

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
NÚCLEO DE EDUCAÇÃO A DISTÂNCIA
Pós-graduação *Lato Sensu* em Ciência de Dados e Big Data

Evandro Ferreira de Avellar

PREVISÃO DE FLUXO DE CAIXA PARA INVESTIMENTO DE VALORES DO FGTS

Brasília
2023

Evandro Ferreira de Avellar

PREVISÃO DE FLUXO DE CAIXA PARA INVESTIMENTO DE VALORES DO FGTS

Trabalho de Conclusão de Curso apresentado
ao Curso de Especialização em Ciência de
Dados e Big Data como requisito parcial à
obtenção do título de especialista.

Brasília
2023

SUMÁRIO

1. Introdução.....	4
1.1. Contextualização.....	4
1.2. O problema proposto.....	4
1.3. Objetivos.....	5
1.4. Conceitos que serão utilizados.....	5
2. Coleta de Dados.....	7
2.1 Arquivo ArrecadaçãoConsolidada.xls.....	7
2.2 Arquivo SaquesConsolidados.xls.....	7
3. Processamento/Tratamento de Dados.....	8
4. Análise e Exploração dos Dados.....	10
4.1 Visualizando os dados dos datasets iniciais.....	10
4.3 Modelos de erro/tendência/sazonalidade.....	13
5. Criação de Modelos.....	14
5.1 Média Móvel Simples (SMA).....	14
5.2 Modelo EWMA (Média móvel ponderada exponencialmente).....	15
5.3 Suavização Exponencial Dupla (DES).....	17
5.4 Suavização Exponencial Tripla (TES).....	18
Os mesmos testes foram efetuados para os Saques e o modelo que teve o melhor desempenho foi o RMSA.....	19
6. Previsões (Forecasting).....	20
. 6. Interpretação dos Resultados.....	20
. 7. Apresentação dos Resultados.....	22
. 8. Links.....	23
REFERÊNCIAS.....	24
Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: principles and practice. OTexts.....	24
. APÊNDICE.....	25

1. Introdução

1.1. Contextualização

A CAIXA ECONÔMICA FEDERAL é o agente operador do FGTS, como agente operador é responsável na prestação de serviços financeiros para controle de saldo, entradas e saídas, possibilitando a aplicação dos recursos do fundo em várias carteiras de investimento de curto, médio e longo prazo.

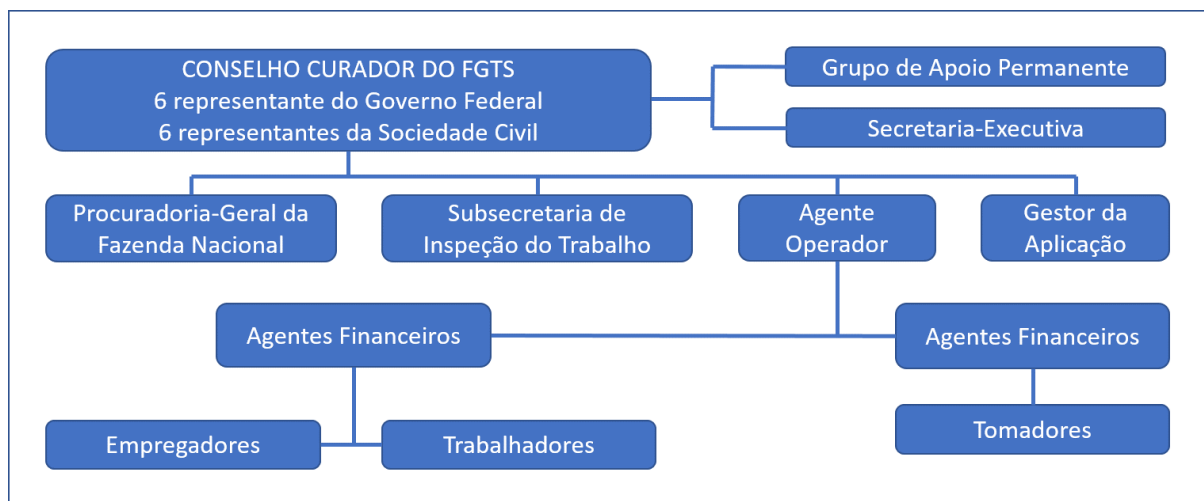


Figura 1: Administração do FGTS

O resultado do FGTS consiste na diferença entre as receitas (rendas/rendimentos com operações de crédito, com títulos públicos federais e demais títulos e valores mobiliários, entre outras) e as despesas (remuneração das contas vinculadas, de TR + 3% ao ano, taxa de administração e outras).¹

1.2. O problema proposto

É um problema importante para sociedade brasileira pois o recurso do FGTS nasceu com o objetivo de garantir ao trabalhador uma indenização pelo tempo de serviço nos casos de demissão sem justa causa e ainda propiciar a formação de uma reserva a ser utilizada por ele, quando de sua aposentadoria, ou por seus dependentes, quando do seu falecimento.

O FGTS é também uma fonte de recursos para financiamento de programas habitacionais, de saneamento básico e de infraestrutura urbana.

Os dados analisados são públicos e estão disponíveis no site do FGTS, o trabalho do TCC estará limitado às informações do site.

¹Site do FGTS

O objetivo desse trabalho é encontrar um modelo preditivo utilizando técnicas de modelos estatísticos e de *machine learning* com assertividade e embasamento teórico para apresentar ao Agente Operador do Fundo, permitindo gerar previsões de entradas e saídas de fluxo de caixa.

O período dos dados que serão analisados é de 01/1997 a 10/2022. Durante os trabalhos pode ser necessário restringir um período menor mais recente para maior assertividade nas previsões.

1.3. Objetivos

Encontrar um modelo de previsão de Fluxo de Caixa utilizando técnicas de *Machine Learning* para permitir aos gestores do fundo planejarem os investimentos de curto, médio e longo prazo com os recursos do FGTS, garantindo a liquidez dos valores necessários para cumprir as obrigações sem perder o melhor retorno com os investimentos.

1.4. Conceitos que serão utilizados

Séries temporais: Uma série temporal é uma sequência de pontos ordenados de forma cronológica. Normalmente, a série temporal possui uma sequência de dados equidistantes no tempo. A análise de séries temporais é realizada com o intuito de explorar o comportamento passado e também de prever o comportamento futuro em um determinado problema.

Previsão: a aplicação mais comum quando tratamos de séries temporais é a previsão de valores futuros (forecast). A previsão de série temporal é o processo de usar um modelo estatístico para prever valores futuros de uma série temporal com base em resultados anteriores.

Componentes de uma série temporal:

Tendência: a tendência mostra uma direção geral dos dados da série temporal durante um longo período de tempo. Uma tendência pode ser crescente (para cima), decrescente (para baixo) ou horizontal (estacionária). (MORETTIN; TOLOI, 2006).

