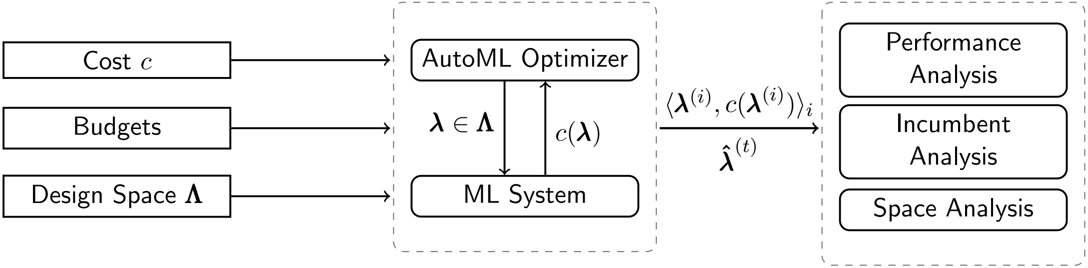
AutoML: Interpretability

Incumbent Analysis and Local Hyperparameter Importance

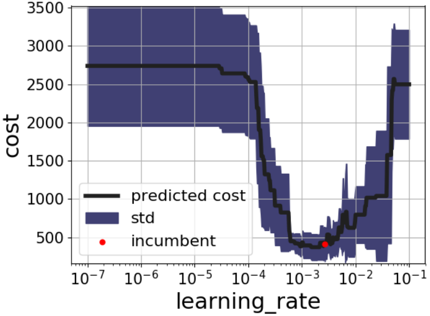
Bernd Bischl Frank Hutter Lars Kotthoff
Marius Lindauer Joaquin Vanschoren

Idea



$\rightsquigarrow$ focus on why is the eventually returned configuration a good choice

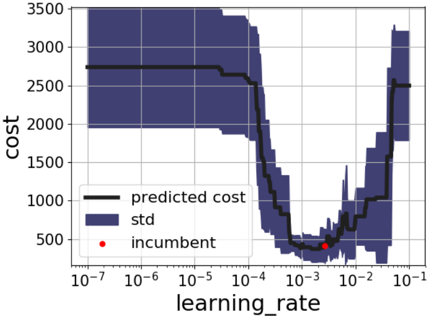
Local Importance [Biedenkapp et al. 2018]



Source: [Lindauer et al. 2019]

- Typical question of users:
  - How would the performance change if we change hyperparameter $\boldsymbol{\lambda}_i$?

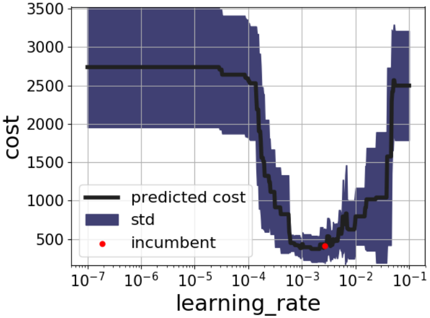
Local Importance [Biedenkapp et al. 2018]



Source: [Lindauer et al. 2019]

- Typical question of users:
  - How would the performance change if we change hyperparameter $\boldsymbol{\lambda}_i$?
- Problem: Running full study is often too expensive
  - Each run of an ML-system is potential expensive

Local Importance [Biedenkapp et al. 2018]



Source: [Lindauer et al. 2019]

- Typical question of users:
  - How would the performance change if we change hyperparameter $\boldsymbol{\lambda}_i$?
- Problem: Running full study is often too expensive
  - Each run of an ML-system is potential expensive
- Key Ideas:
  - Re-use probabilistic models as trained in BO
  - Plot performance change around $\hat{\boldsymbol{\lambda}}^{(t)}$ along each dimension

Quantifying Local Importance [Biedenkapp et al. 2018]

$$\mathsf{VAR}_{\boldsymbol{\lambda}}(i) = \sum_{v \in \boldsymbol{\Lambda}_i} \left(\mathbb{E}_{v \sim \boldsymbol{\Lambda}_i}[L(\boldsymbol{\lambda})] - L(\boldsymbol{\lambda}[\boldsymbol{\lambda}_i := v])\right)^2 \tag{1}$$

Quantifying Local Importance [Biedenkapp et al. 2018]

$$\mathsf{VAR}_{\boldsymbol{\lambda}}(i) = \sum_{v \in \boldsymbol{\Lambda}_i} \left(\mathbb{E}_{v \sim \boldsymbol{\Lambda}_i}[L(\boldsymbol{\lambda})] - L(\boldsymbol{\lambda}[\boldsymbol{\lambda}_i := v])\right)^2 \tag{1}$$

$$\mathsf{LPI}(i \mid \boldsymbol{\lambda}) = \frac{VAR_{\boldsymbol{\lambda}}(i)}{\sum_j VAR_{\boldsymbol{\lambda}}(j)} \tag{2}$$

