



THE DEVELOPER'S CONFERENCE

Trilha – Arquitetura Python

Antônio Janael Pinheiro

Doutor em ciência da computação
Engenheiro de software no CESAR

Agenda



- Contexto
- Séries temporais
- Predição de séries temporais
- Desafios da predição
- Implementação
- Resultados
- Conclusão



THE
DEVELOPER'S
CONFERENCE

Avaliação de modelos preditivos para séries temporais em Python

Contexto



- Dados são produzidos em volume, velocidade e variedade crescentes;
- Entender comportamentos passados para prever o futuro.

Séries temporais



- Dados ordenados sequências ao longo do tempo;
- Dados componentes temporais.

Séries temporais



- Oportunidades:
 - Predição;
 - Classificação;
 - Detecção de anomalias.

Predição de séries temporais



THE
DEVELOPER'S
CONFERENCE

Produção de energia elétrica da China



Dados disponibilizados publicamente pelo *Federal Reserve Economic*

Data (FRED)

Desafios da avaliação



Como garantimos que o nosso modelo de predição é eficiente?

Desafios da avaliação



- A avaliação de modelos preditivos é processo complexo;
- Escolha da métrica mais adequada.

Desafios da avaliação



- Métricas:
 - *Mean Absolute Error* (MAE);
 - *Root Mean Squared Error* (RMSE);
 - U de *Theil*.

Desafios da avaliação



- U de *Theil*:
 - Valida o modelo preditivo em relação ao método ingênuo;
 - Métricas como MAE e RMSE podem produzir valores baixos, ainda que o modelo seja inferior ao método ingênuo.

Desafios da avaliação



THE
DEVELOPER'S
CONFERENCE

- U de *Theil*:

$$U1 = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - f_i)^2}}{\sqrt{\frac{1}{N} \sum_{i=1}^N y_i^2} + \sqrt{\frac{1}{N} \sum_{i=1}^N f_i^2}}$$

$$U2 = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^{N-1} \left(\frac{f_{i+1} - y_{i+1}}{y_i} \right)^2}}{\sqrt{\frac{1}{N} \sum_{i=1}^{N-1} \left(\frac{y_{i+1} - y_i}{y_i} \right)^2}}$$

Desafios da avaliação



- U de *Theil*:

$$U2 = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^{N-1} \left(\frac{f_{i+1} - y_{i+1}}{y_i} \right)^2}}{\sqrt{\frac{1}{N} \sum_{i=1}^{N-1} \left(\frac{y_{i+1} - y_i}{y_i} \right)^2}}$$

Desafios da avaliação



- U de *Theil*:

$$U2 = \frac{\text{Erro do modelo preditivo}}{\text{Erro do método ingênuo}}$$

Desafios da avaliação



- Quando o valor de U_2 é menor que 1, o modelo analisado é superior ao método ingênuo.

Desafios da avaliação



- Quando o valor de U_2 é menor que 1, o modelo analisado é superior ao método ingênuo;
- **Caso a U_2 seja maior que 1, o modelo é inferior ao mecanismo ingênuo.**

Desafios da avaliação



- Quando o valor de $U2$ é menor que 1, o modelo analisado é superior ao método ingênuo;
- Caso a $U2$ seja maior que 1, o modelo é inferior ao mecanismo ingênuo;
- **Se $U2$ for igual a 1, o modelo avaliado apresenta um desempenho idêntico ao método ingênuo.**

Implementação



- Escassez de implementações da U de Theil em Python.

Implementação



- Biblioteca *numpy* para as operações matemáticas;
- Operações com *arrays* por questões de desempenho.

Implementação



THE
DEVELOPER'S
CONFERENCE

- `subtraction = y_pred[1:] - y_true[1:]`

$$f_{i+1} - y_{i+1}$$

Implementação



- `subtraction = y_pred[1:] - y_true[1:]`
- `division = subtraction / y_true[:-1]`

$$\frac{f_{i+1} - y_{i+1}}{y_i}$$

Implementação



THE
DEVELOPER'S
CONFERENCE

- `subtraction = y_pred[1:] - y_true[1:]`
- `division = subtraction / y_true[:-1]`
- `power = np.power(division, 2)`

$$\left(\frac{f_{i+1} - y_{i+1}}{y_i} \right)^2$$

Implementação



THE
DEVELOPER'S
CONFERENCE

- `subtraction = y_pred[1:] - y_true[1:]`
- `division = subtraction / y_true[:-1]`
- `power = np.power(division, 2)`
- `summation = np.sum(power)`

$$\sum_{i=1}^{N-1} \left(\frac{f_{i+1} - y_{i+1}}{y_i} \right)^2$$

Implementação



THE
DEVELOPER'S
CONFERENCE

- `subtraction = y_pred[1:] - y_true[1:]`
- `division = subtraction / y_true[:-1]`
- `power = np.power(division, 2)`
- `summation = np.sum(power)`
- `N = len(y_true)`
- `mean = (1/N) * summation`

$$\frac{1}{N} \sum_{i=1}^{N-1} \left(\frac{f_{i+1} - y_{i+1}}{y_i} \right)^2$$

Implementação



THE
DEVELOPER'S
CONFERENCE

- `subtraction = y_pred[1:] - y_true[1:]`
- `division = subtraction / y_true[:-1]`
- `power = np.power(division, 2)`
- `summation = np.sum(power)`
- `N = len(y_true)`
- `mean = (1/N) * summation`
- `numerator = np.sqrt(mean)`

$$\sqrt{\frac{1}{N} \sum_{i=1}^{N-1} \left(\frac{f_{i+1} - y_{i+1}}{y_i} \right)^2}$$

