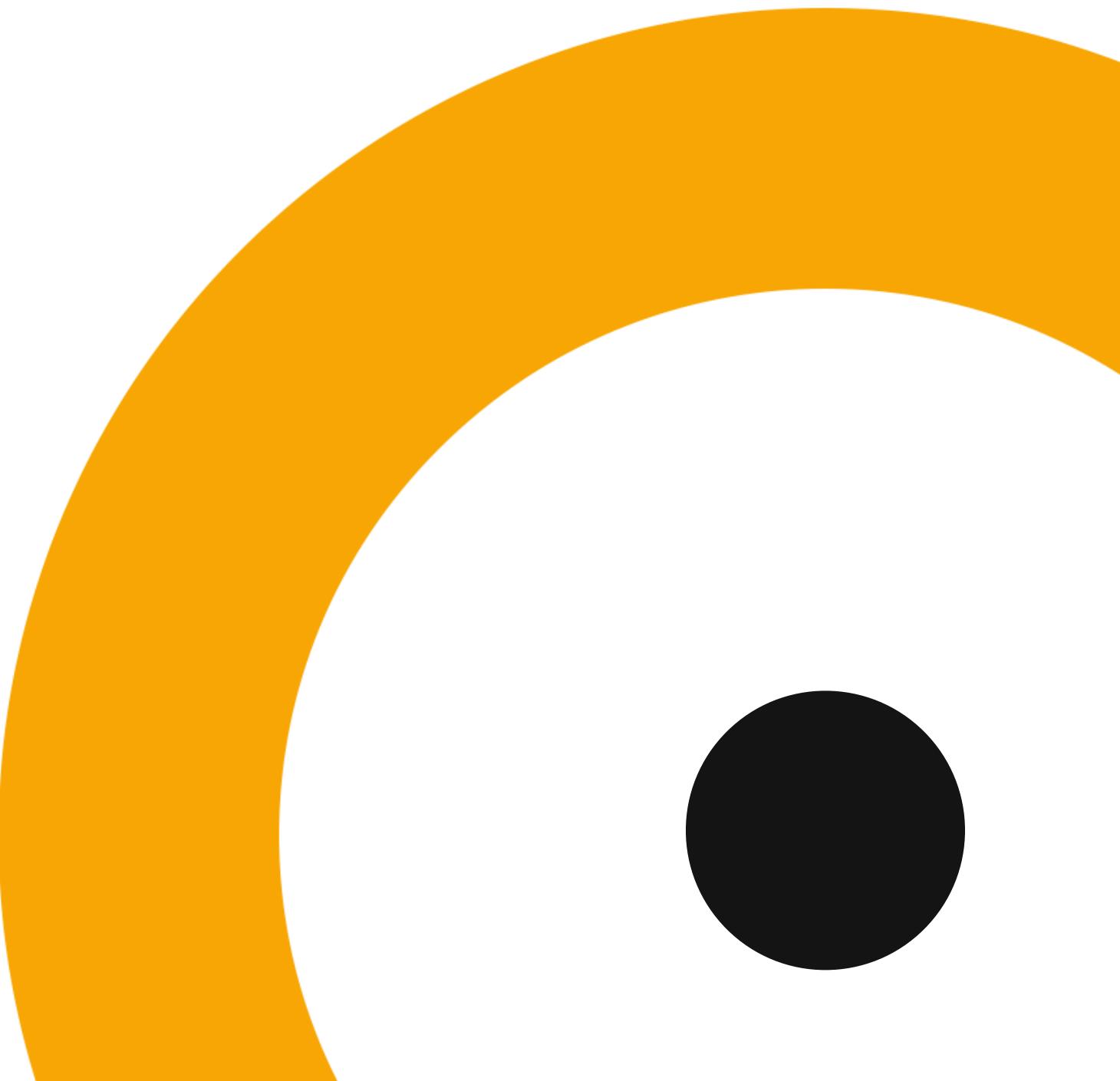


Mestrado em Data Science

# Séries Temporais

Avaliação de Modelos de Séries Temporais



# Conteúdo

4.1 Validação

---

4.2 Métricas de Avaliação

---

4.3 Validação Cruzada

---

## 4.1 Validação



# Avaliação de Modelos de Previsão de Séries Temporais

## Estimação de Desempenho

Estimar o erro que um modelo irá cometer quando aplicado em novos dados, desconhecidos até então.