Quantifying Local Importance [Biedenkapp et al. 2018]

$$\text{VAR}_{\boldsymbol{\lambda}}(i) = \sum_{v \in \boldsymbol{\Lambda}_i} \left(\mathbb{E}_{v \sim \boldsymbol{\Lambda}_i}[L(\boldsymbol{\lambda})] - L(\boldsymbol{\lambda}[\boldsymbol{\lambda}_i := v])\right)^2 \tag{1}$$

$$\text{LPI}(i \mid \boldsymbol{\lambda}) = \frac{VAR_{\boldsymbol{\lambda}}(i)}{\sum_j VAR_{\boldsymbol{\lambda}}(j)} \tag{2}$$

$\rightsquigarrow$ While fixing all other hyperparameters to the incumbent value, the hyperparameter with the highest variance is the most important one

Ablation Study for Importance

- Users often start from some kind of default configuration
  1. As given in the documentation
  2. Or as always used in the last time

Ablation Study for Importance

- Users often start from some kind of default configuration
  1. As given in the documentation
  2. Or as always used in the last time
- Key Idea: Going from the default to the automatically optimized configuration, which choices where important?

$$
\begin{aligned}
\boldsymbol{\lambda}^{(\mathsf{start})} &= [1, 1, 0, 100] \\
\boldsymbol{\lambda}^{(\mathsf{end})} &= [0.98, 2.42, 1, 42]
\end{aligned}
$$

Ablation Study for Importance

- Users often start from some kind of default configuration
  1. As given in the documentation
  2. Or as always used in the last time
- Key Idea: Going from the default to the automatically optimized configuration, which choices where important?

$$\begin{aligned}\boldsymbol{\lambda}^{(\mathrm{start})} &= [1, 1, 0, 100] \\ \boldsymbol{\lambda}^{(\mathrm{end})} &= [0.98, 2.42, 1, 42]\end{aligned}$$

- Cheap approach: Assess $\boldsymbol{\lambda}^{(\mathrm{end})}$ with each hyperparameter value from $\boldsymbol{\lambda}^{(\mathrm{start})}$

Ablation Study for Importance

- Users often start from some kind of default configuration
  1. As given in the documentation
  2. Or as always used in the last time
- Key Idea: Going from the default to the automatically optimized configuration, which choices where important?

$$\boldsymbol{\lambda}^{(\mathsf{start})} = [1, 1, 0, 100]$$
$$\boldsymbol{\lambda}^{(\mathsf{end})} = [0.98, 2.42, 1, 42]$$

- Cheap approach: Assess $\boldsymbol{\lambda}^{(\mathsf{end})}$ with each hyperparameter value from $\boldsymbol{\lambda}^{(\mathsf{start})}$
- Expensive approach: Try all mixtures of $\boldsymbol{\lambda}^{(\mathsf{end})}$ and $\boldsymbol{\lambda}^{(\mathsf{start})}$
  - Only feasible for small spaces and fairly cheap ML systems

Ablation Study for Importance

- Users often start from some kind of default configuration
  1. As given in the documentation
  2. Or as always used in the last time
- Key Idea: Going from the default to the automatically optimized configuration, which choices where important?

$$
\begin{aligned}
\boldsymbol{\lambda}^{(\text{start})} &= [1, 1, 0, 100] \\
\boldsymbol{\lambda}^{(\text{end})} &= [0.98, 2.42, 1, 42]
\end{aligned}
$$

- Cheap approach: Assess $\boldsymbol{\lambda}^{(\text{end})}$ with each hyperparameter value from $\boldsymbol{\lambda}^{(\text{start})}$
- Expensive approach: Try all mixtures of $\boldsymbol{\lambda}^{(\text{end})}$ and $\boldsymbol{\lambda}^{(\text{start})}$
  - Only feasible for small spaces and fairly cheap ML systems
- Trade-off: Find a way from $\boldsymbol{\lambda}^{(\text{start})}$ to $\boldsymbol{\lambda}^{(\text{end})}$ in a greedy fashion [Fawcett and Hoos. 2016]

Greedy Ablation Study

Given:

$$
\begin{aligned}
\boldsymbol{\lambda}^{(\mathsf{start})} &= [1, 1, 0, 100] & L_{\mathsf{start}} &= 20\% \\
\boldsymbol{\lambda}^{(\mathsf{end})} &= [0.98, 2.42, 1, 42] & L_{\mathsf{end}} &= 4\%
\end{aligned}
$$