Implementação



THE
DEVELOPER'S
CONFERENCE

- `denominator_subtraction = y_true[1:] - y_true[:-1]`

$$y_{i+1} - y_i$$

Implementação



- `denominator_subtraction = y_true[1:] - y_true[:-1]`
- `denominator_division = denominator_subtraction / y_true[:-1]`

$$\frac{y_{i+1} - y_i}{y_i}$$

Implementação



THE
DEVELOPER'S
CONFERENCE

- `denominator_subtraction = y_true[1:] - y_true[:-1]`
- `denominator_division = denominator_subtraction / y_true[:-1]`
- `denominator_power = np.power(denominator_division, 2)`

$$\left(\frac{y_{i+1} - y_i}{y_i} \right)^2$$

Implementação



THE
DEVELOPER'S
CONFERENCE

- `denominator_subtraction = y_true[1:] - y_true[:-1]`
- `denominator_division = denominator_subtraction / y_true[:-1]`
- `denominator_power = np.power(denominator_division, 2)`
- `denominator_summation = np.sum(denominator_power)`

$$\sum_{i=1}^{N-1} \left(\frac{y_{i+1} - y_i}{y_i} \right)^2$$

Implementação



THE
DEVELOPER'S
CONFERENCE

- `denominator_subtraction = y_true[1:] - y_true[:-1]`
- `denominator_division = denominator_subtraction / y_true[:-1]`
- `np.power(denominator_division, 2)`
- `denominator_summation = np.sum(power)`
- `N = len(y_true)`
- `mean = (1/N) * denominator_summation`

$$\frac{1}{N} \sum_{i=1}^{N-1} \left(\frac{y_{i+1} - y_i}{y_i} \right)^2$$

Implementação

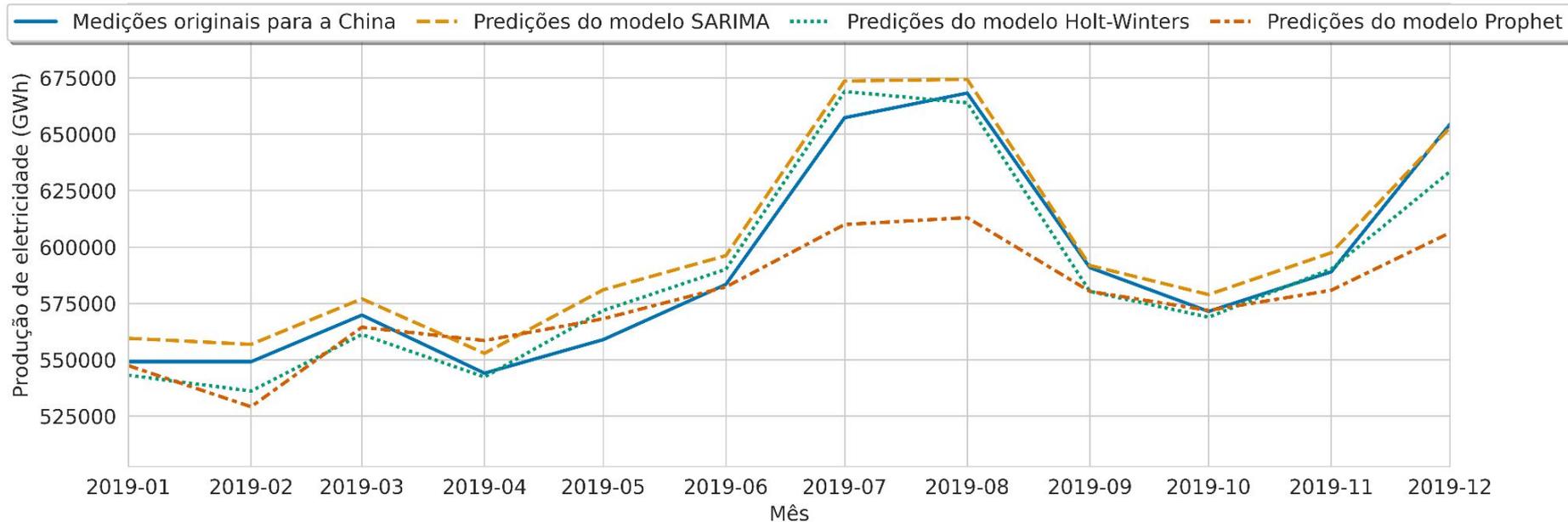


THE
DEVELOPER'S
CONFERENCE

- `theil_u = numerator / denominator`

$$\frac{\sqrt{\frac{1}{N} \sum_{i=1}^{N-1} \left(\frac{f_{i+1} - y_{i+1}}{y_i} \right)^2}}{\sqrt{\frac{1}{N} \sum_{i=1}^{N-1} \left(\frac{y_{i+1} - y_i}{y_i} \right)^2}}$$

Resultados



Com base na métrica U de Thiel, os três modelos são superiores ao método ingênuo.