Sazonalidade: o componente de sazonalidade exibe uma tendência que se repete em relação ao tempo, direção e magnitude. Alguns exemplos incluem um aumento no consumo de água no verão devido às condições climáticas quentes ou um au-

mento no número de passageiros de companhias aéreas durante as férias a cada ano. (MONTGOMERY; JONHSON; GARDNER, 2015).

Varição irregular: são as flutuações nos dados da série temporal que se tornam evidentes quando a tendência e as variações cíclicas são removidas. Essas variações são imprevisíveis, erráticas e podem ou não ser aleatórias. (MONTGOMERY; JONHSON; GARDNER, 2015).

2. Coleta de Dados

Todos os dados são públicos e foram obtidos no site do FGTS na seção Números do FGTS / Passivo: <https://www.fgts.gov.br/Pages/numeros-fgts/passivo-fgts.aspx> extraídos no dia 09 de dezembro de 2022.

Dois datasets foram utilizados no estudo:

- ArrecadaçãoConsolidada.xls
- SaquesConsolidados.xls

Originalmente os dados foram coletados com as seguintes características, antes do tratamento dos dados e logo após a importação para o notebook:

2.1 Arquivo ArrecadaçãoConsolidada.xls

Nome da coluna	Descrição	Tipo
Mês/Ano	Mês/Ano de arrecadação no formato mm/aaaa	String
QUANTIDADE(GUIAS)	Quantidade de guias referentes ao Mês/Ano	Número inteiro
VALOR (R\$)	Valor total arrecadado no Mês/Ano	Float

2.2 Arquivo SaquesConsolidados.xls

Nome da coluna	Descrição	Tipo
MÊS/ANO	Mês/Ano do saque no formato mm/aaaa	String
QUANTIDADE	Quantidade de saques referentes ao Mês/Ano	Número inteiro
VALOR (R\$)	Valor total de saques no Mês/Ano	Float

3. Processamento/Tratamento de Dados

A IDE utilizada para realização das análises é o Jupyter-Notebook.

Para processamento e tratamento dos dados será utilizada a biblioteca Pandas. A importação pode ser feita através do código: `import pandas as pd`.

Os arquivos disponibilizados no site estão no formato xls, na leitura e conversão para o Data Frame com nome *arrec* foi preciso desconsiderar as primeiras linhas do arquivo excel pois se tratavam de informações explicativas sobre o conteúdo. A exclusão das linhas do cabeçalho foi realizada com o parâmetro **header** do método `pd.read_excel()`.

Os parâmetros `index_col` e `parse_dates` transformaram o índice no formato de data.

Dados de Arrecadação

```
In [6]: 1 arrec = pd.read_excel('dataset/ArrecadaçãoConsolidada.xls', header=8, index_col='Mês/Ano', parse_dates=True)
```

```
In [7]: 1 arrec
```

```
Out[7]:
```

	Unnamed: 1	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	QUANTIDADE(GUIAS)	Unnamed: 7	VALOR (R\$)
Mês/Ano								
01/1997	NaN	NaN	NaN	NaN	NaN	1910996	NaN	1.355415e+09
02/1997	NaN	NaN	NaN	NaN	NaN	1864867	NaN	1.012064e+09
03/1997	NaN	NaN	NaN	NaN	NaN	1955507	NaN	9.969265e+08
04/1997	NaN	NaN	NaN	NaN	NaN	2028810	NaN	1.013695e+09
05/1997	NaN	NaN	NaN	NaN	NaN	2010079	NaN	1.021931e+09
...
07/2022	NaN	NaN	NaN	NaN	NaN	6037145	NaN	1.253901e+10
08/2022	NaN	NaN	NaN	NaN	NaN	6285605	NaN	1.290656e+10
09/2022	NaN	NaN	NaN	NaN	NaN	6170799	NaN	1.262671e+10
10/2022	NaN	NaN	NaN	NaN	NaN	6242096	NaN	1.261092e+10
Total	NaN	NaN	NaN	NaN	NaN	1350260636	NaN	1.789601e+12

311 rows × 8 columns

A última linha do dataset também continha informações de totalização, que não seriam úteis ao estudo. A remoção se deu através do código abaixo:

```
In [8]: 1 arrec = arrec.head(310)
        2 arrec
```

```
Out[8]:
```

	Unnamed: 1	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	QUANTIDADE(GUIAS)	Unnamed: 7	VALOR (R\$)
Mês/Ano								
01/1997	NaN	NaN	NaN	NaN	NaN	1910996	NaN	1.355415e+09
02/1997	NaN	NaN	NaN	NaN	NaN	1864867	NaN	1.012064e+09
03/1997	NaN	NaN	NaN	NaN	NaN	1955507	NaN	9.969265e+08
04/1997	NaN	NaN	NaN	NaN	NaN	2028810	NaN	1.013695e+09
05/1997	NaN	NaN	NaN	NaN	NaN	2010079	NaN	1.021931e+09
...
06/2022	NaN	NaN	NaN	NaN	NaN	6011827	NaN	1.240369e+10
07/2022	NaN	NaN	NaN	NaN	NaN	6037145	NaN	1.253901e+10
08/2022	NaN	NaN	NaN	NaN	NaN	6285605	NaN	1.290656e+10
09/2022	NaN	NaN	NaN	NaN	NaN	6170799	NaN	1.262671e+10
10/2022	NaN	NaN	NaN	NaN	NaN	6242096	NaN	1.261092e+10

310 rows × 8 columns

Foram selecionadas as colunas que continham os dados, descartando as colunas que ficaram vazias no Data Frame, através do código a seguir:

```
In [10]: 1 df_arrec = arrec[['QUANTIDADE(GUIAS)', 'VALOR (R$)']]
          2 df_arrec.head()
```

```
Out[10]:
```

	QUANTIDADE(GUIAS)	VALOR (R\$)
Mês/Ano		
01/1997	1910996	1.355415e+09
02/1997	1864867	1.012064e+09
03/1997	1955507	9.969265e+08
04/1997	2028810	1.013695e+09
05/1997	2010079	1.021931e+09

Todos esses passos para as informações dos registros de arrecadação também foram seguidos para tratamento do *dataset* de saques alterando alguns parâmetros:

Dados de saques

```
In [14]: 1 saque = pd.read_excel('dataset/SaquesConsolidados.xls', header=7, index_col='MÊS/ANO', parse_dates=True)
          2 saque = saque.head(310)
          3 df_saque = saque[['QUANTIDADE', 'VALOR (R$)']]
          4 df_saque.head()
```

```
Out[14]:
```

	QUANTIDADE	VALOR (R\$)
MÊS/ANO		
1997-01-01	1028047	9.792931e+08
1997-02-01	891569	8.404584e+08
1997-03-01	953599	9.850851e+08
1997-04-01	1046439	1.116987e+09
1997-05-01	967636	1.149939e+09