Greedy Ablation Study

Given:

$$\begin{aligned} \boldsymbol{\lambda}^{(\mathsf{start})} &= [1, 1, 0, 100] \quad & L_{\mathsf{start}} = 20\% \\ \boldsymbol{\lambda}^{(\mathsf{end})} &= [0.98, 2.42, 1, 42] \quad & L_{\mathsf{end}} = 4\% \end{aligned}$$

1st Iteration:

$$\boldsymbol{\lambda}^{(1)} = [0.98, 1, 0, 100] \quad L_1 = 19\%$$

Greedy Ablation Study

Given:

$$\begin{aligned} \boldsymbol{\lambda}^{(\mathsf{start})} &= [1, 1, 0, 100] & L_{\mathsf{start}} &= 20\% \\ \boldsymbol{\lambda}^{(\mathsf{end})} &= [0.98, 2.42, 1, 42] & L_{\mathsf{end}} &= 4\% \end{aligned}$$

1st Iteration:

$$\begin{aligned} \boldsymbol{\lambda}^{(1)} &= [0.98, 1, 0, 100] & L_1 &= 19\% \\ \boldsymbol{\lambda}^{(2)} &= [1, 2.42, 0, 100] & L_2 &= 20\% \end{aligned}$$

Greedy Ablation Study

Given:

$$\begin{aligned}
\boldsymbol{\lambda}^{(\mathsf{start})} &= [1, 1, 0, 100] & L_{\mathsf{start}} &= 20\% \\
\boldsymbol{\lambda}^{(\mathsf{end})} &= [0.98, 2.42, 1, 42] & L_{\mathsf{end}} &= 4\%
\end{aligned}$$

1st Iteration:

$$\begin{aligned}
\boldsymbol{\lambda}^{(1)} &= [0.98, 1, 0, 100] & L_1 &= 19\% \\
\boldsymbol{\lambda}^{(2)} &= [1, 2.42, 0, 100] & L_2 &= 20\% \\
\boldsymbol{\lambda}^{(3)} &= [1, 1, 1, 100] & L_3 &= 7\%
\end{aligned}$$

Greedy Ablation Study

Given:

$$\begin{aligned} \boldsymbol{\lambda}^{(\mathsf{start})} &= [1, 1, 0, 100] & L_{\mathsf{start}} &= 20\% \\ \boldsymbol{\lambda}^{(\mathsf{end})} &= [0.98, 2.42, 1, 42] & L_{\mathsf{end}} &= 4\% \end{aligned}$$

1st Iteration:

$$\begin{aligned} \boldsymbol{\lambda}^{(1)} &= [0.98, 1, 0, 100] & L_1 &= 19\% \\ \boldsymbol{\lambda}^{(2)} &= [1, 2.42, 0, 100] & L_2 &= 20\% \\ \boldsymbol{\lambda}^{(3)} &= [1, 1, 1, 100] & L_3 &= 7\% \\ \boldsymbol{\lambda}^{(4)} &= [1, 1, 0, 42] & L_4 &= 16\% \end{aligned}$$

Greedy Ablation Study

Given:

$$\begin{aligned} \boldsymbol{\lambda}^{(\text{start})} &= [1, 1, 0, 100] & L_{\text{start}} &= 20\% \\ \boldsymbol{\lambda}^{(\text{end})} &= [0.98, 2.42, 1, 42] & L_{\text{end}} &= 4\% \end{aligned}$$

1st Iteration:

$$\begin{aligned} \boldsymbol{\lambda}^{(1)} &= [0.98, 1, 0, 100] & L_1 &= 19\% \\ \boldsymbol{\lambda}^{(2)} &= [1, 2.42, 0, 100] & L_2 &= 20\% \\ \boldsymbol{\lambda}^{(3)} &= [1, 1, 1, 100] & L_3 &= 7\% \\ \boldsymbol{\lambda}^{(4)} &= [1, 1, 0, 42] & L_4 &= 16\% \end{aligned}$$

$\rightsquigarrow$ 1st step: $\boldsymbol{\lambda}_2$ – flipping hyperparameter 3

Greedy Ablation Study

Given:

$$
\begin{aligned}
\boldsymbol{\lambda}^{(\text{start})} &= [1, 1, 0, 100] & L_{\text{start}} &= 20\% \\
\boldsymbol{\lambda}^{(s1)} &= [1, 1, 1, 100] & L &= 7\% \\
\boldsymbol{\lambda}^{(\text{end})} &= [0.98, 2.42, 1, 42] & L_{\text{end}} &= 4\%
\end{aligned}
$$