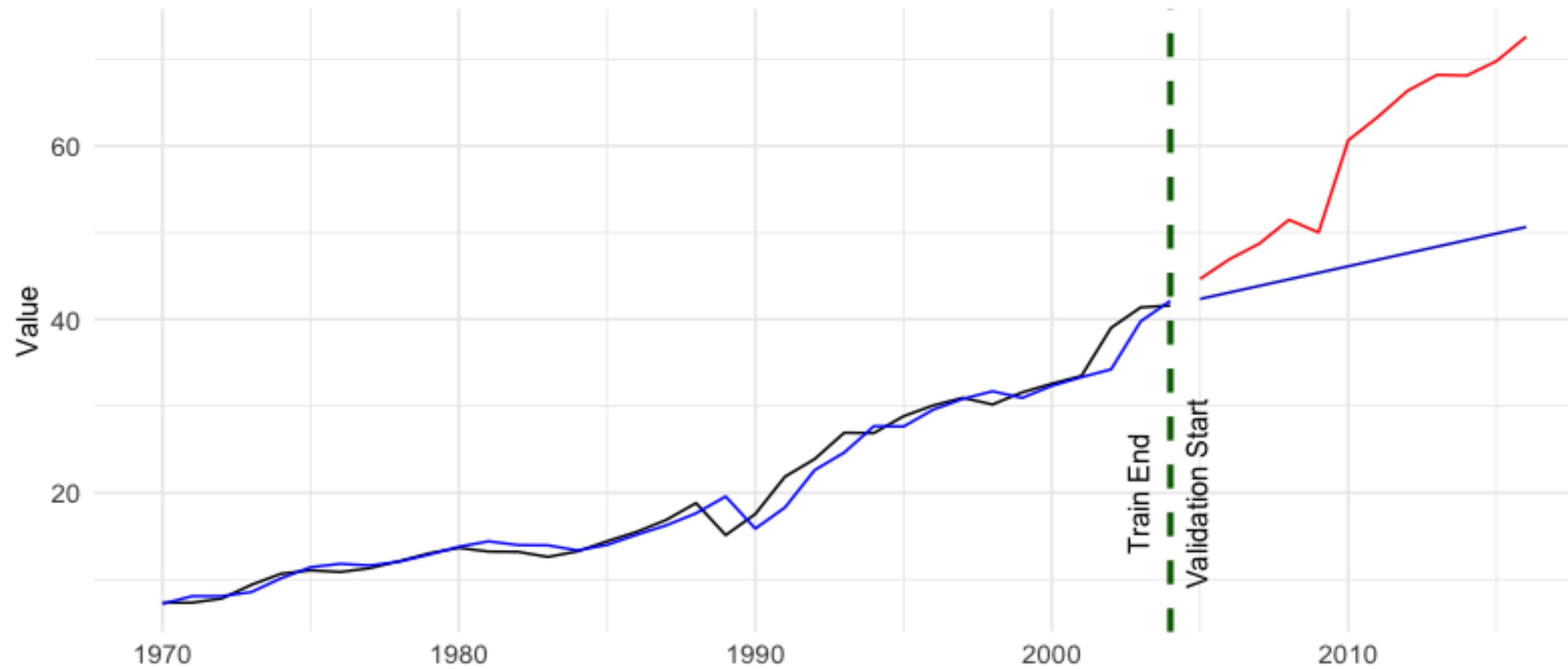
- Importante para perceber se o modelo poderá ser usado

## Seleção de Modelos

Usar as estimativas de desempenho para selecionar o melhor modelo dentro de várias alternativas.