Resultados

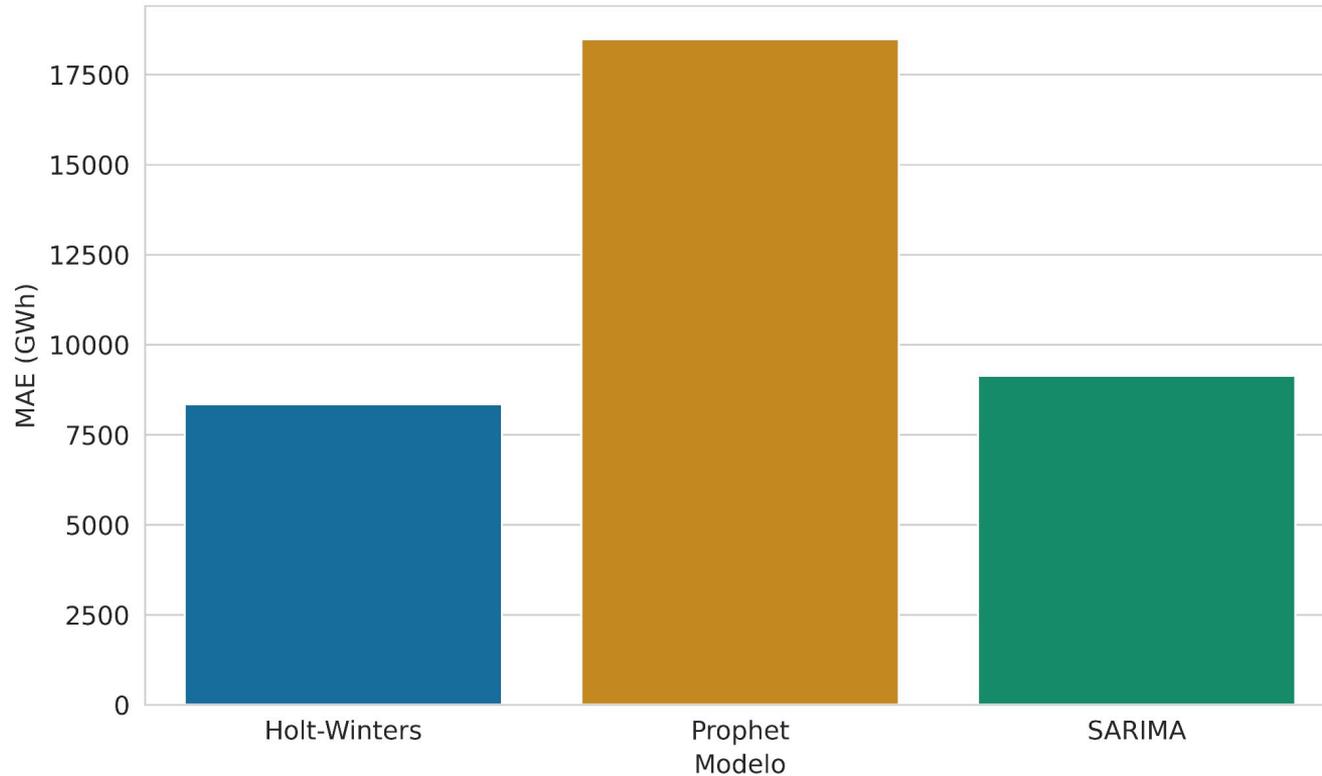


Algoritmo	U2
Holt-Winters	0.26
SARIMA	0.38
Facebook Prophet	0.68

Resultados



THE
DEVELOPER'S
CONFERENCE



Conclusão



- Séries temporais são fundamentais em diversos domínios;
- Avaliação de modelos preditivos não é trivial;
- Aprofundamento e código: <https://bit.ly/309v5GQ>

Dúvidas?



THE
DEVELOPER'S
CONFERENCE



Contato

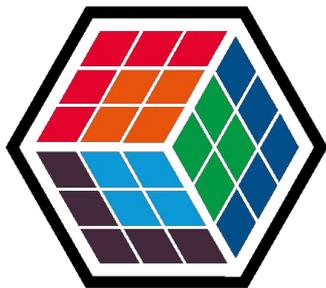


THE
DEVELOPER'S
CONFERENCE

 janael-pinheiro

 ajanaelpinheiro@gmail.com

 janael-pinheiro



THE DEVELOPER'S CONFERENCE