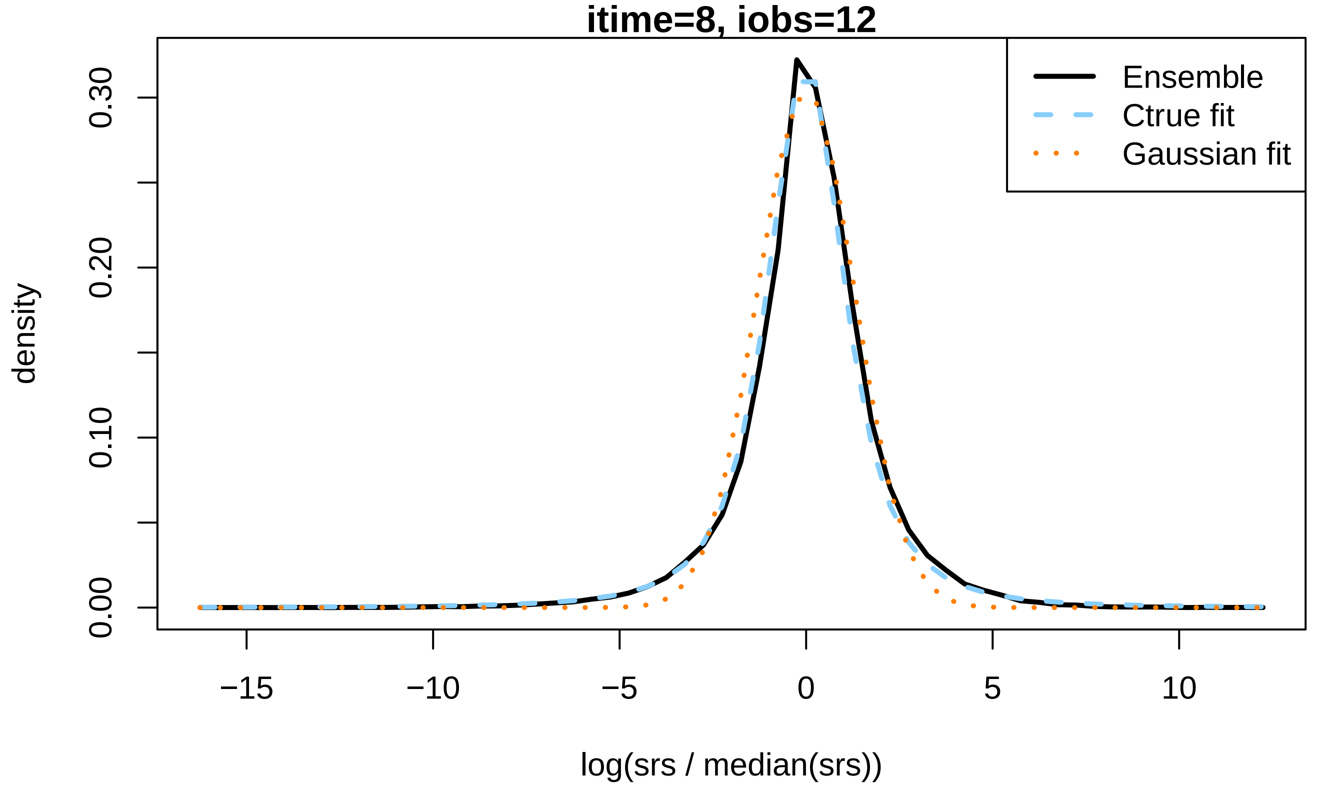

itime=7, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

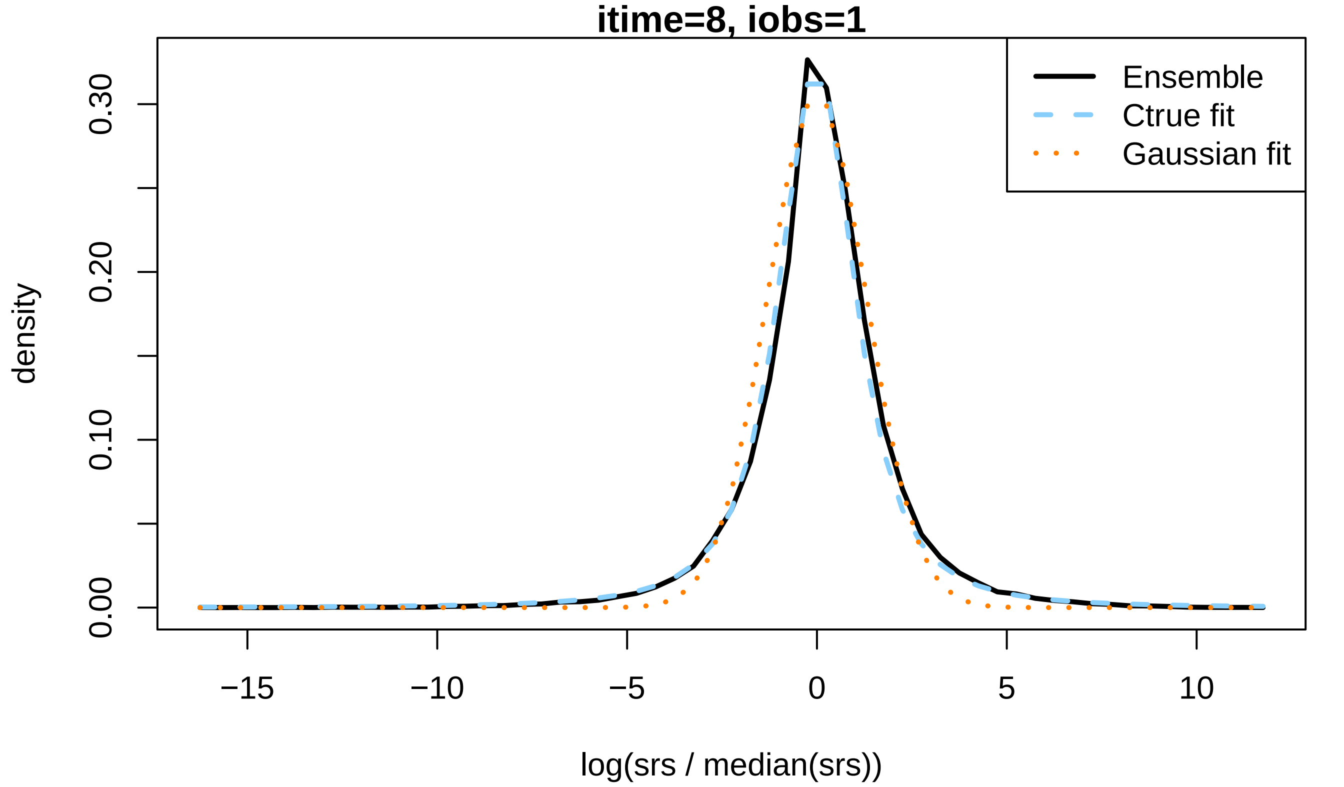

itime=7, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

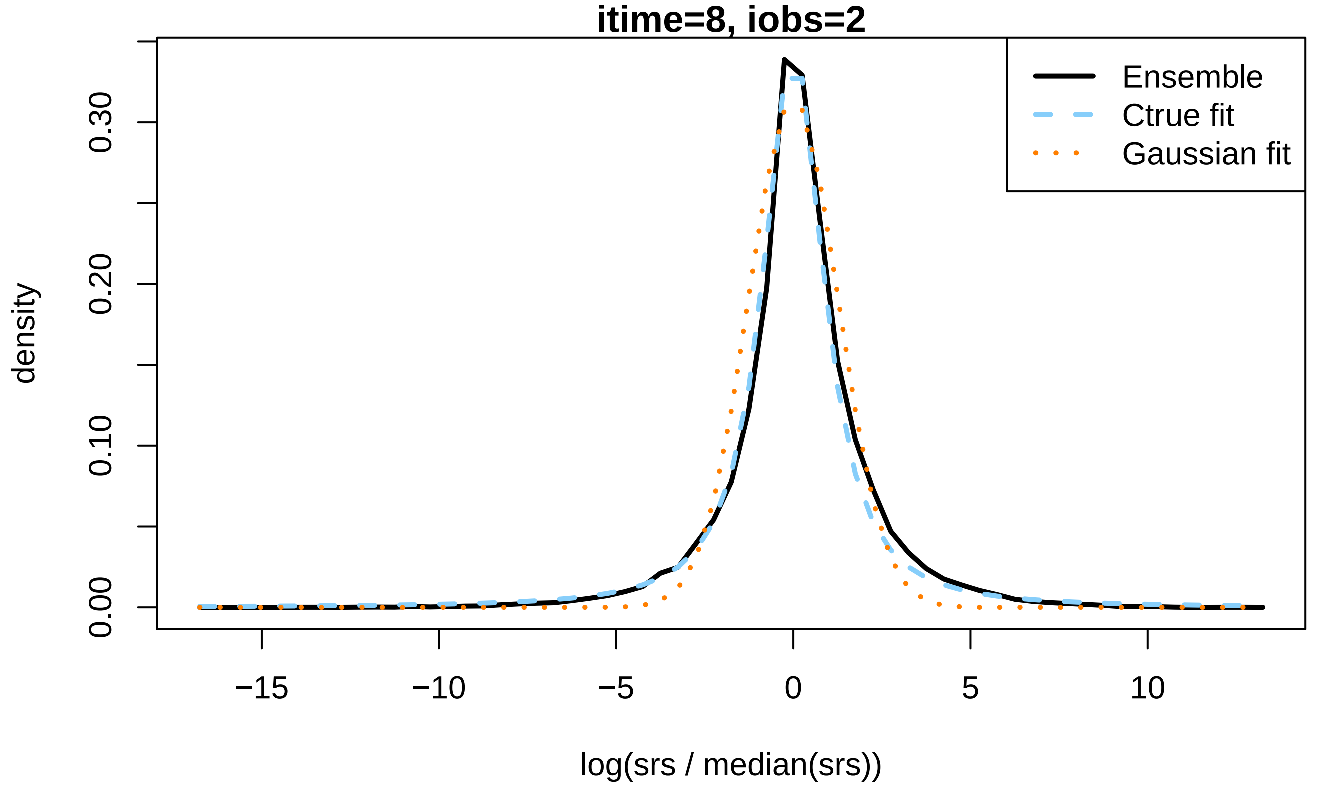


itime=8, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))