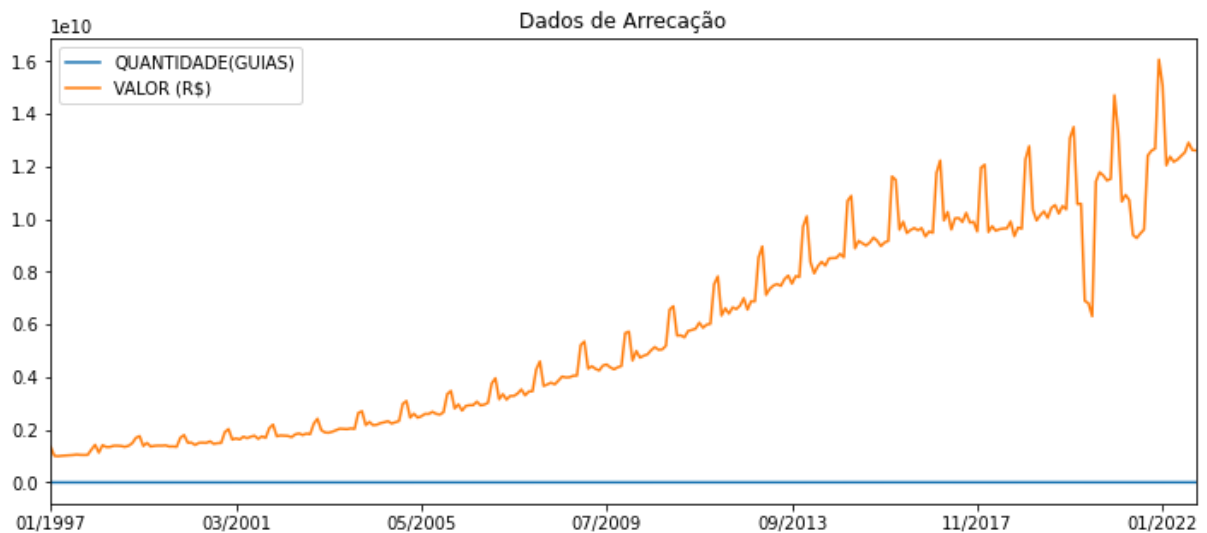
Não foram identificados dados ausentes nas colunas selecionadas.

4. Análise e Exploração dos Dados

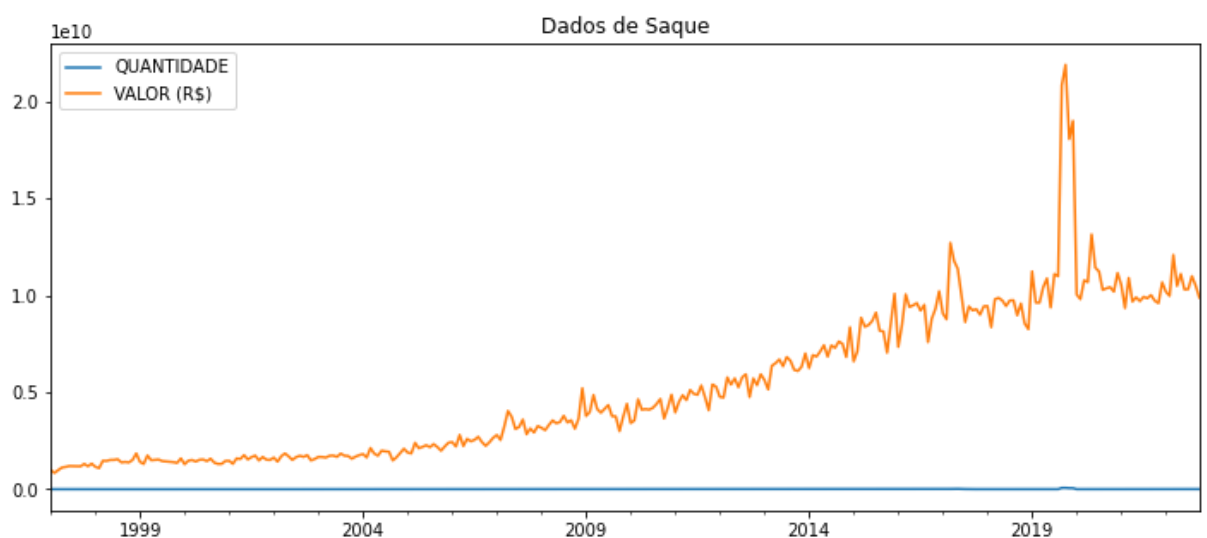
A visualização dos dados auxilia na tomada de decisão para escolher o melhor caminho da análise e exploração. Os dois Data Frames geraram os gráficos abaixo:

4.1 Visualizando os dados dos datasets iniciais

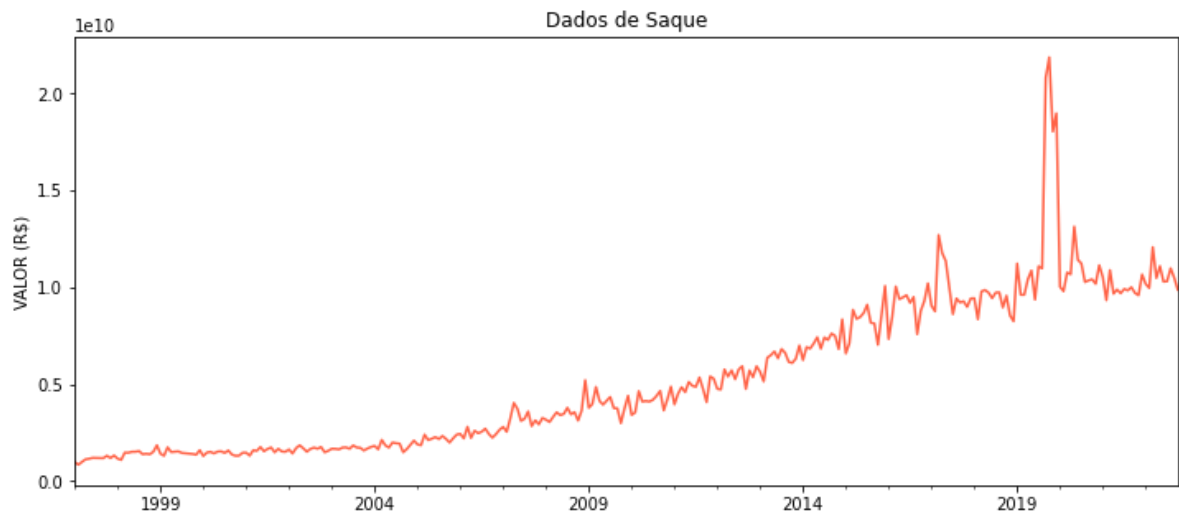
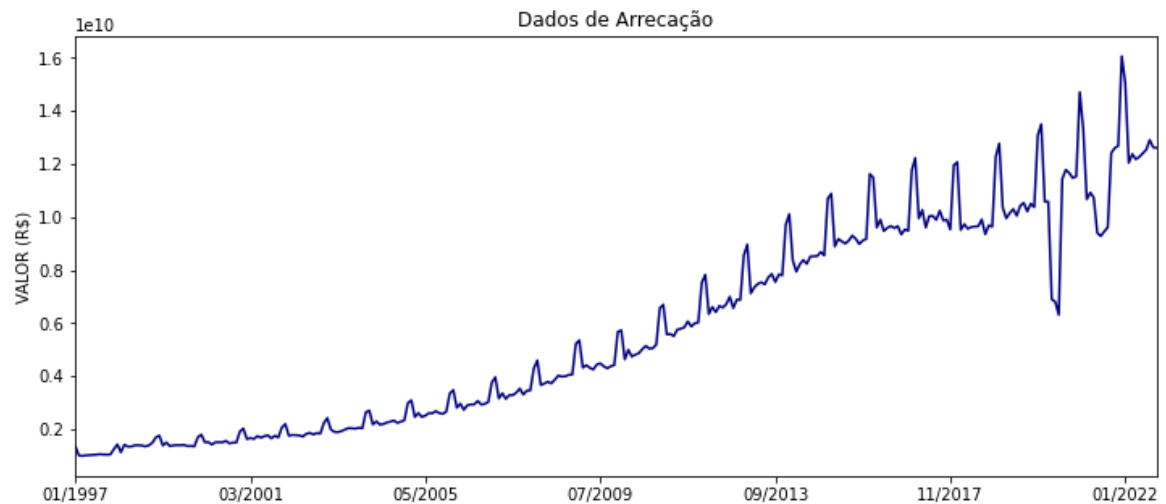
```
ax = df_arrec.plot(figsize=(12,5), title='Dados de Arrecação')
ax.autoscale(axis='x',tight=True)
ax.set(ylabel='', xlabel='');
```



```
ax = df_saque.plot(figsize=(12,5), title='Dados de Saque')
ax.autoscale(axis='x',tight=True)
ax.set(ylabel='', xlabel='');
```