2nd Iteration:

$$\boldsymbol{\lambda}^{(1)} = [0.98, 1, 1, 100] \quad L_1 = 6\%$$

Greedy Ablation Study

Given:

$$\begin{aligned}
\boldsymbol{\lambda}^{(\mathsf{start})} &= [1, 1, 0, 100] & L_{\mathsf{start}} &= 20\% \\
\boldsymbol{\lambda}^{(s1)} &= [1, 1, 1, 100] & L &= 7\% \\
\boldsymbol{\lambda}^{(\mathsf{end})} &= [0.98, 2.42, 1, 42] & L_{\mathsf{end}} &= 4\%
\end{aligned}$$

2nd Iteration:

$$\begin{aligned}
\boldsymbol{\lambda}^{(1)} &= [0.98, 1, 1, 100] & L_1 &= 6\% \\
\boldsymbol{\lambda}^{(2)} &= [1, 2.42, 1, 100] & L_2 &= 7\%
\end{aligned}$$

Greedy Ablation Study

Given:

$$\begin{aligned}
\boldsymbol{\lambda}^{(\text{start})} &= [1, 1, 0, 100] & L_{\text{start}} &= 20\% \\
\boldsymbol{\lambda}^{(s1)} &= [1, 1, 1, 100] & L &= 7\% \\
\boldsymbol{\lambda}^{(\text{end})} &= [0.98, 2.42, 1, 42] & L_{\text{end}} &= 4\%
\end{aligned}$$

2nd Iteration:

$$\begin{aligned}
\boldsymbol{\lambda}^{(1)} &= [0.98, 1, 1, 100] & L_1 &= 6\% \\
\boldsymbol{\lambda}^{(2)} &= [1, 2.42, 1, 100] & L_2 &= 7\% \\
\boldsymbol{\lambda}^{(3)} &= [1, 1, 1, 42] & L_3 &= 5\%
\end{aligned}$$

$\rightsquigarrow$ 2nd step: $\boldsymbol{\lambda}_3$ – flipping hyperparameter 4

Greedy Ablation Study

Given:

$$\begin{aligned}
\boldsymbol{\lambda}^{(\mathsf{start})} &= [1, 1, 0, 100] & L_{\mathsf{start}} &= 20\% \\
\boldsymbol{\lambda}^{(s1)} &= [1, 1, 1, 100] & L &= 7\% \\
\boldsymbol{\lambda}^{(s2)} &= [1, 1, 1, 42] & L &= 5\% \\
\boldsymbol{\lambda}^{(\mathsf{end})} &= [0.98, 2.42, 1, 42] & L_{\mathsf{end}} &= 4\%
\end{aligned}$$

3rd Iteration:

$$\begin{aligned}
\boldsymbol{\lambda}^{(1)} &= [0.98, 1, 1, 100] & L_1 &= 4\% \\
\boldsymbol{\lambda}^{(2)} &= [1, 2.42, 1, 100] & L_2 &= 5\%
\end{aligned}$$

$\rightsquigarrow$ 2nd step: $\boldsymbol{\lambda}_3$ – flipping hyperparameter 1

Greedy Ablation Study

Ablation Path:

$$
\begin{array}{ll}
\boldsymbol{\lambda}^{(\mathsf{start})} = [1, 1, 0, 100] & L_{\mathsf{start}} = 20\% \\
\boldsymbol{\lambda}^{(s1)} = [1, 1, 1, 100] & L = 7\% \\
\boldsymbol{\lambda}^{(s1)} = [1, 1, 1, 42] & L = 5\% \\
\boldsymbol{\lambda}^{(s3)} = [0.98, 1, 1, 42] & L = 4\% \\
\boldsymbol{\lambda}^{(s4)} = [0.98, 2.42, 1, 42] & L = 4\% \\
\boldsymbol{\lambda}^{(\mathsf{end})} = [0.98, 2.42, 1, 42] & L_{\mathsf{end}} = 4\%
\end{array}
$$

Greedy Ablation Pseudo Code

Algorithm 1 Greedy Ablation

Input : Algorithm $\mathcal{A}$ with configuration space $\mathbf{\Lambda}$, start configuration $\boldsymbol{\lambda}^{(\text{start})}$, end configuration $\boldsymbol{\lambda}^{(\text{end})}$, cost metric c

$\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda}^{(\text{start})}$;

$P \leftarrow [\,]$;

Greedy Ablation Pseudo Code

Algorithm 2 Greedy Ablation

Input : Algorithm $\mathcal{A}$ with configuration space $\mathbf{\Lambda}$, start configuration $\boldsymbol{\lambda}^{(\text{start})}$, end configuration $\boldsymbol{\lambda}^{(\text{end})}$, cost metric c