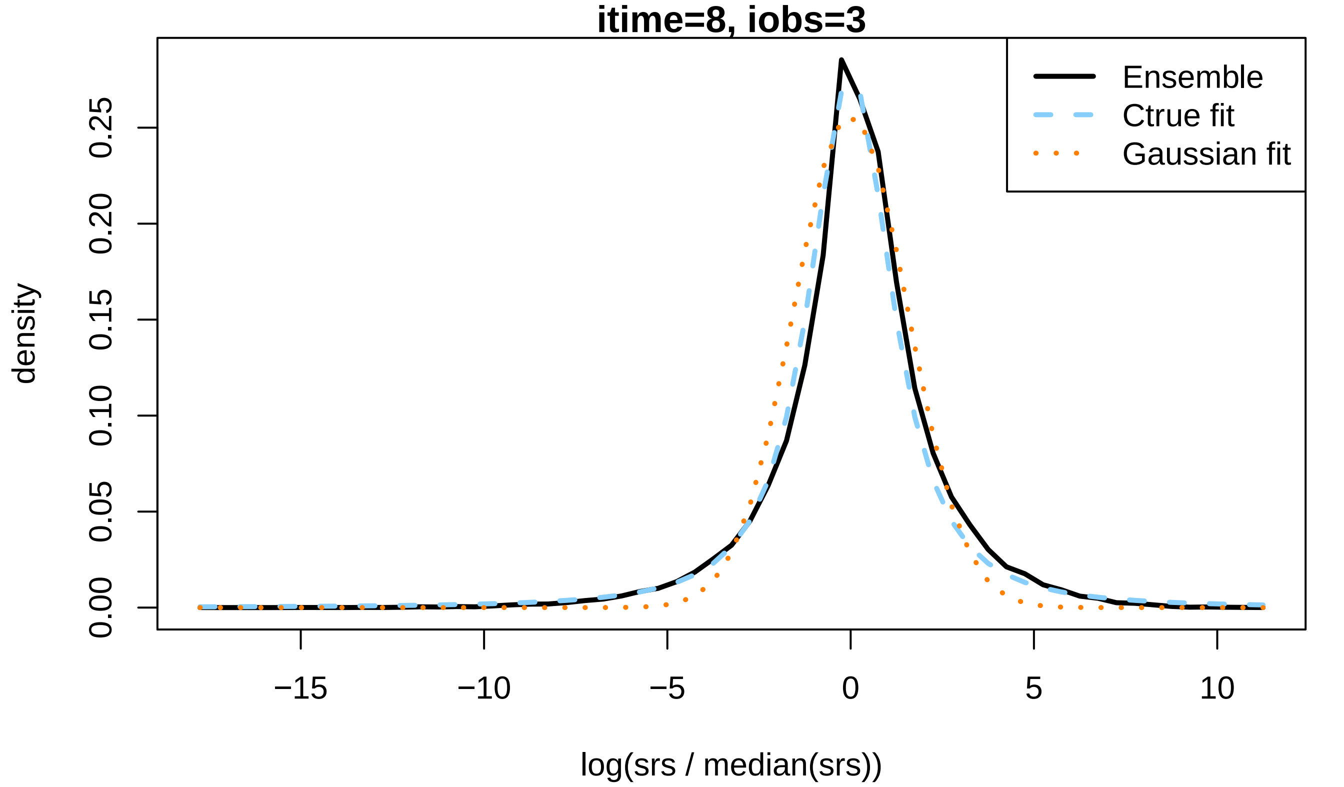

itime=8, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=8, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