Através da análise dos gráficos serão descartadas as colunas de quantidade dos dois Data Frames, e as análises serão efetuadas somente da coluna valor.

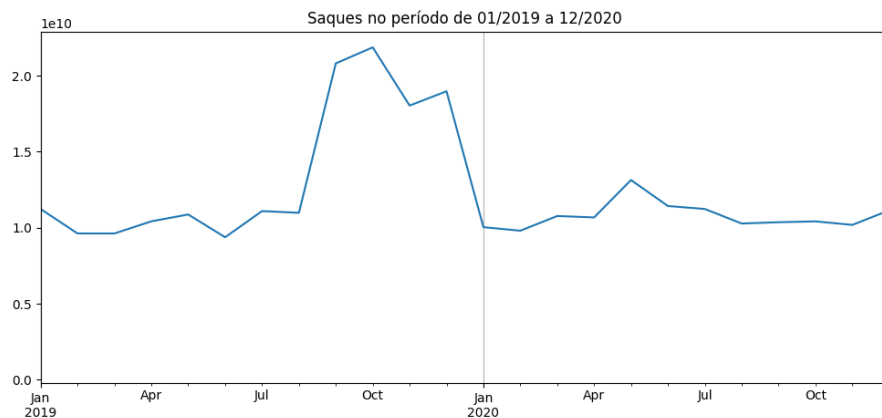


Observando o gráfico de saques é possível verificar um comportamento fora do padrão entre 09/2019 a 01/2020. Ao pesquisar no site do FGTS é possível concluir que esse comportamento foi gerado pelo Saque Imediato (2019/2020).

```
[3]: df_saque.idxmax()

[3]: QUANTIDADE    2019-10-01
      VALOR (R$)    2019-10-01
      dtype: datetime64[ns]

[52]: ax = df_saque['VALOR (R$)'].plot(xlim=['2019-01-01', '2020-12-01'], figsize=(12,5))
      ax.set(xlabel='Mês')
      ax.set_title('Saques no período de 01/2019 a 12/2020')
      ax.xaxis.grid(True)
```



4.2.1 Informações sobre o Saque Imediato (2019/2020) que estão no site do FGTS

Os trabalhadores puderam solicitar o Saque Emergencial no período regular de setembro de 2019 a março de 2020.

Para viabilizar os saques, ocorreram os débitos nas contas vinculadas dos trabalhadores, conforme tabela descritiva abaixo:

Mês/ano	Quantidade*	Valor **
Set/19	64.735,6	10.164
Out/19	65.359,9	10.469
Nov/19	51.892,2	8.447
Dez/19	49.666,5	8.568
Jan/20	326,2	76
Fev/20	231,6	54
Mar/20	360,0	91
Abr/20	184,3	45
Total	232.757	37.915

* Quantidade em mil

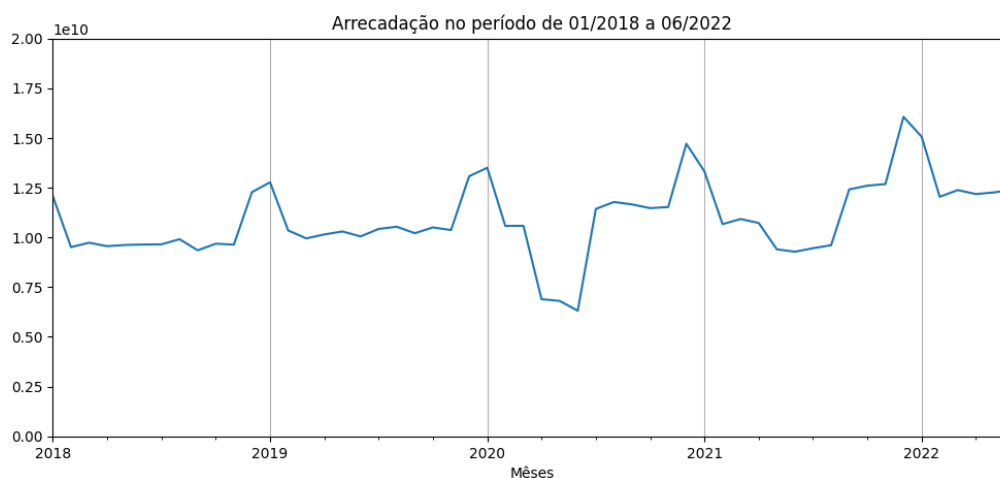
** Valor em R\$ milhão

O comportamento da arrecadação também apresenta um comportamento anormal no ano de 2020 com queda expressiva no mês de junho de 2020, o que pode ter sido gerado pelo impacto da pandemia COVID-19 na economia do país.

```
[72]: df_arrec['VALOR (R$)']['2018-01-01':'2022-06-01'].idxmin()

[72]: Timestamp('2020-06-01 00:00:00')

[74]: ax = df_arrec['VALOR (R$)'].plot(xlim=['2018-01-01', '2022-06-01'], figsize=(12,5), ylim=(0,2e10))
ax.set(xlabel='Mês')
ax.set_title('Arrecadação no período de 01/2018 a 06/2022')
ax.xaxis.grid(True)
```



Durante os estudos essas informações serão consideradas não impactar na geração do modelo preditivo.