$\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda}^{(\text{start})}$;

$P \leftarrow []$;

foreach $t \in \{1 \ldots |\mathbf{\Lambda}|\}$ **do**

Greedy Ablation Pseudo Code

Algorithm 3 Greedy Ablation

Input : Algorithm $\mathcal{A}$ with configuration space $\mathbf{\Lambda}$, start configuration $\boldsymbol{\lambda}^{(\text{start})}$, end configuration $\boldsymbol{\lambda}^{(\text{end})}$, cost metric c

$\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda}^{(\text{start})}$;
 $P \leftarrow []$;
 foreach $t \in \{1 \ldots |\mathbf{\Lambda}|\}$ **do**
  foreach $\delta \in \Delta(\boldsymbol{\lambda}, \boldsymbol{\lambda}^{(end)})$ **do**
   $\boldsymbol{\lambda}'_\delta \leftarrow$ apply δ to $\boldsymbol{\lambda}$;
   evaluate $c(\boldsymbol{\lambda}'_\delta)$;

Greedy Ablation Pseudo Code

Algorithm 4 Greedy Ablation

Input : Algorithm $\mathcal{A}$ with configuration space $\mathbf{\Lambda}$, start configuration $\boldsymbol{\lambda}^{(\text{start})}$, end configuration $\boldsymbol{\lambda}^{(\text{end})}$, cost metric c

```
λ ← λ^(start);
P ← [] ;
foreach t ∈ {1 ... |Λ|} do
    foreach δ ∈ Δ(λ, λ^(end)) do
        λ'_δ ← apply δ to λ;
        evaluate c(λ'_δ);
    Determine most important change δ* ∈ argmin_{δ∈Δ(λ,λ^(end))} c(λ_δ);
    λ ← apply δ* to λ;
    P.append(δ*);
```

Greedy Ablation Pseudo Code

Algorithm 5 Greedy Ablation

Input : Algorithm $\mathcal{A}$ with configuration space $\mathbf{\Lambda}$, start configuration $\boldsymbol{\lambda}^{(\text{start})}$, end configuration $\boldsymbol{\lambda}^{(\text{end})}$, cost metric c

$\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda}^{(\text{start})}$;
$P \leftarrow []$;
foreach $t \in \{1 \ldots |\mathbf{\Lambda}|\}$ **do**
  foreach $\delta \in \Delta(\boldsymbol{\lambda}, \boldsymbol{\lambda}^{(end)})$ **do**
    $\boldsymbol{\lambda}'_\delta \leftarrow$ apply δ to $\boldsymbol{\lambda}$;
    evaluate $c(\boldsymbol{\lambda}'_\delta)$;
  Determine most important change $\delta^* \in \arg\min_{\delta \in \Delta(\boldsymbol{\lambda}, \boldsymbol{\lambda}^{(\text{end})})} c(\boldsymbol{\lambda}_\delta)$;
  $\boldsymbol{\lambda} \leftarrow$ apply δ^* to $\boldsymbol{\lambda}$;
  P.append(δ^*);
return *Ablation path* P

Remarks on Ablation

- Even this greedy ablation requires $\mathcal{O}(n^2)$ steps

Remarks on Ablation

- Even this greedy ablation requires $\mathcal{O}(n^2)$ steps

$\leadsto$ We can also speedup that up by using surrogate models
[Biedenkapp et al. 2017]

Remarks on Ablation

- Even this greedy ablation requires $\mathcal{O}(n^2)$ steps

$\rightsquigarrow$ We can also speedup that up by using surrogate models [Biedenkapp et al. 2017]

- Common observations:
  1. Some hyperparameters might not matter ($\boldsymbol{\lambda}_2$ in the example)

Remarks on Ablation

- Even this greedy ablation requires $\mathcal{O}(n^2)$ steps

$\rightsquigarrow$ We can also speedup that up by using surrogate models [Biedenkapp et al. 2017]

- Common observations:
  1. Some hyperparameters might not matter ($\boldsymbol{\lambda}_2$ in the example)
  2. Often only a few of the hyperparameters have an big impact

Remarks on Ablation

- Even this greedy ablation requires $\mathcal{O}(n^2)$ steps

⇝ We can also speedup that up by using surrogate models
[Biedenkapp et al. 2017]

- Common observations:
  1. Some hyperparameters might not matter ($\boldsymbol{\lambda}_2$ in the example)
  2. Often only a few of the hyperparameters have an big impact
  3. You have plateaus in your ablation path because of interaction effects