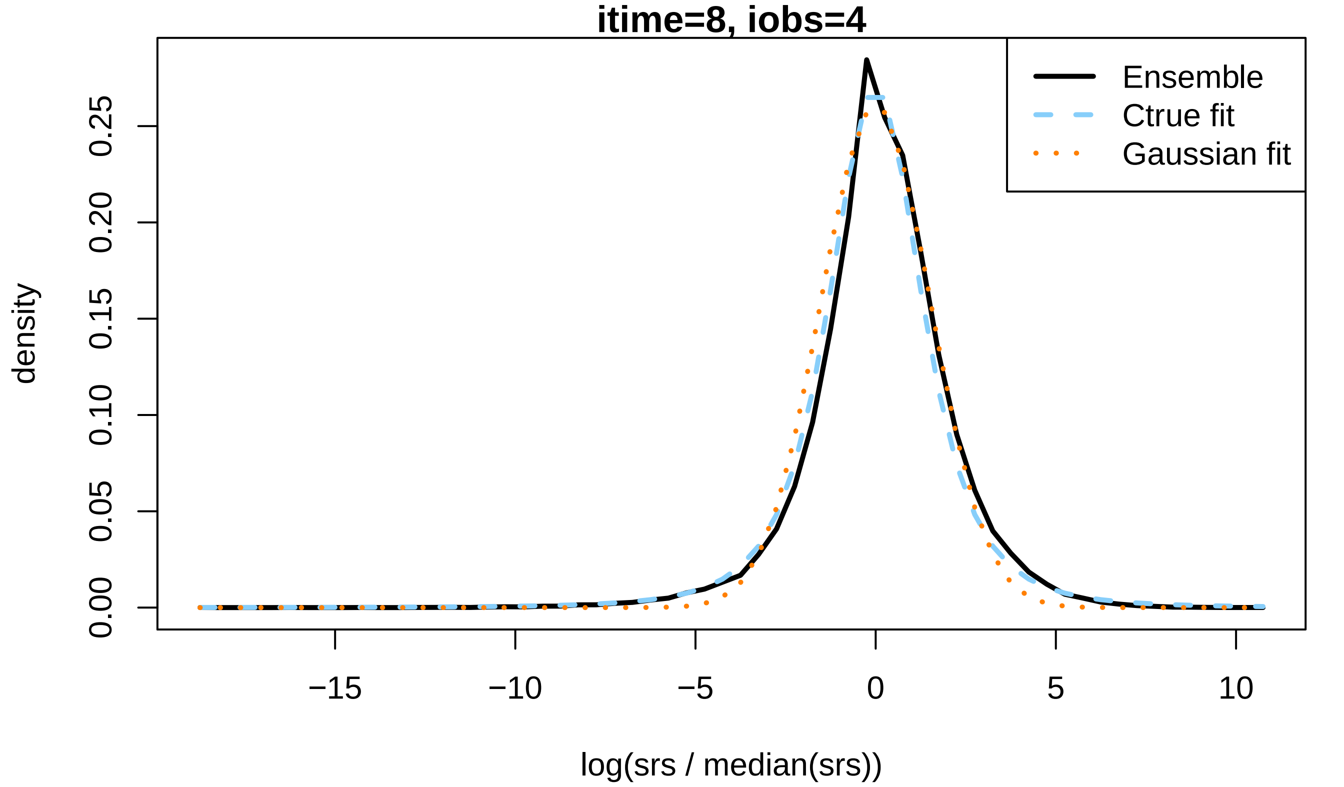

itime=8, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

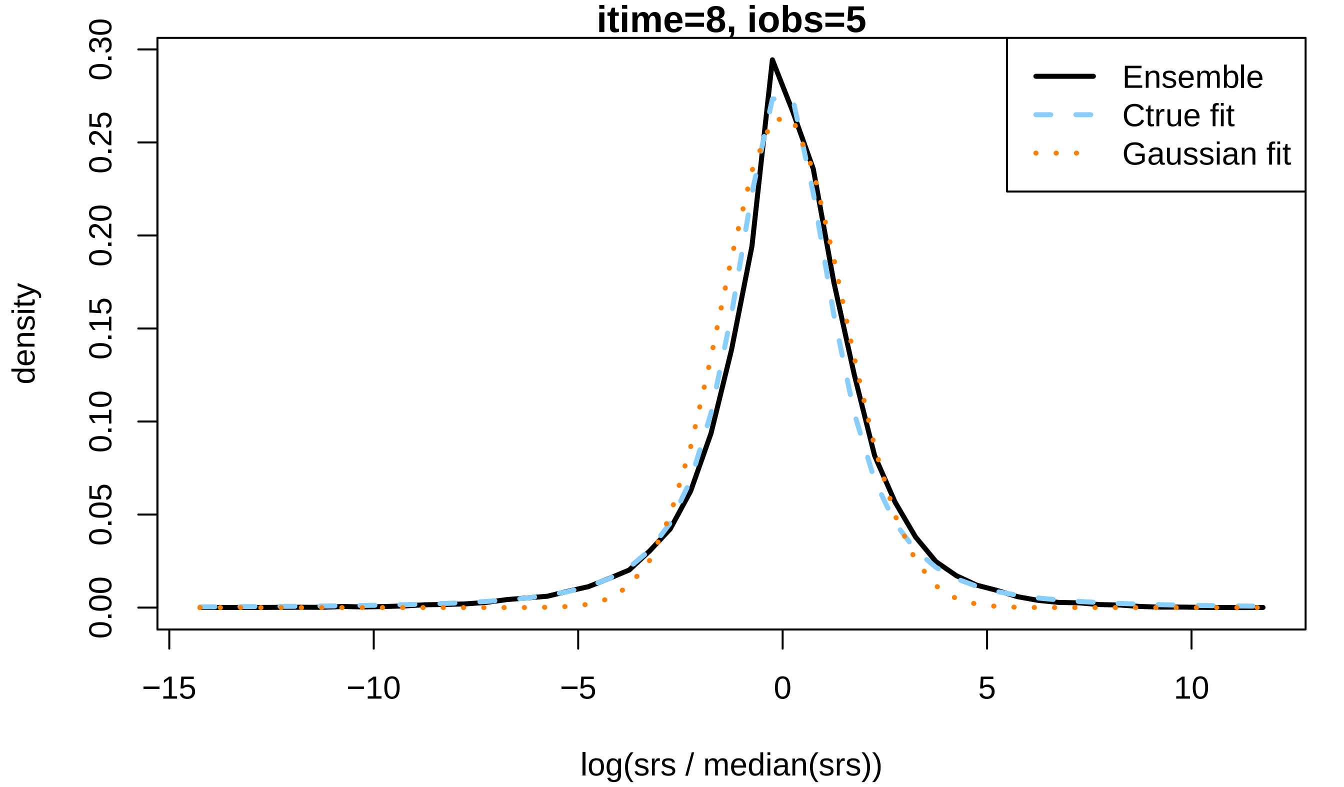


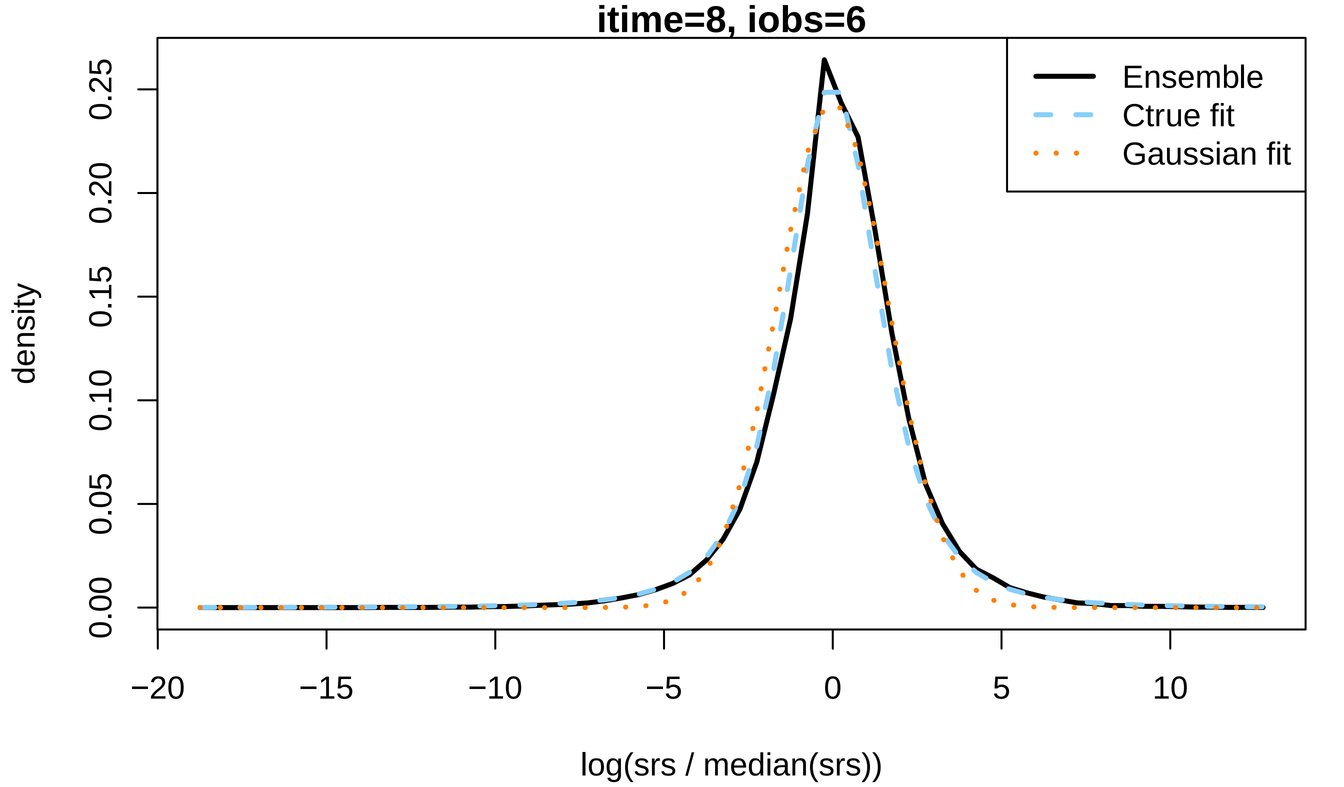
itime=8, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