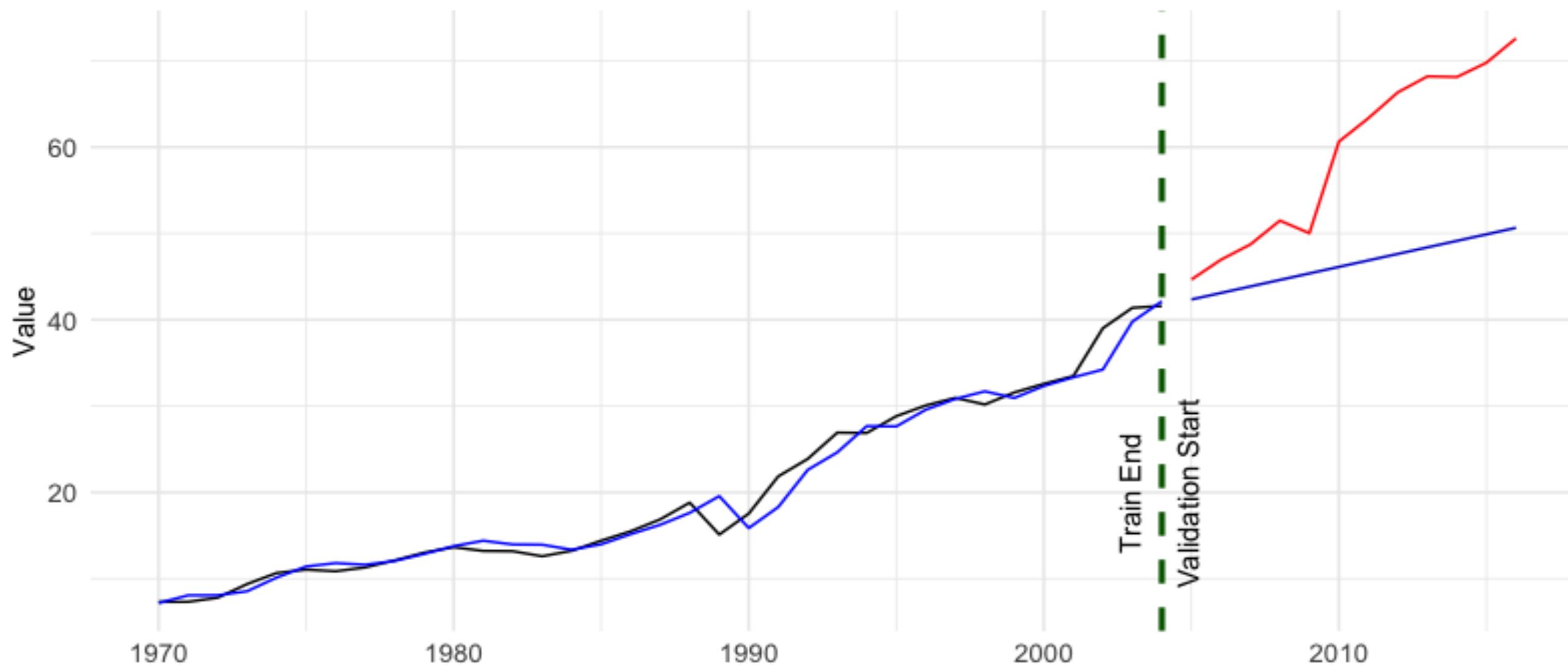
# Usando um Conjunto de Validação

- Treinar o modelo e avaliá-lo nos mesmos dados leva a estimativas otimistas.