4.3 Modelos de erro/tendência/sazonalidade

Ao começarmos a trabalhar com dados endógenos e começarmos a desenvolver modelos de previsão, é útil identificar e isolar fatores que atuam dentro do sistema e influenciam o comportamento. Aqui, o nome "endógeno" considera fatores internos, enquanto "exógeno" se relaciona a forças externas. Esses modelos se enquadram na categoria de modelos de espaço de estado e incluem a decomposição e o alisamento exponencial (descrito em uma seção futura) (MORETTIN; TOLOI, 2006).

A decomposição de uma série temporal tenta isolar componentes individuais, como erro, tendência e sazonalidade (ETS).

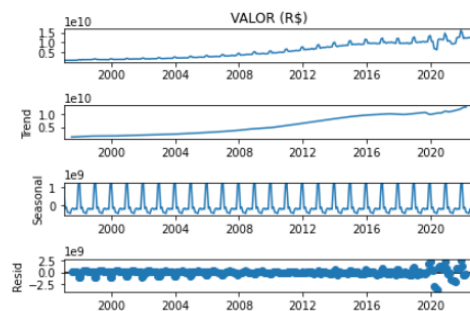
A função `seasonal_decompose` do módulo `statsmodels.tsa.seasonal` é útil para analisar as diferentes componentes de uma série temporal, incluindo a tendência, sazonalidade e os resíduos (ou erro). Através dessa análise, podemos ter uma melhor compreensão do comportamento da série temporal e identificar padrões que possam ser úteis na criação de modelos preditivos. Além disso, a decomposição pode ser usada para remover a sazonalidade e a tendência de uma série, tornando os dados mais estacionários e, assim, permitindo que técnicas estatísticas mais avançadas sejam aplicadas. No entanto, é importante lembrar que a decomposição não é uma técnica de previsão por si só e deve ser usada em conjunto com outros métodos para fazer previsões precisas.

É possível observar que as duas séries temporais possuem tendência e sazonalidade.

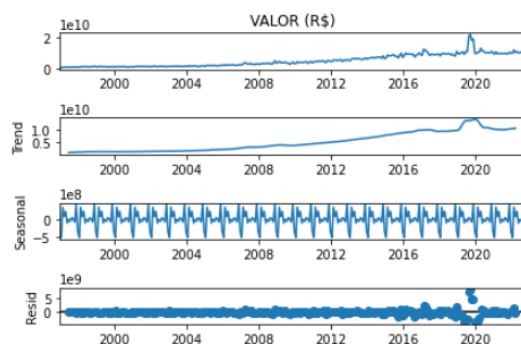
Decomposição ETS

```
In [113]: 1 from statsmodels.tsa.seasonal import seasonal_decompose
```

```
In [125]: 1 result = seasonal_decompose(df_arrec['VALOR (R$)'], model='add')
          2 result.plot();
```



```
In [126]: 1 result = seasonal_decompose(df_saue['VALOR (R$)'], model='add')
          2 result.plot();
          3
```



5. Criação de Modelos

Serão aplicados diversos modelos para comparação do modelo que melhor consegue prever com maior assertividade.

Para comparação, além dos gráficos, serão utilizados dois métodos para avaliação de desempenho dos modelos, o MSE e o RMSE.

MSE (Mean Squared Error) é um método de avaliação de desempenho de modelos de previsão que calcula a média dos erros quadrados das previsões em relação aos valores reais. Quanto menor o valor do MSE, melhor é a qualidade das previsões do modelo.

RMSE (Root Mean Squared Error) é a raiz quadrada do MSE e é utilizado para expressar o erro médio de previsão em uma mesma unidade da variável de interesse. É uma medida de dispersão dos erros em relação ao valor real. Quanto menor o valor do RMSE, mais preciso é o modelo de previsão.

5.1 Média Móvel Simples (SMA)

A aplicação da média móvel simples, **MMS** ou **SMA** (*Simple Moving Average*), sobre uma sequência $(p_i)_{i=1}^m$ resulta na sequência das médias das subsequências de n elementos da sequência, tal que $n < m$, e $n, m \in \mathbb{N}$. Desta maneira, dado uma sequência de m elementos $P = (p_1, \dots, p_m)$, o cálculo de um termo qualquer da sequência resultante pela média móvel é dado por

$$\begin{aligned}\bar{p}_i &= \frac{p_{i+1} + \dots + p_{i+n}}{n} \\ &= \frac{1}{n} \sum_{j=1}^n p_{i+j} \\ \text{e } \bar{p}_{i+1} &= \bar{p}_i + \frac{p_{n+i+1}}{n} - \frac{p_{i+1}}{n}.\end{aligned}$$

É preciso expandir os extremos de uma sequência de dados por $n-1$ elementos para aplicar a média móvel a todos os seus elementos; a maneira mais simples de fazer essa expansão é adicionar $n-1$ zeros aos seus extremos.

A média móvel simples possui 3 formas de ser aplicada: considerando-se os n elementos posteriores, futuros como foi feito acima; os n elementos anteriores, passados (como em séries temporais); ou considerando-se ainda elementos anteriores e posteriores. Neste último caso, apenas $n/2$ elementos devem ser adicionados aos dois extremos. (MÉDIA MÓVEL. In: WIKIPÉDIA)

É possível observar que a média móvel simples já consegue prever com alguma assertividade.

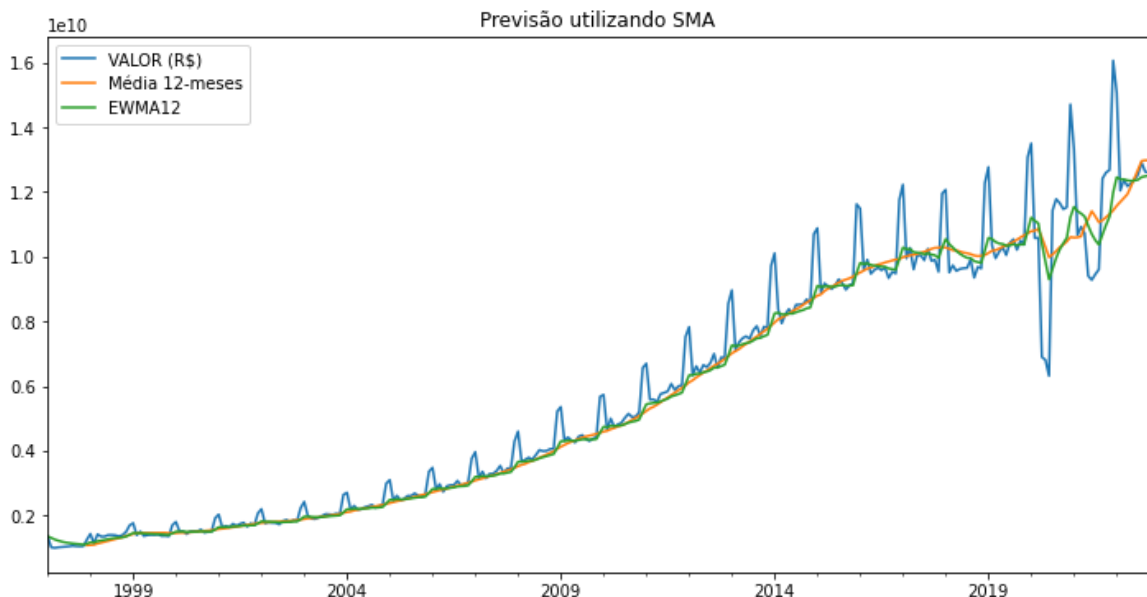
Média Móvel Simples

```
In [137]: 1 df_arrec['Média 12-meses'] = df_arrec['VALOR (R$)'].rolling(window=12).mean()
```

```
In [159]: 1 df_arrec
```