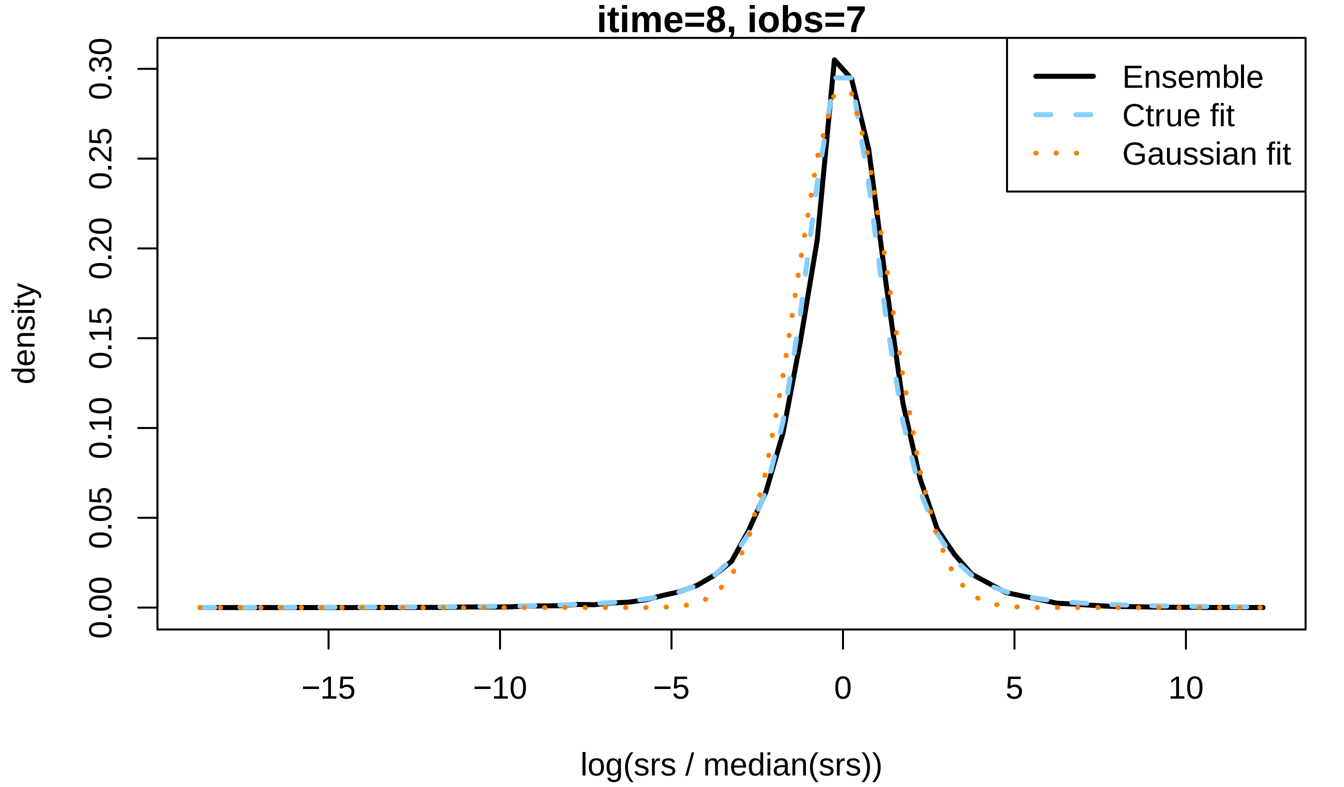


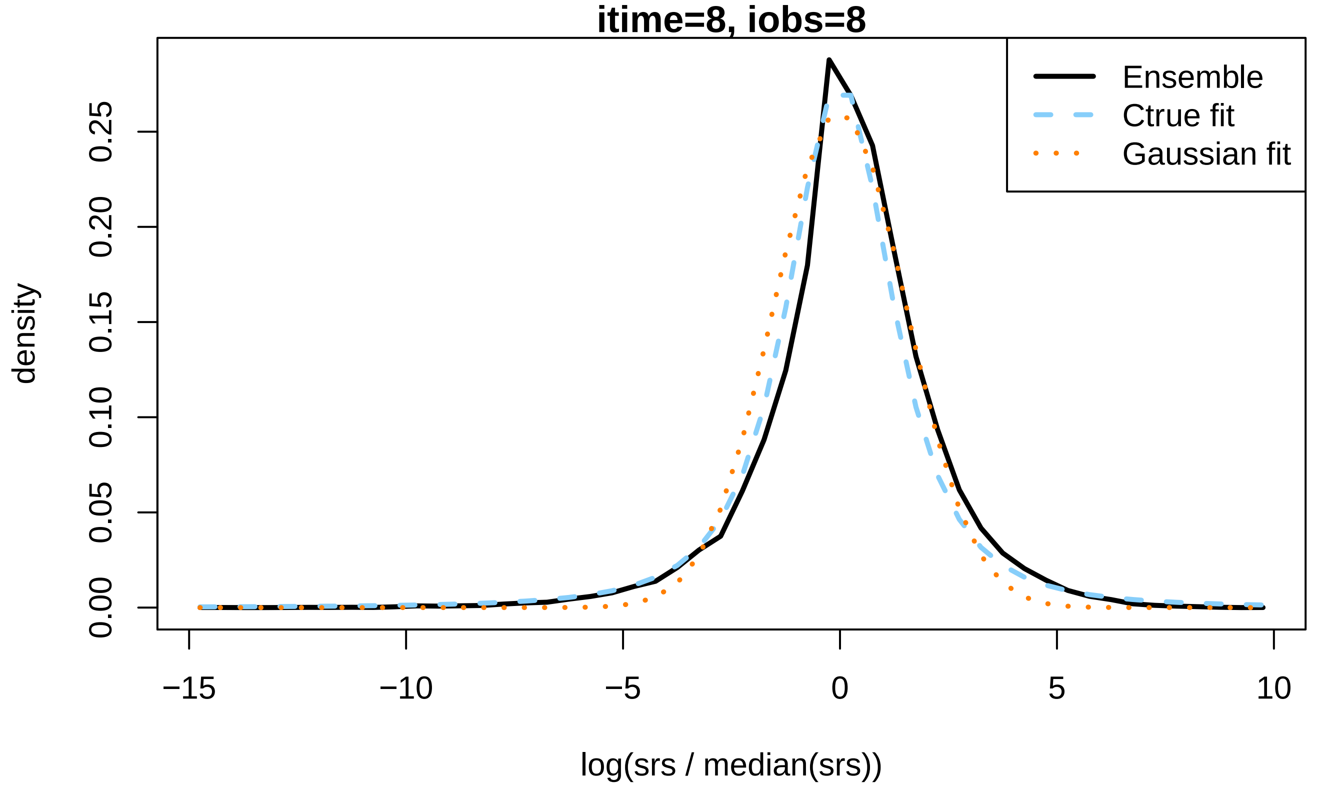
itime=8, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=8, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

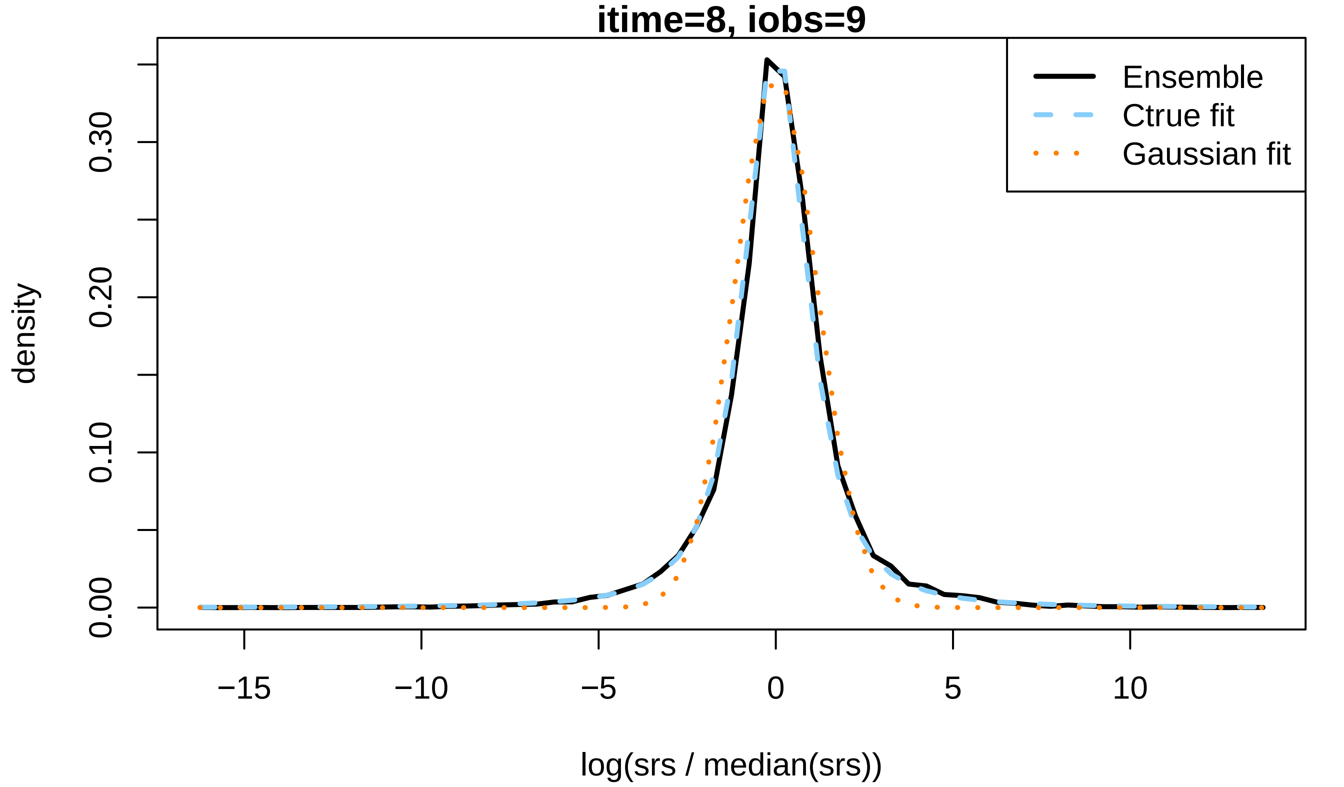

itime=8, iobs=5
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=8, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−20
−15
−10
−5
0
5
10
log(srs / median(srs))

