Model documentation and write-up

1. **Who are you (mini-bio) and what do you do professionally? If you are on a team, please complete this block for each member of the team.**

   I am a data scientist with 3 years of commercial experience in sales, pharmacovigilance and insurance industries. I specialize in data processing, feature engineering and gradient boosting algorithms. I am currently looking for a job where I could work on machine learning problems more frequently.

2. **What motivated you to compete in this challenge?**

   I wanted to participate in a data science competition where I could use my analytical and machine learning skills. Participating in a competition that tries to tackle real world problems was another motivation.

3. **High level summary of your approach: what did you do and why?**

   I used LightGBM models as main predictors for each quantile and historical data distribution estimates for 0.1 and 0.9 quantile corrections. Best distribution was chosen from many different distributions for each reservoir in order to find optimal data representation and quantile values. A LightGBM prediction of quantile 0.5 was used as a center for the fitted distribution, based on which other quantiles were calculated.

   The LightGBM models were created separately for each month to enable the use of better features when more information is available with time.

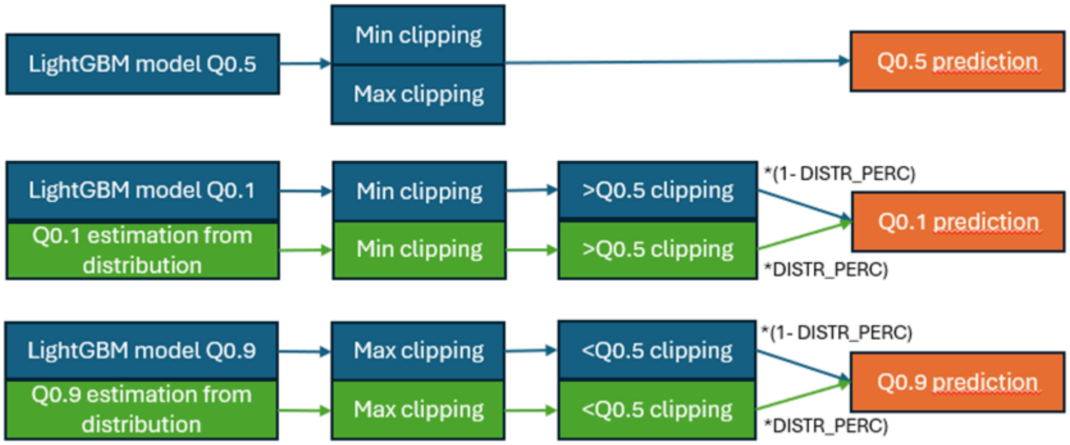
   For the last three months (May-July), already known naturalized flow from past months was used and forecasts were made to predict the remaining amount until July.

   Only features that significantly improve model performance were used, no more than ten features per month, as the training data is quite small.

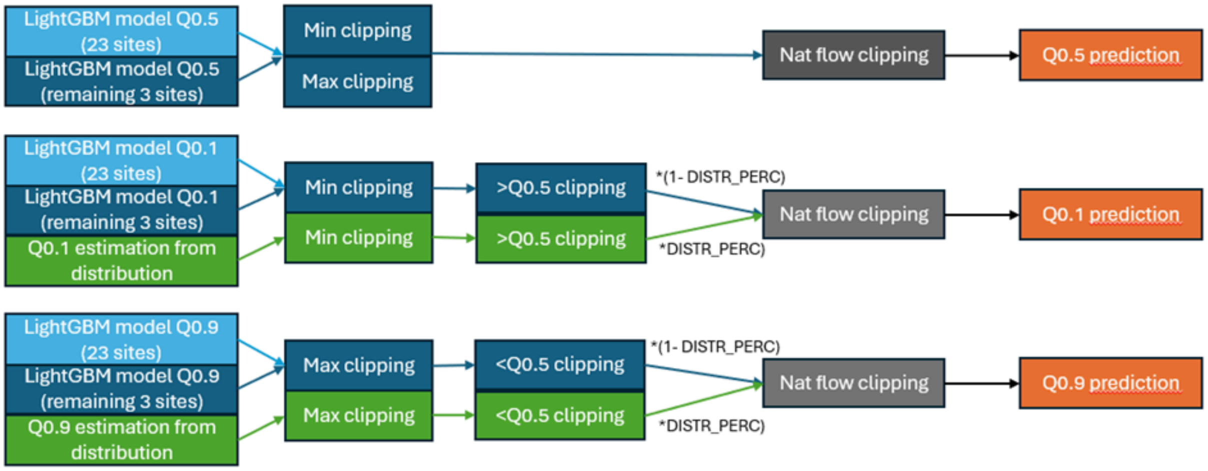
   The solution includes clipping methods to make forecasts only within the historical data value range for a given reservoir and to make forecast adjustments where a lower quantile has greater value than a higher quantile. It also addresses cases where already known naturalized flow from the previous month is at a similar level as the total forecast for the considered period.