Out[159]:

	VALOR (R\$)	Média 12-meses
Mês/Ano		
1997-01-01	1.355415e+09	NaN
1997-02-01	1.012064e+09	NaN
1997-03-01	9.969265e+08	NaN
1997-04-01	1.013695e+09	NaN
1997-05-01	1.021931e+09	NaN
...
2022-06-01	1.240369e+10	1.243104e+10
2022-07-01	1.253901e+10	1.268786e+10
2022-08-01	1.290656e+10	1.296263e+10
2022-09-01	1.262671e+10	1.298019e+10
2022-10-01	1.261092e+10	1.298088e+10



5.2 Modelo EWMA (Média móvel ponderada exponencialmente)

No item anterior foi mostrado como calcular a SMA com base em uma janela. No entanto, a SMA básica tem algumas fraquezas:

- Janelas menores levarão a mais ruído, em vez de sinal.
- Sempre irá atrasar pelo tamanho da janela.
- Nunca alcançará o pico ou vale completo dos dados devido à média.
- Realmente não informa sobre possíveis comportamentos futuros, tudo o que realmente faz é descrever tendências em seus dados.
- Valores históricos extremos podem distorcer significativamente sua SMA.

Para ajudar a corrigir algumas dessas questões, podemos usar uma EWMA (Média móvel ponderada exponencialmente).

A Média Móvel Ponderada Exponencialmente (EWMA, do inglês Exponentially Weighted Moving Average) é um método utilizado na análise técnica de dados financeiros para calcular uma média móvel ponderada que dá mais peso aos valores mais recentes dos dados analisados. Esse método foi desenvolvido como uma alternativa à Média Móvel Simples (SMA, do inglês Simple Moving Average), que pode apresentar algumas limitações em determinadas situações.

O cálculo da EWMA é feito através de uma fórmula que aplica pesos exponenciais crescentes aos valores mais recentes dos dados, dando mais importância a esses valores em detrimento dos mais antigos. Dessa forma, a EWMA é capaz de suavizar a curva de uma série de dados, reduzindo o ruído e o efeito de atraso que podem ser observados na SMA. (HUANG; ZHOU; ZHANG, 2003)

Média móvel ponderada exponencialmente

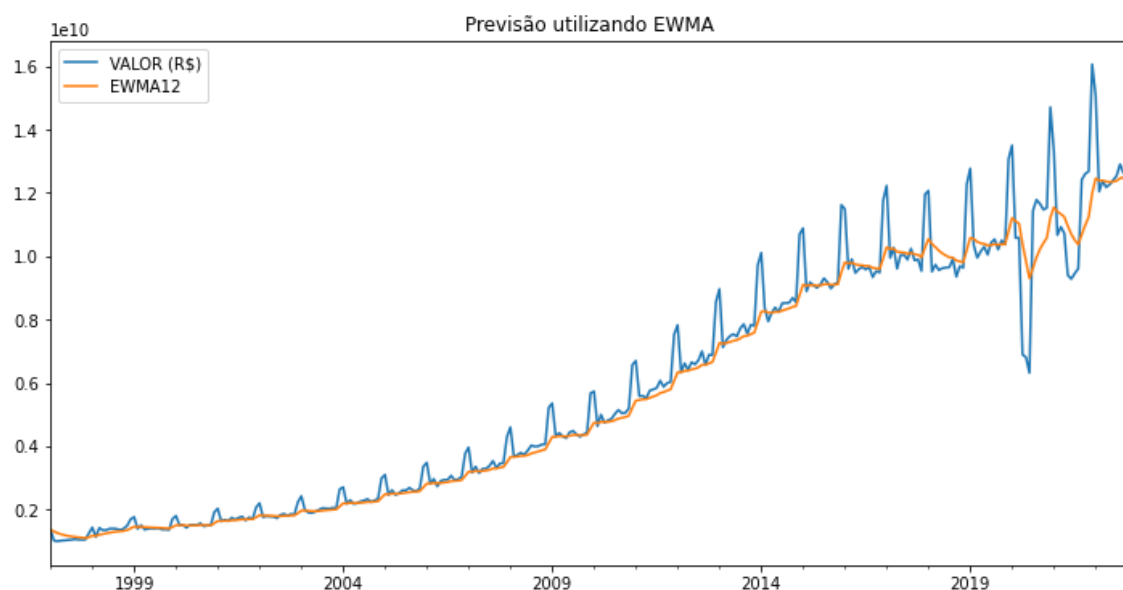
```
In [160]: 1 df_arrec['EWMA12'] = df_arrec['VALOR (R$)'].ewm(span=12,adjust=False).mean()
```

```
In [161]: 1 df_arrec
```

```
Out[161]:
```

	VALOR (R\$)	Média 12-meses	EWMA12
Mês/Ano			
1997-01-01	1.355415e+09	NaN	1.355415e+09
1997-02-01	1.012064e+09	NaN	1.302592e+09
1997-03-01	9.969265e+08	NaN	1.255566e+09
1997-04-01	1.013695e+09	NaN	1.218355e+09
1997-05-01	1.021931e+09	NaN	1.188136e+09
...
2022-06-01	1.240369e+10	1.243104e+10	1.235419e+10
2022-07-01	1.253901e+10	1.268786e+10	1.238263e+10
2022-08-01	1.290656e+10	1.296263e+10	1.246323e+10
2022-09-01	1.262671e+10	1.298019e+10	1.248838e+10
2022-10-01	1.261092e+10	1.298088e+10	1.250723e+10

310 rows × 3 columns



Através do gráfico é possível observar que houve um ganho no desempenho quanto utilizamos o EWMA.

Utilizando os métodos quantitativos é possível confirmar que o modelo EWMA apresenta um número menor de erros.