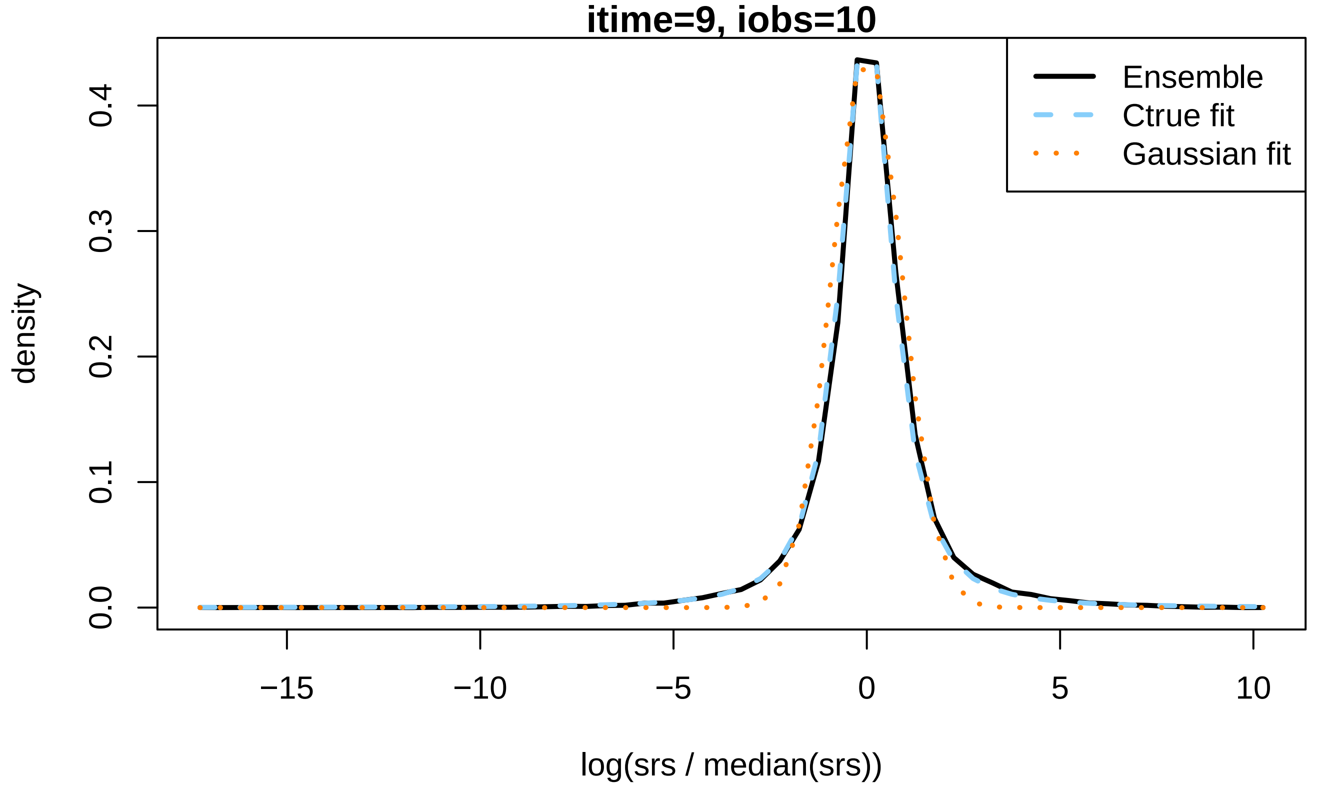

itime=8, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

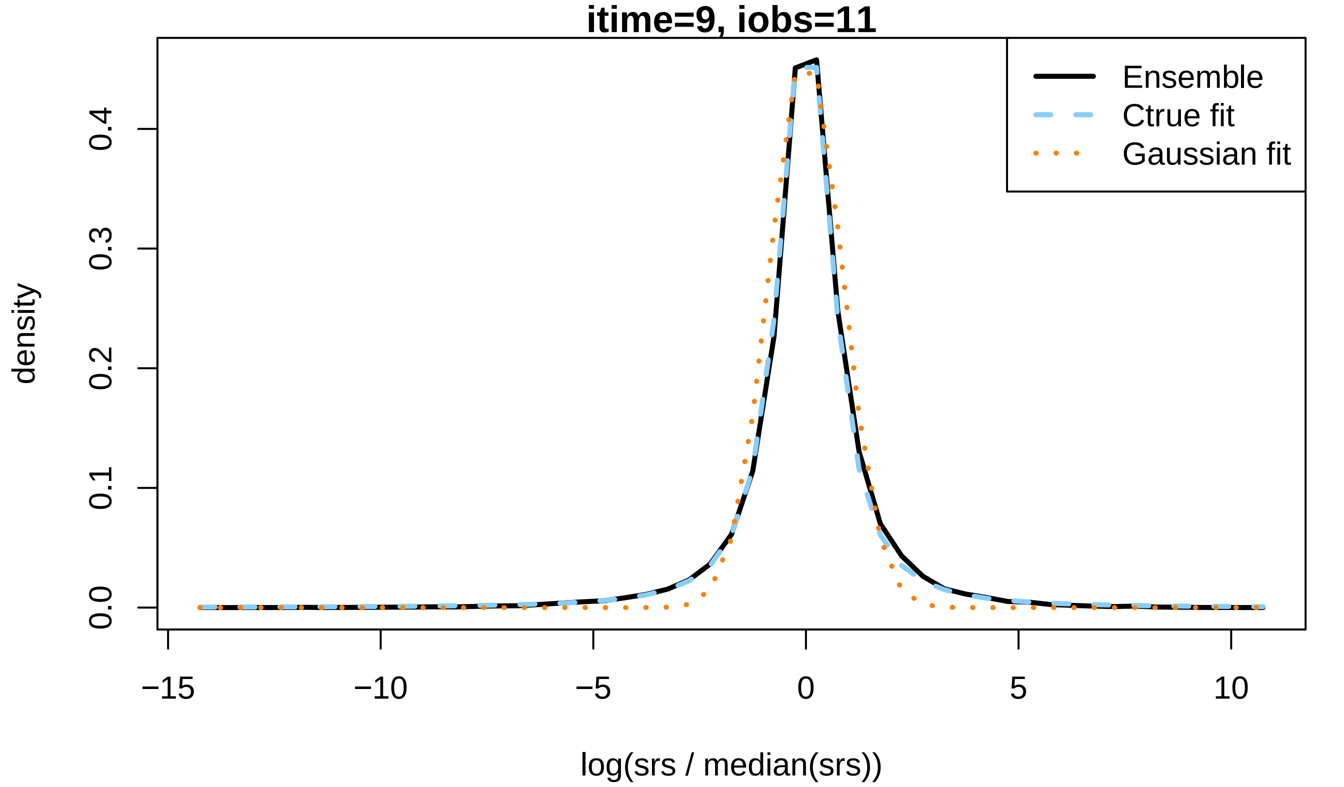


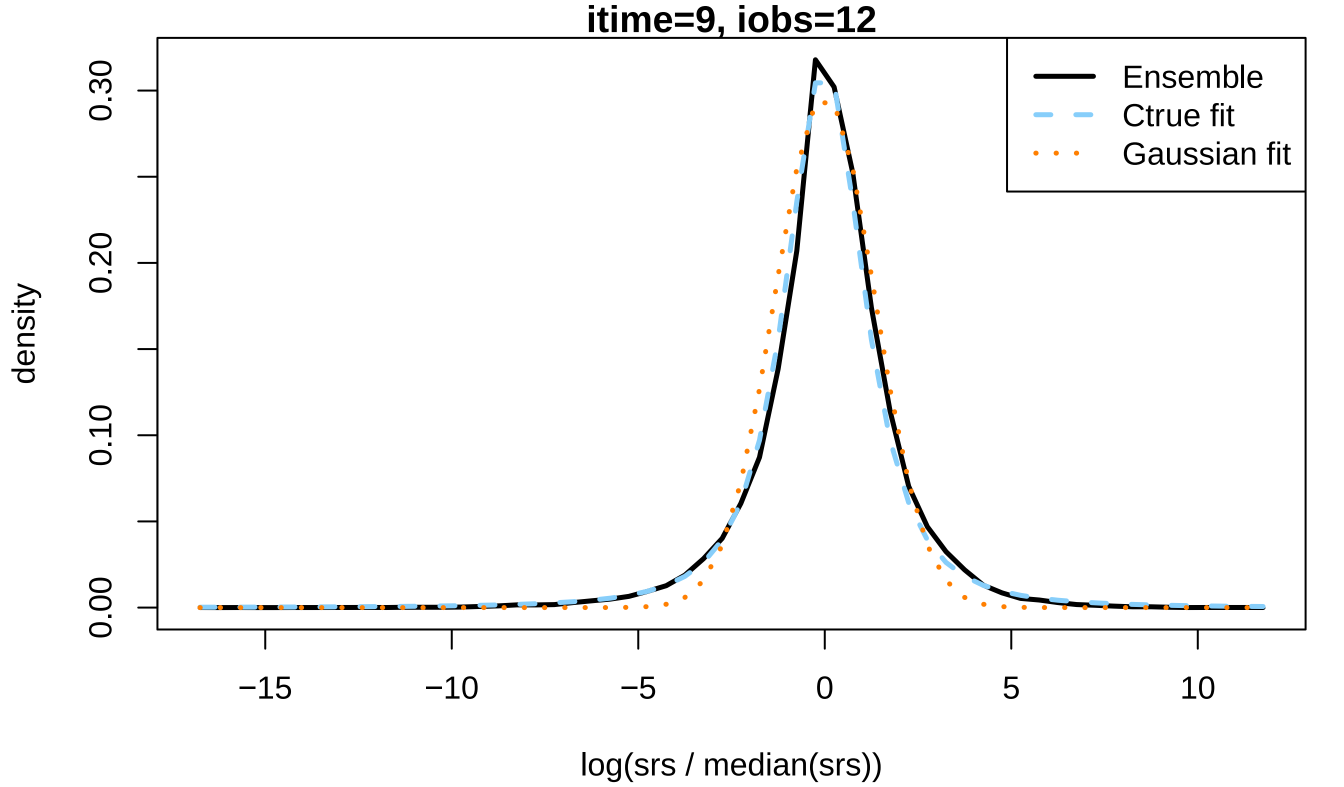
itime=8, iobs=8
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))



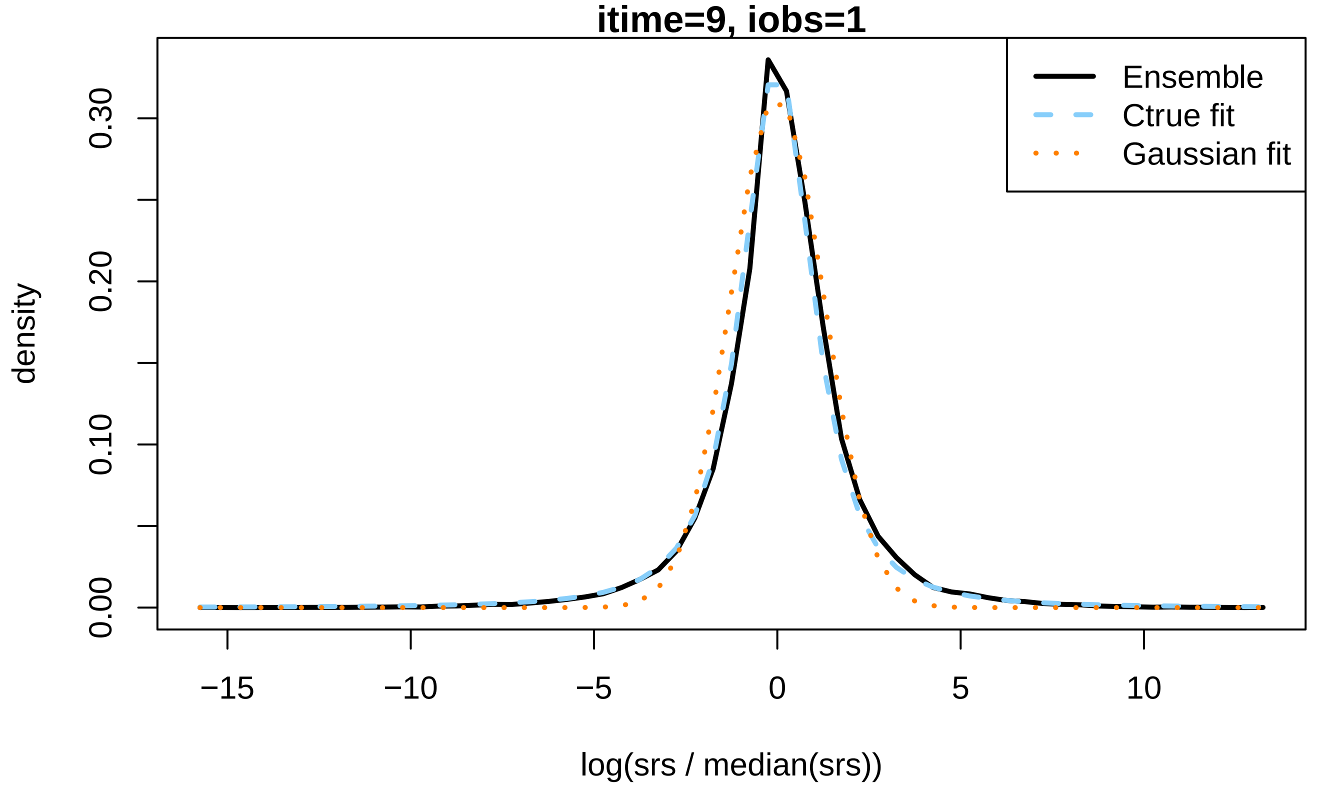
itime=8, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=9, iobs=10
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))