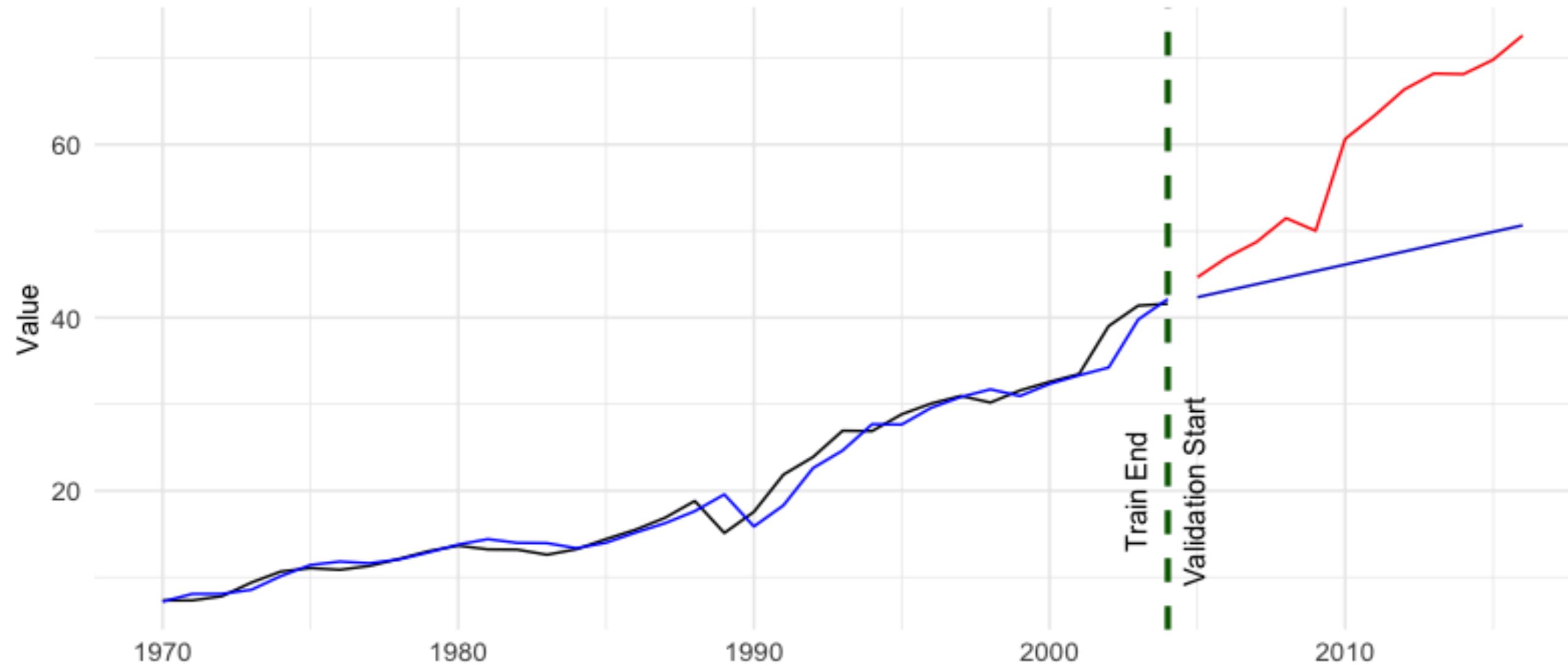
# Usando um Conjunto de Validação

Emular um cenário realista: Guardar a última parte dos dados para avaliação



# Usando um Conjunto de Validação

Regra dourada: Não quebrar a ordem temporal dos dados



# Exemplo com a biblioteca Darts

```
import pandas as pd
import matplotlib.pyplot as plt
from darts.models import ExponentialSmoothing
from darts import TimeSeries

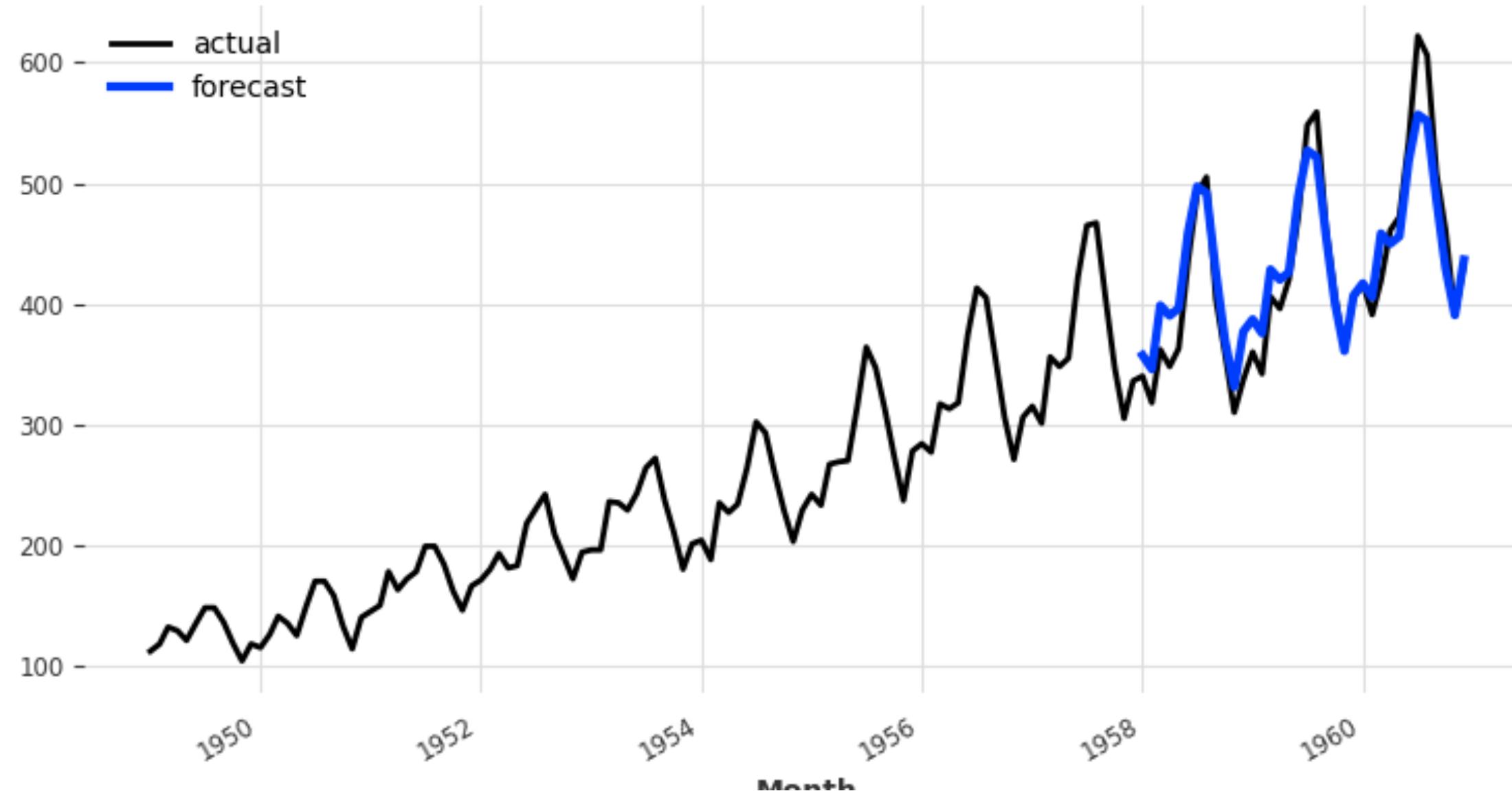
data = pd.read_csv('air_passengers.csv')
series = TimeSeries.from_dataframe(data, 'Month', 'V1')

train, validation = series.split_before(pd.Timestamp('19580101'))

model = ExponentialSmoothing()
model.fit(train)
prediction = model.predict(len(validation))

series.plot(label='actual')
prediction.plot(label='forecast', lw=3)
plt.legend()
```