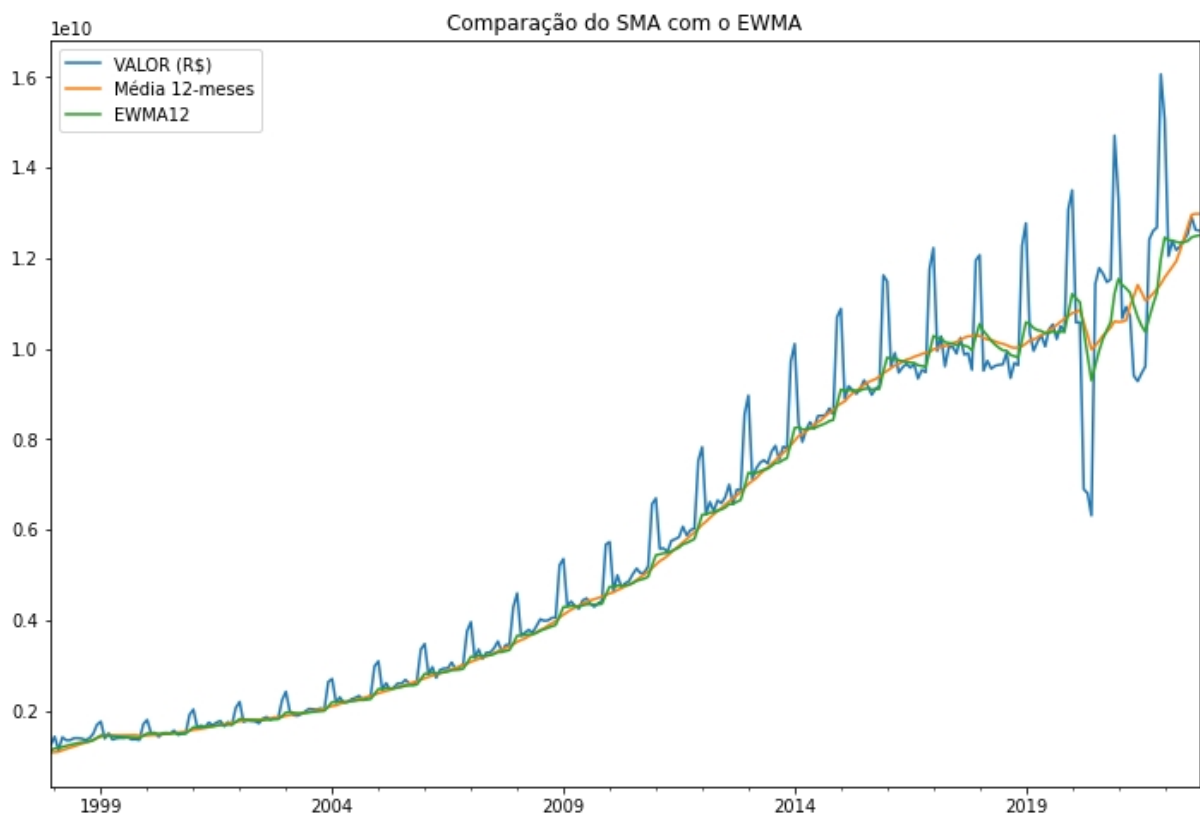
Avaliação de Desempenho

```
In [143]: 1 from sklearn.metrics import mean_squared_error
          2 from statsmodels.tools.eval_measures import rmse

In [166]: 1 arrec_desemp = df_arrec.iloc[11:]

In [167]: 1 error1 = mean_squared_error(arrec_desemp['VALOR (R$)'], arrec_desemp['Média 12-meses'])
          2 error2 = mean_squared_error(arrec_desemp['VALOR (R$)'], arrec_desemp['EWMA12'])
          3 error3 = rmse(arrec_desemp['VALOR (R$)'], arrec_desemp['Média 12-meses'])
          4 error4 = rmse(arrec_desemp['VALOR (R$)'], arrec_desemp['EWMA12'])
          5
          6 print(f'MSE SMA: {error1:11.10}')
          7 print(f'MSE EWMA: {error2:11.10}')
          8 print(f'RMSE SMA: {error3:11.10}')
          9 print(f'RMSE EWMA: {error4:11.10}')
```

```
MSE SMA: 7.555003786e+17
MSE EWMA: 5.986829452e+17
RMSE SMA: 869195247.7
RMSE EWMA: 773746047.0
```



5.3 Suavização Exponencial Dupla (DES)

A Suavização Exponencial Dupla (ou Double Exponential Smoothing, em inglês) é uma técnica de previsão de séries temporais que incorpora o componente de tendência do con-

junto de dados. Ela é uma extensão da Suavização Exponencial Simples (SES) que, além do fator de suavização α , adiciona um segundo fator de suavização β para capturar a tendência.

De acordo com Jain e Kumar (2021), a Suavização Exponencial Dupla se tornou uma das técnicas mais populares para previsão de séries temporais devido à sua simplicidade e eficácia. Ela é amplamente utilizada em áreas como finanças, vendas, operações de negócios, entre outras. Além disso, a técnica é capaz de lidar com tendências lineares e exponenciais, permitindo ajustes aditivos ou multiplicativos, respectivamente.

A Suavização Exponencial Dupla é implementada utilizando uma combinação de médias móveis ponderadas e suavização exponencial simples. O primeiro passo é calcular a média móvel ponderada (ou weighted moving average) dos valores da série temporal. Em seguida, é aplicada a suavização exponencial simples para essa média móvel ponderada, que gera uma previsão suavizada da série temporal.

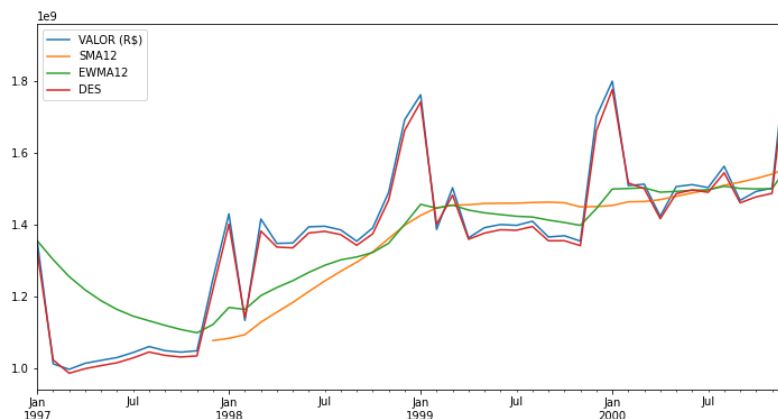
Suavização Exponencial Dupla (DES)

```
In [178]: 1 from statsmodels.tsa.holtwinters import ExponentialSmoothing

In [222]: 1 df_arrec['DES'] = ExponentialSmoothing(df_arrec['VALOR (R$)'], trend='add').fit().fittedvalues.shift(-1)

In [227]: 1 ax = df_arrec[['VALOR (R$)', 'SMA12', 'EWMA12', 'DES']].iloc[:48].plot(figsize=(12,6))
          2 ax.set(xlabel='');