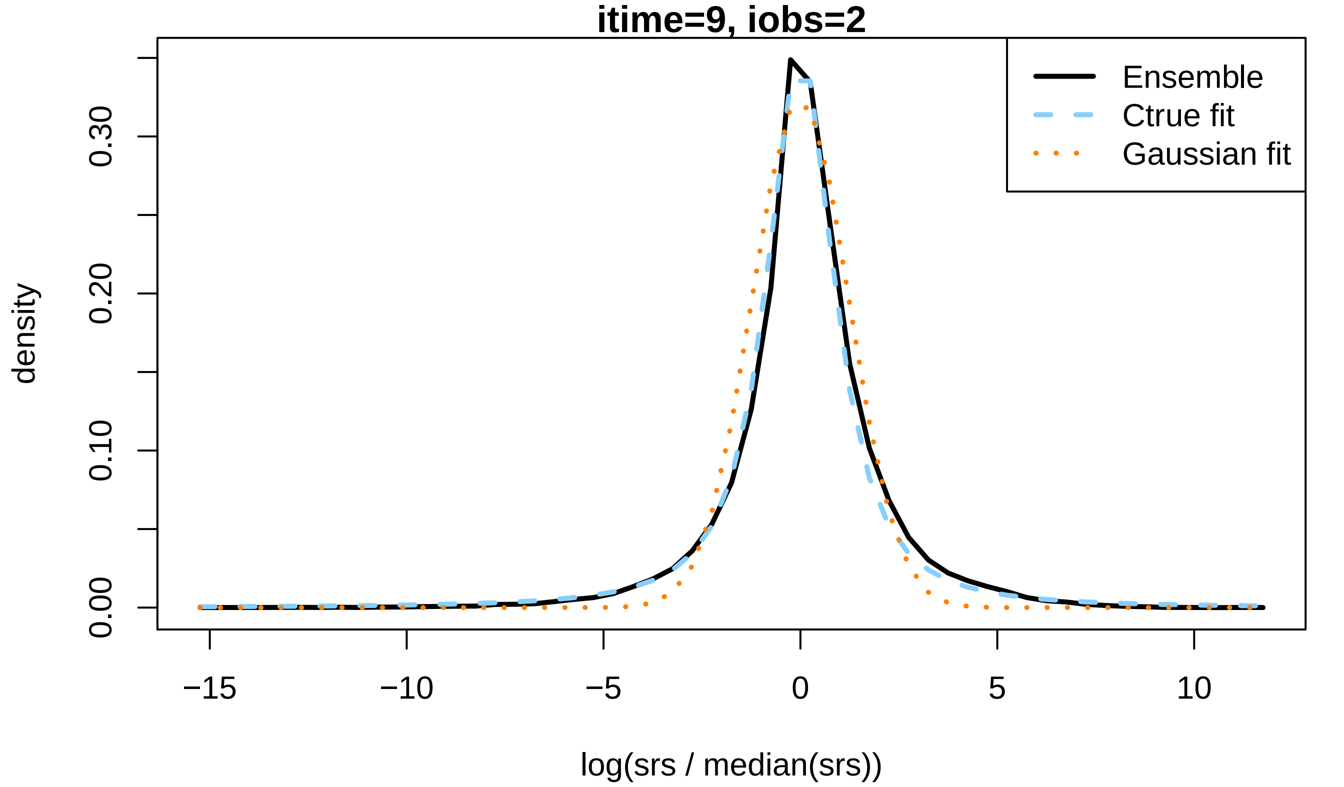

itime=9, iobs=11
Ensemble
Ctrue fit
Gaussian fit
density
0.0
0.1
0.2
0.3
0.4
−15
−10
−5
0
5
10
log(srs / median(srs))