# Exemplo com a biblioteca Darts



4.2

## Métricas de Avaliação



# Desempenho de Previsão

## Erro de Previsão

Diferença entre previsões e valores reais

## Erro de Validação

É fundamental calcular o erro em dados de validação.

- Ainda assim, o erro nos dados de treino pode ser um complemento para perceber se existe sobre-ajustamento aos dados (overfit)

# Desempenho de Previsão

## Definições

- Previsões:  $\hat{y}_i$
- Valores reais:  $y_i$

## Erro

$$e_i = y_i - \hat{y}_i$$

# Métricas

## Mean Absolute Error (MAE)

$$\text{mean}(|e|)$$

## Mean Squared Error (MSE)

$$\text{mean}(e^2)$$

- Penaliza erros maiores em relação ao MAE

# Métricas

## Root Mean Squared Error (RMSE)

$$\sqrt{\text{mean} (e^2)}$$

## Mean Absolute Percentage Error (MAPE)

$$100 \times \text{mean} \left( \frac{|e|}{|y|} \right)$$

- Poderá levar a problemas numéricos para valores perto de 0

# Métricas

## Mean Absolute Scaled Error (MASE)

$$\frac{\text{MAE}}{\text{MAE}_{\text{Naive}}(\text{train})}$$

- Independente da escala;
- Não tem problemas se a série tende para 0 (como o MAPE);

# Exemplo com Darts

```
import pandas as pd

from darts.models import ExponentialSmoothing
from darts import TimeSeries
from sklearn.metrics import mean_absolute_error

data = pd.read_csv('air_passengers.csv')

series = TimeSeries.from_dataframe(data, 'Month', 'V1')

train, validation = series.split_before(pd.Timestamp('19580101'))

model = ExponentialSmoothing()
model.fit(train)
prediction = model.predict(len(validation))

mean_absolute_error(validation.values().flatten(), prediction.values().flatten())
# 21.54
```

## 4.3

# Validação Cruzada



# Validação Cruzada

## Definição

Partição da série múltiplas vezes.

- Em cada partição, parte dos dados é usada para treinar o modelo (o conjunto de treino). A parte restante (conjunto de validação) é usada para avaliar o desempenho desse modelo.

## Múltiplas Partições

- Uma única divisão pode ser tendenciosa devido à origem selecionada;
- Várias divisões cobrirão diferentes partes da série.
  - Por exemplo, diferentes regimes ou padrões de tendência/sazonalidade.