4. Do you have any useful charts, graphs, or visualizations from the process?

- Single block of predictions for the first 4 months (January-April)



- Single block of predictions for the last 3 months (May-July)



- LOOCV prediction errors for different years with total water flow volume and precipitation values

Year	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
MQL Result	74	89	89	66	79	76	79	166	127	65
Interval coverage	0.84	0.82	0.80	0.87	0.87	0.87	0.90	0.54	0.69	0.85
Volume	16579	22537	28335	18036	24405	23015	21715	37294	24813	18908
SNOTEL Precipitation	831	890	960	867	957	923	908	1201	903	858
Year	**2014**	**2015**	**2016**	**2017**	**2018**	**2019**	**2020**	**2021**	**2022**	**2023**
MQL Result	69	107	59	107	76	92	67	72	80	87
Interval coverage	0.87	0.53	0.93	0.69	0.81	0.69	0.90	0.69	0.83	0.80
Volume	24262	15880	20578	34293	25875	25429	22083	15547	21005	26966
SNOTEL Precipitation	853	826	969	1176	869	939	816	764	937	950

5. **Copy and paste the 3 most impactful parts of your code and explain what each does and how it helped your model.**

- Fitting different distributions for each reservoir. Thanks to that, the optimal distribution for each reservoir could have been selected, from which 0.1 and 0.9 distribution estimate quantiles were derived that have a smoothing effect to the LightGBM predictions. LightGBM models alone struggled with those quantiles, especially in the early months.

```
distr_results = []
for site_id in tqdm(site_ids_unique):
    print(site_id)
    for distr_name, distr in all_distr_dict.items():
        args = distr._fitstart(df[df.site_id == site_id].volume)
        x0, func, restore, args = distr._reduce_func(args, {})
        try:
            res = minimize(func, x0, args=(df[df.site_id ==
site_id].volume,),
                           method='Nelder-Mead')
            distr_results.append([site_id, distr_name, res.fun,
tuple(res.x)])
        except:
            logger.info(f'There was an error with {distr_name} for
{site_id}. Ignoring the distribution and trying the next one.')
```

- Assignment of different weights for distribution estimates and LightGBM models for each month (distribution estimate weight is shown, whereas the remaining amount up to 1 corresponds to LightGBM models). Thanks to that, distribution estimates for 0.1 and 0.9 quantiles for earlier months have greater importance compared to later months when more information is available and LightGBM models work better.

```
distr_perc_dict = {1: 0.6,
                   2: 0.5,
                   3: 0.45,
                   4: 0.3,
                   5: 0.25,
                   6: 0.15,
                   7: 0.05}
```

- A separate function for forecasting residuals of results instead of whole results. The function was used for May-July predictions for 23 reservoirs with known naturalized flow since April. Thanks to that, the models only had to forecast remaining amount of the naturalized flow. Then, the already known flow was appended to the residuals.

```
lgbm_cv_residuals(...)
```

6. **Please provide the machine specs and time you used to run your model.**

    - CPU (model): Intel Core i9-13900KF
    - GPU (model or N/A): NVIDIA GeForce RTX 3090 Ti (not used in model training)
    - Memory (GB): 64
    - OS: Windows 11 Pro
    - Train duration: 55h (including hyperparameter optimization; more precise running time was provided in the final report)
    - Inference duration: 37m

7. **Anything we should watch out for or be aware of in using your model (e.g. code quirks, memory requirements, numerical stability issues, etc.)?**

    The solution seems to be stable.

8. **Did you use any tools for data preparation or exploratory data analysis that aren't listed in your code submission?**

    Data visualization using Jupyter Notebook, matplotlib and seaborn libraries was performed to examine correlation between different features and the historical water flow, taking into account values from different months, for which forecasts were made.

9. **How did you evaluate performance of the model other than the provided metric, if at all?**

    Models were evaluated using mostly the provided metric, taking interval coverage also into account. Data visualization was another crucial factor, to only use the features that visibly influence water supply.

10. **What are some other things you tried that didn't necessarily make it into the final workflow (quick overview)?**

- Extracting and appending data from other datasets that weren't good enough to include them in the final solution
- Creating two separate datasets and models – one containing features with a long-term history of values and the other one with features available only since ~2000. It was more of an idea, abandoned due to the decision to use only one LightGBM model for given date-reservoir-quantile combination

11. **If you were to continue working on this problem for the next year, what methods or techniques might you try in order to build on your work so far? Are there other fields or features you felt would have been very helpful to have?**

- Trying oversampling techniques
- Examining more datasets
- Finding features to improve stability of results over different years
- Concentrating more on middle months (March, April, May) to create better features for them

12. **What simplifications could be made to run your solution faster without sacrificing significant accuracy?**

- Using only more recent data
- Calculating quantiles based just on historical data from the given reservoir instead of fitting best distribution for each reservoir