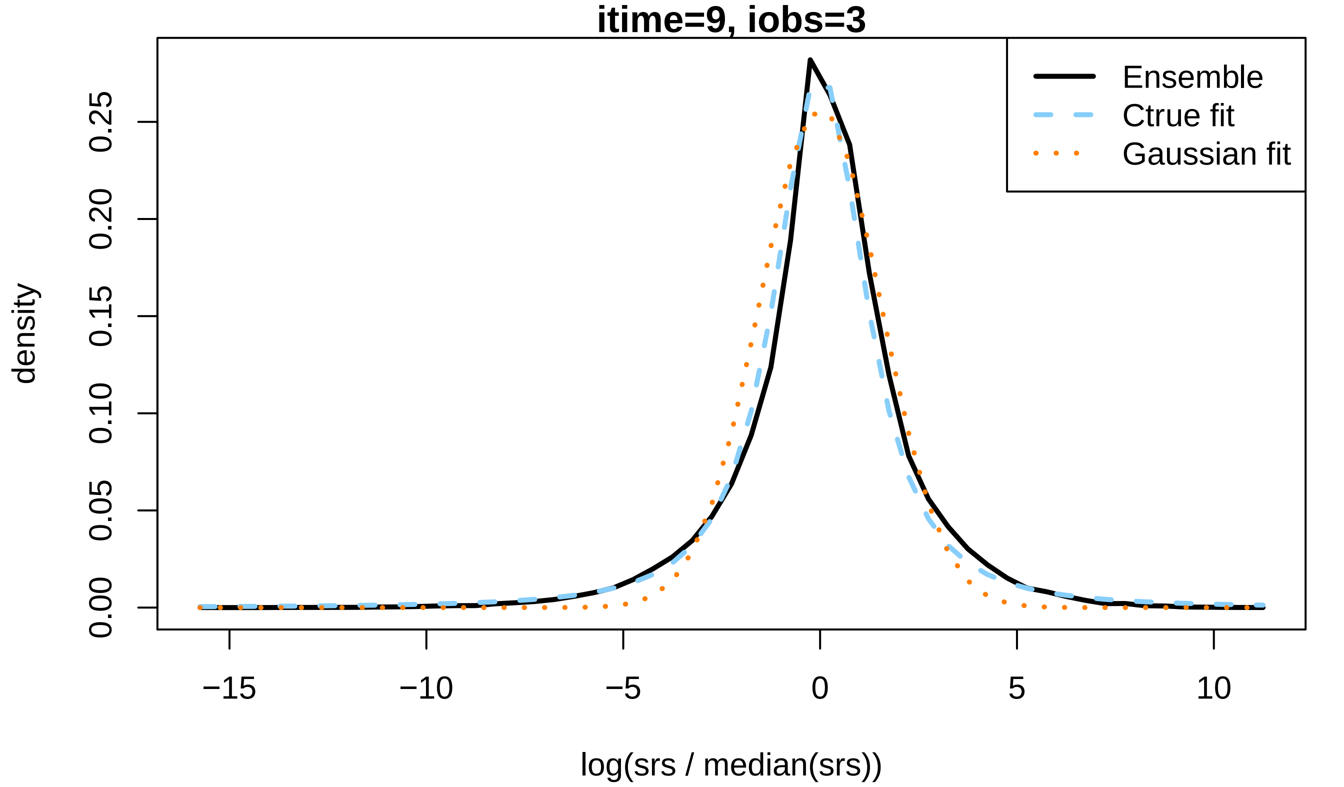


itime=9, iobs=12
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

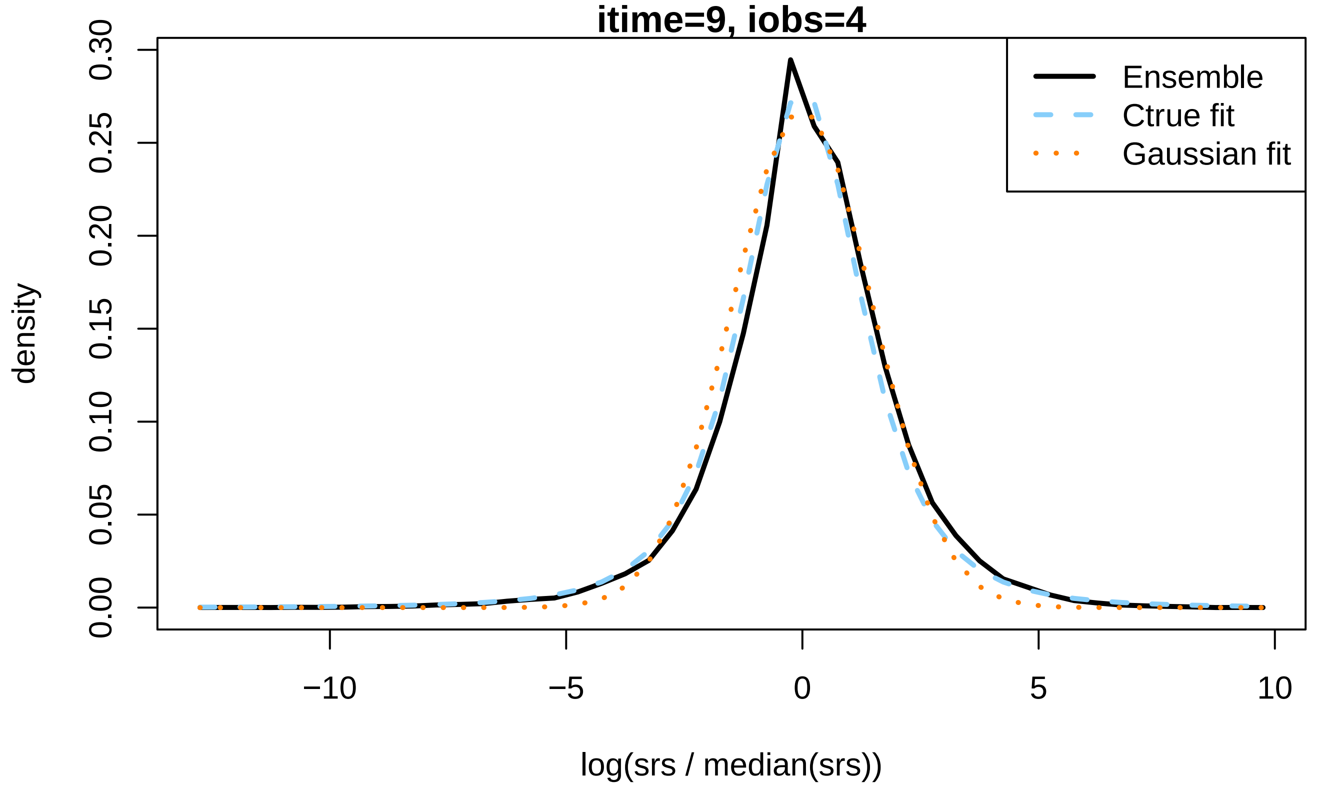


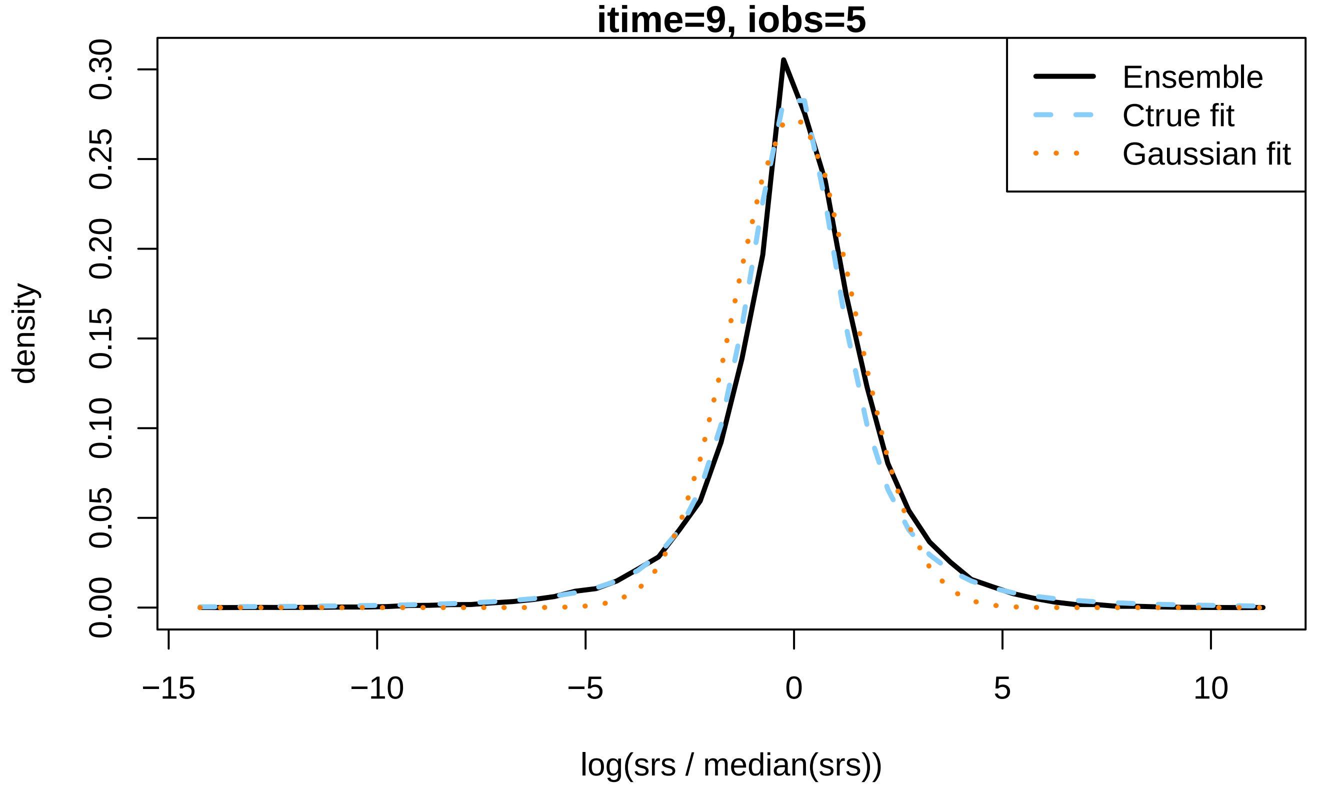
itime=9, iobs=1
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=9, iobs=2
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))