# Recomendações Gerais para Avaliação

## 1 - Preservar Ordem Temporal

- Cada observação depende um pouco do que aconteceu antes.
- Ao quebrar a ordem temporal:
  - Modelo aprende nuances do futuro, que não se revelaram no passado.

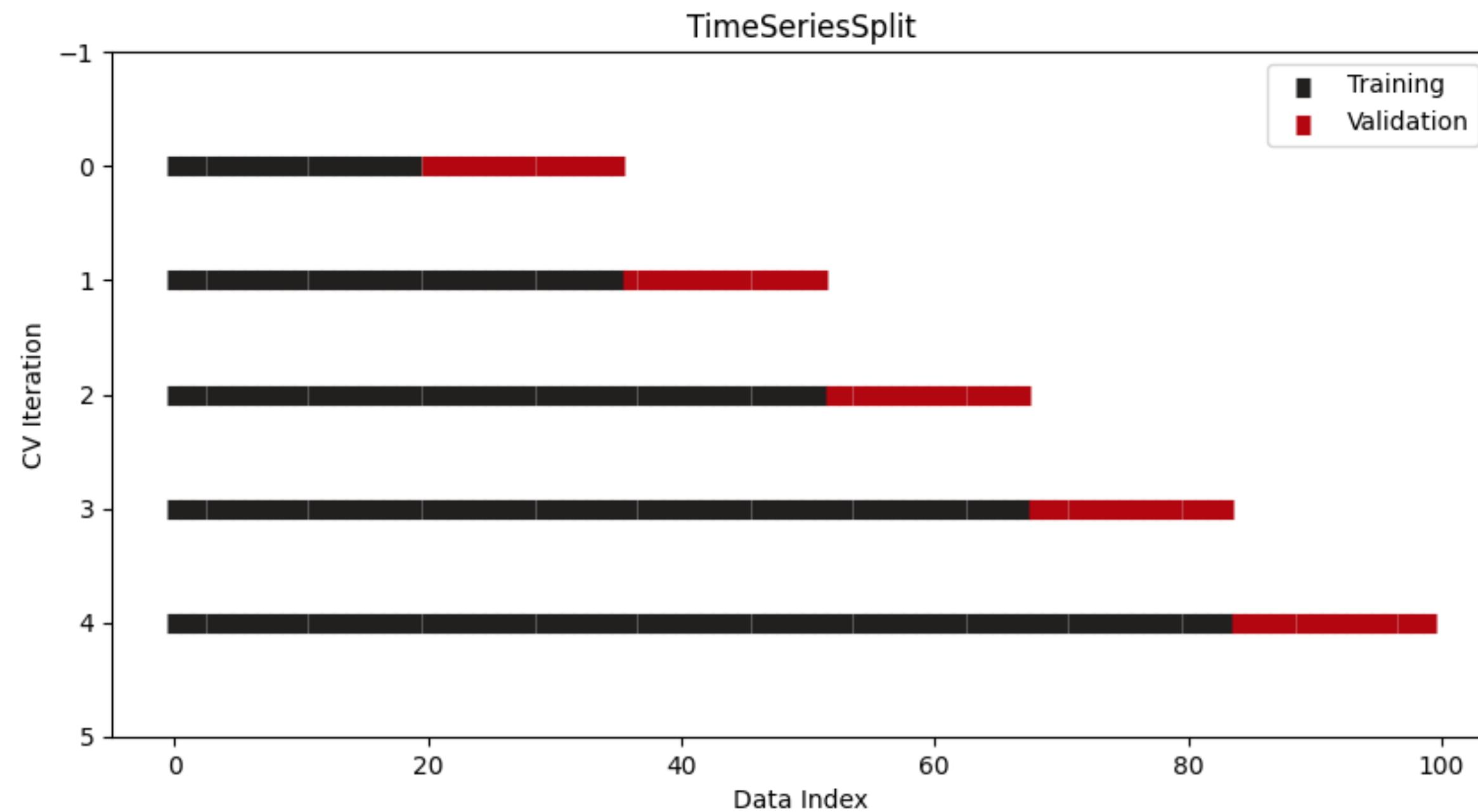
## 2 - Criar Múltiplas Iterações

- Uma única divisão pode ser tendenciosa devido à origem selecionada.
  - Não é tão importante para grandes conjuntos de dados
  - 5 a 10 iterações é um valor comum

