Probabilistic Load Forecasting using Ensemble Weather Forecast

Yizhou Qian, Jacques De Chalendar, Mykel Kochenderfer, Eric Darve

Motivation

- Accurate prediction of building energy load is crucial for control optimization and management of energy resources
- Numerical Weather Predictions can provide probabilistic forecasting of weather variables related to energy demand, but requires some calibration and statistical-postprocessing

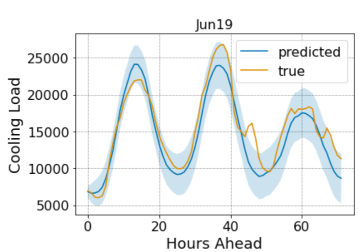
Challenges

- Cooling water consumption are usually underpredicted during heat waves (rare and extreme events)
- Limited amount of data are available for days with unusually high temperature

Linear Regression with Autoregressive Errors

We can predict the logarithm of cooling load using weather forecast and historical data:

$$\log(c_t) = \beta t + \omega_t + \epsilon_t, \omega_t = \alpha_1 \omega_{t-1} + \alpha_2 \omega_{t-2}$$



Gaussian Process Regression

Denote x as our weather forecast variable and $f(x)$ as our true observation of temperature, then we construct our model using the following covariance matrix:

$$Cov(f(x_1), f(x_2)) = C(x_1, x_2)$$

Then given training data, $\boldsymbol{y_{train}} = (f(x_1), \dots, f(x_n)), \boldsymbol{x_{train}} = (x_1, \dots, x_n)$, to predict temperature $f(x)$ with weather forecast variable, we have

$$\begin{pmatrix} f(x) \\ \boldsymbol{y_{train}} \end{pmatrix} \sim N(\mu, \Sigma),$$

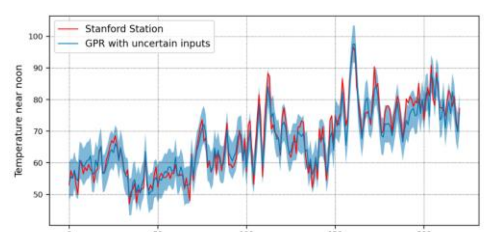
$$\Sigma = \begin{bmatrix} C(x,x) & C(x, x_{train}) \\ C(\boldsymbol{x_{train}}, x) & C(x_{train}, \boldsymbol{x_{train}}) \end{bmatrix}, \mu = \begin{pmatrix} \mu_x \\ \mu_{train} \end{pmatrix}$$

Hence,

$$E(f(x)|\boldsymbol{y_{train}}) = \mu_x - C(x, x_{train}) C(x_{train}, \boldsymbol{x_{train}})^{-1} (\boldsymbol{y_{train}} - \boldsymbol{\mu_{train}})$$

$$Var(f(x)|\boldsymbol{y_{train}}) = C(x,x) - C(x, x_{train}) C(x_{train}, \boldsymbol{x_{train}})^{-1} C(\boldsymbol{x_{train}}, x)$$

Probabilistic Prediction of Temperature



Ensemble Model Output Statistics

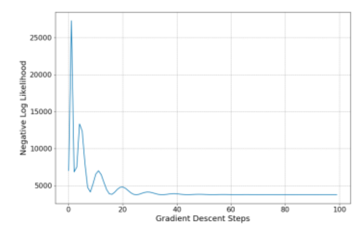
We view our true temperature T_i as realizations from a parametrized normal distribution:

$$T_i \sim N(\alpha + \beta m_i, \gamma + \delta s_i)$$

Then we can find optimal values of parameters α, β, γ and δ by minimizing the negative log-likelihood:

$$L(\alpha, \beta, \gamma, \delta) = \frac{1}{2} \sum_i \frac{(T_i - \alpha - \beta m_i)^2}{(\gamma + \delta s_i)^2} + \sum_i \log(\gamma + \delta s_i)$$

We can optimize by using automatic differentiation and custom machine learning optimizers:



Calibrated Normal distribution