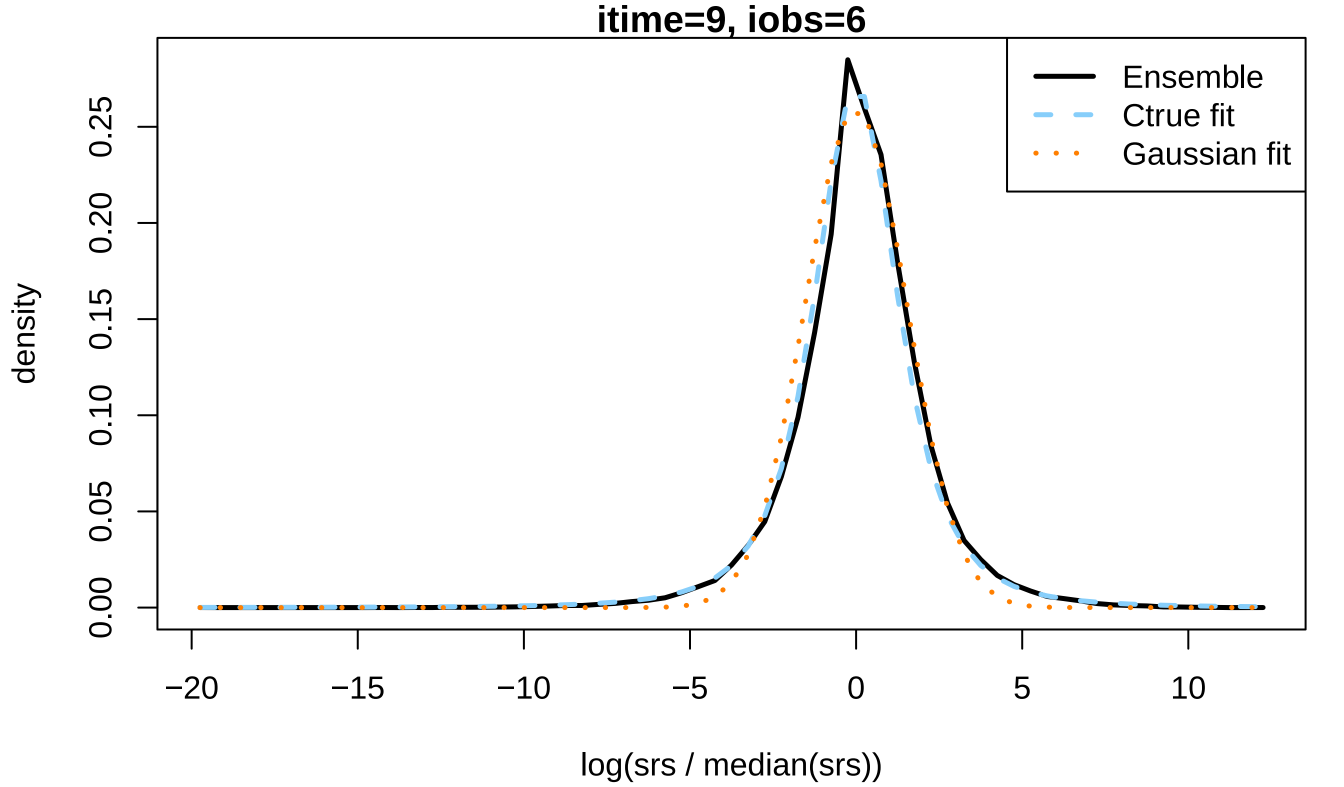


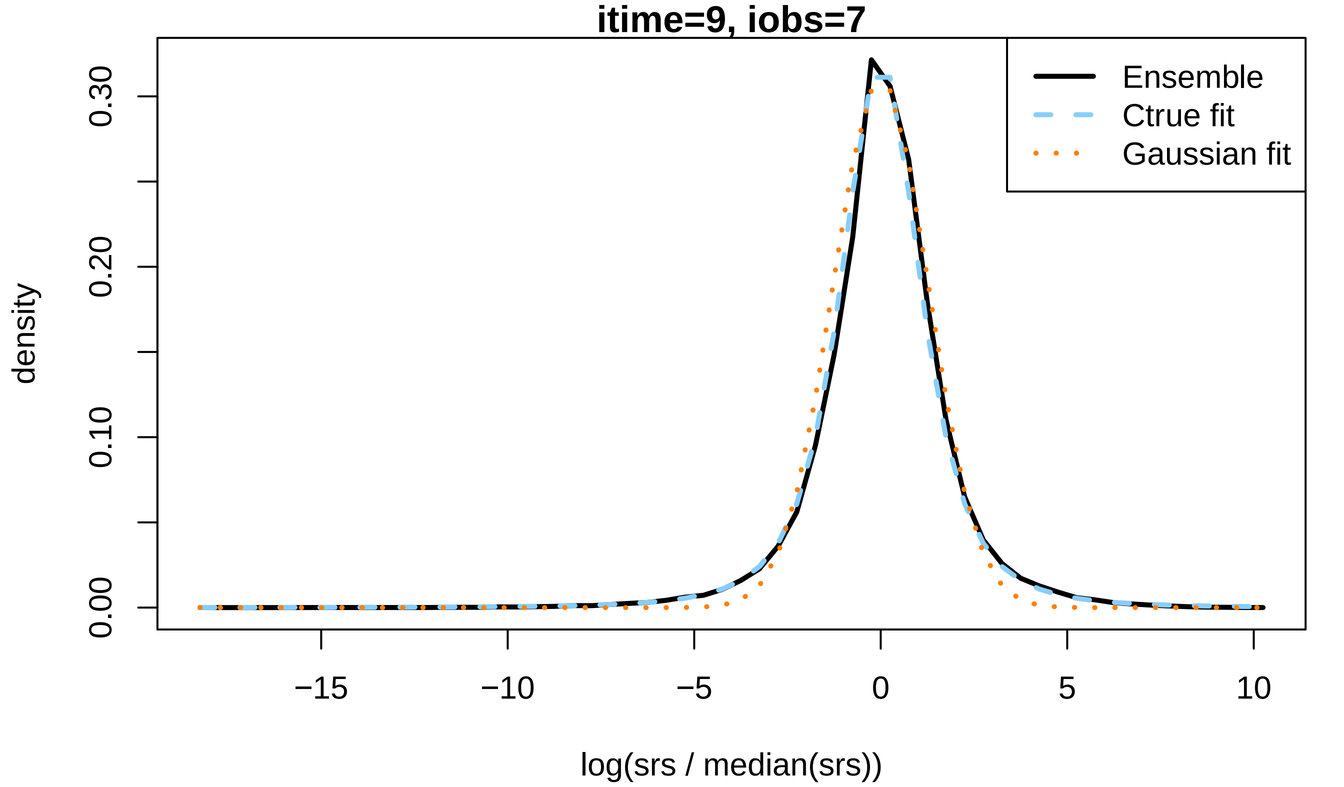
itime=9, iobs=3
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−15
−10
−5
0
5
10
log(srs / median(srs))

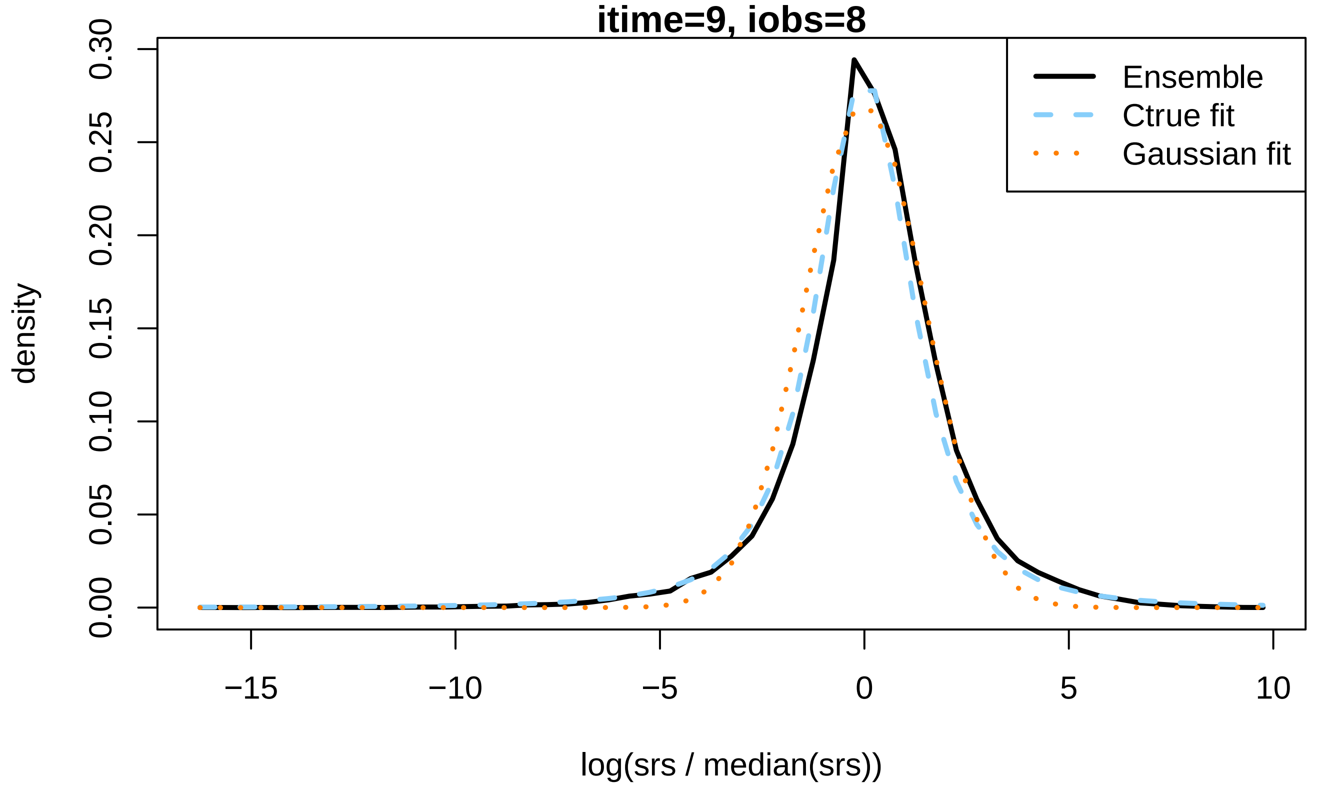

itime=9, iobs=4
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−10
−5
0
5
10
log(srs / median(srs))

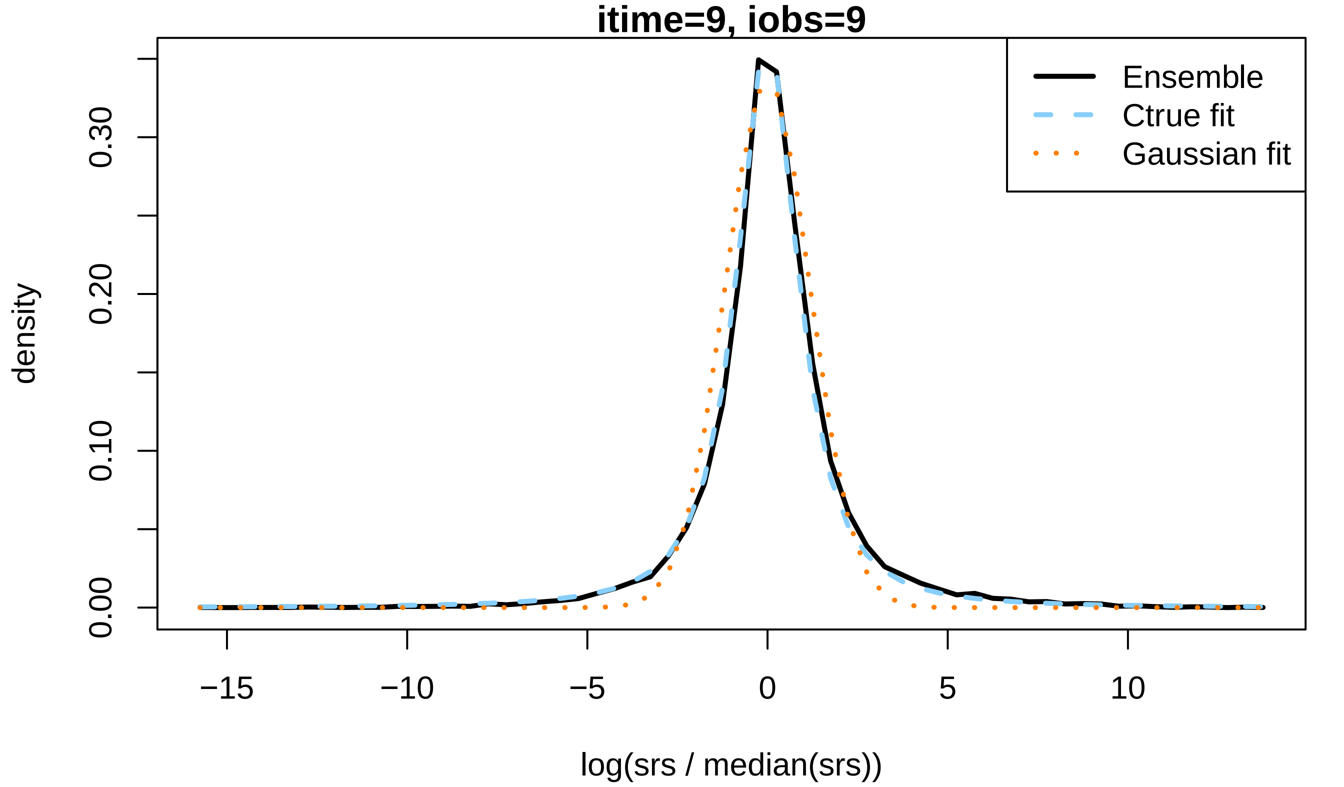


itime=9, iobs=5
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit


itime=9, iobs=6
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.05
0.10
0.15
0.20
0.25
−20
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=9, iobs=7
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))


itime=9, iobs=8
density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))
Ensemble
Ctrue fit
Gaussian fit



itime=9, iobs=9
Ensemble
Ctrue fit
Gaussian fit
density
0.00
0.10
0.20
0.30
−15
−10
−5
0
5
10
log(srs / median(srs))