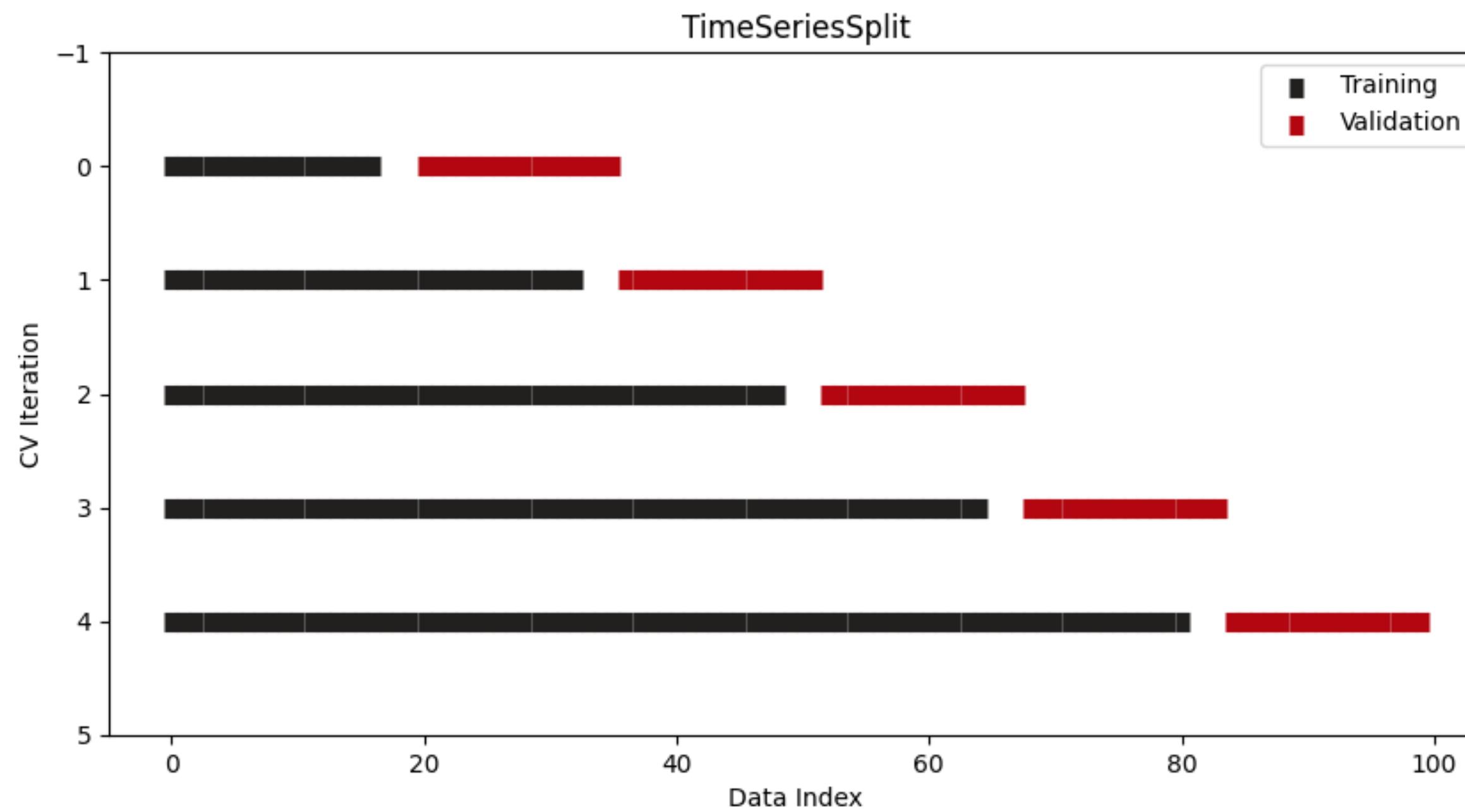
## 3 - Deixar um Espaço entre Treino e Validação

- A parte inicial do conjunto de validação é altamente correlacionada com a última parte do conjunto de treinamento.
  - Então pode ser importante remover observações adjacentes

# Validação Cruzada para Séries Temporais



# Validação Cruzada para Séries Temporais



# Exemplo

```
import pandas as pd
import numpy as np

from statsforecast.models import auto_arima
from sklearn.model_selection import TimeSeriesSplit
from sklearn.metrics import mean_absolute_error

data = pd.read_csv('air_passengers.csv')
series = data.set_index('Month')['V1']

tscv = TimeSeriesSplit(n_splits=5)

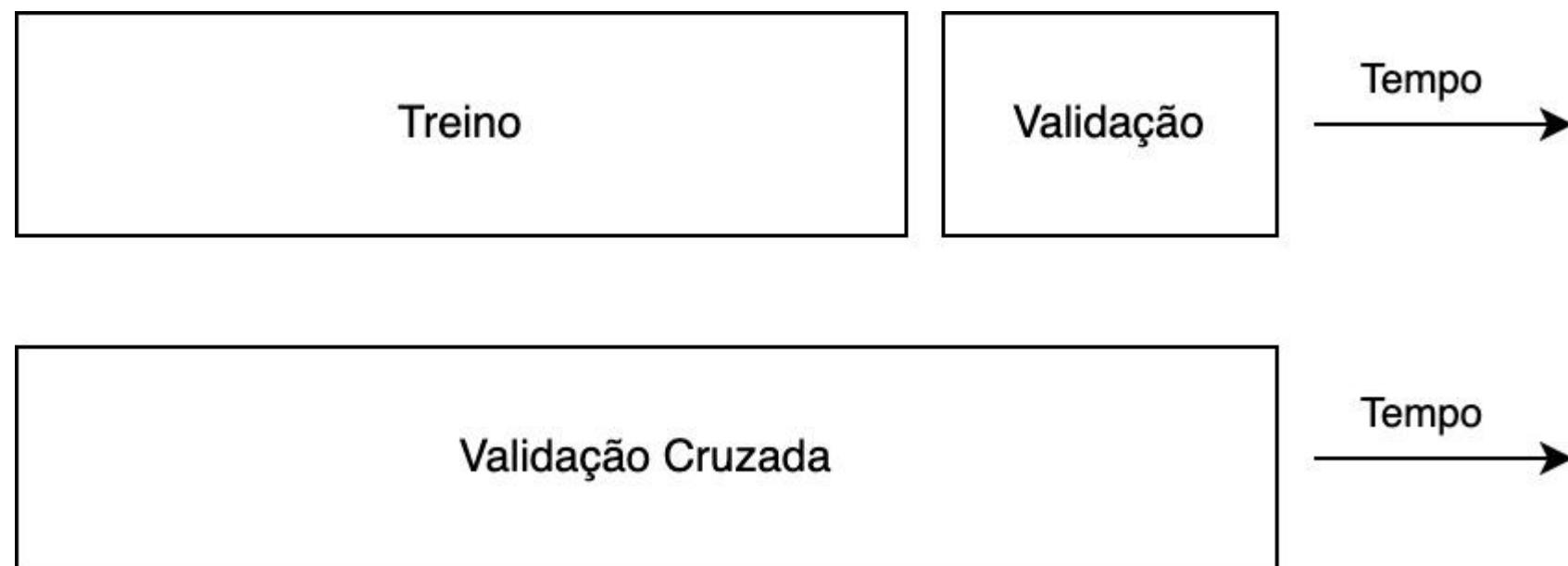
cv_error = []
for train_index, test_index in tscv.split(series):
    forecasts = auto_arima(series.values[train_index], h=len(test_index))

    err = mean_absolute_error(series.values[test_index], forecasts)
    cv_error.append(err)

model_error = np.mean(cv_error)
```

# Seleção de Modelo: Usar 3 Partições

De validação para validação cruzada



- O modelo selecionado é aquele que minimiza o erro de validação ou validação cruzada

# Usando 3 Partições



## Treino, Validação, Teste

- Usar apenas treino e validação pode também enviesar os resultados
  - modelo ou parâmetros são escolhidos com base em validação, mas essa escolha pode não se manter no futuro
- Deixar outra partição para **teste**
  - Escolher modelo/parâmetros em validação/validação cruzada
  - Estimar o desempenho final no teste
    - Re-treinar modelo final com treino+validação

# Usando 3 Partições



## Treino, Validação, Teste

- Usar apenas treino e validação pode também enviesar os resultados
  - modelo ou parâmetros são escolhidos com base em validação, mas essa escolha pode não se manter no futuro
- Deixar outra partição para **teste**
  - Escolher modelo/parâmetros em validação/validação cruzada
  - Estimar o desempenho final no teste
    - Re-treinar modelo final com treino+validação

# Avaliação de Modelos de Séries Temporais

**Vitor Cerqueira**

<https://www.linkedin.com/in/vcerq/>  
cerqueira.vitormanuel@gmail.com