```



5.4 Suavização Exponencial Tripla (TES)

A suavização exponencial tripla, também conhecida como método Holt-Winters, é uma extensão da suavização exponencial dupla que leva em consideração tanto a tendência quanto a sazonalidade nos dados de séries temporais. Envolve a adição de um terceiro fator de suavização, conhecido como γ , ao modelo. O fator γ ajusta a sazonalidade dos dados e ajuda a prever os padrões sazonais futuros. O método é particularmente útil para prever dados de vendas, preços de ações e outros dados de séries temporais que exibem tanto tendências quanto padrões sazonais. A suavização exponencial tripla também pode ser adaptada para lidar com vários períodos sazonais, tornando-se uma ferramenta poderosa para analisar e prever dados de séries temporais complexos. (Hyndman)

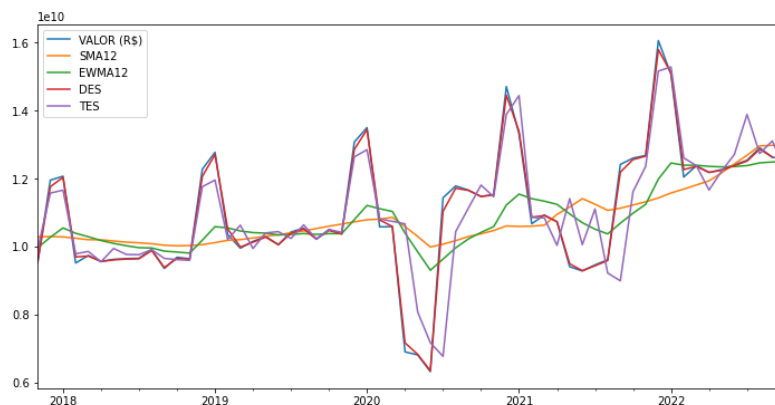
Suavização Exponencial Tripla (TES)

```
In [276]: 1 df_arrec['TES'] = ExponentialSmoothing(df_arrec['VALOR (R$)'], trend='add', seasonal='add', seasonal_periods=12).fit()
          2 df_arrec.head()
```

```
Out[276]:
```

	VALOR (R\$)	SMA12	EWMA12	DES	TES
Mês/Ano					
1997-01-01	1.355415e+09	NaN	1.355415e+09	1.324798e+09	1.360885e+09
1997-02-01	1.012064e+09	NaN	1.302592e+09	1.022873e+09	1.028044e+09
1997-03-01	9.969265e+08	NaN	1.255566e+09	9.860185e+08	1.141224e+09
1997-04-01	1.013695e+09	NaN	1.218355e+09	9.987292e+08	9.623540e+08
1997-05-01	1.021931e+09	NaN	1.188136e+09	1.007307e+09	1.068124e+09

```
In [277]: 1 ax = df_arrec[['VALOR (R$)', 'SMA12', 'EWMA12', 'DES', 'TES']].iloc[-60:].plot(figsize=(12,6))
          2 ax.set(xlabel='');
```



Comparando os três modelos através das métricas de desempenho o modelo de Suavização Exponencial Dupla (DES) apresetou o menor erro.

MSE (Mean Squared Error):

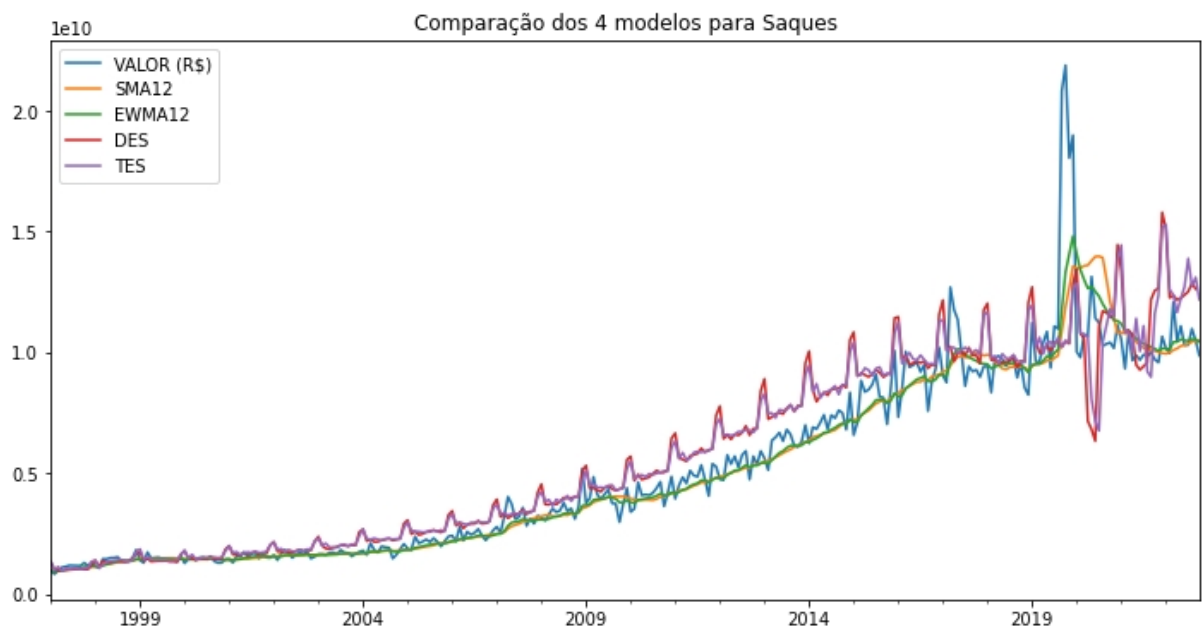
```
MSE SMA:      757576.32 tri
MSE EWMA:     600655.87 tri
MSE DES:      4090.57 tri
MSE TES:      267480.31 tri
```

RMSE (Root Mean Squared Error):

```
RMSE SMA:      870.4 mi
RMSE EWMA:     775.0 mi
RMSE DES:      64.0 mi
RMSE TES:      517.2 mi
```

Com base nos gráficos acima, pode-se pensar que a suavização exponencial tripla faz um trabalho pior de ajuste do que a suavização exponencial dupla. No entanto, é importante considerar as previsões. Veremos que ter a capacidade de prever padrões sazonais flutuantes melhora muito nossa previsão.

Os mesmos testes foram efetuados para os Saques e o modelo que teve o melhor desempenho foi o RMSA.



MSE (Mean Squared Error):

MSE SMA: 1448919.0 tri
MSE EWMA: 1058354.8 tri
 MSE DES: 2846636.4 tri
 MSE TES: 2694734.2 tri

RMSE (Root Mean Squared Error):

RMSE SMA: 1.204e+03 mi
RMSE EWMA: 1.029e+03 mi
 RMSE DES: 1.687e+03 mi
 RMSE TES: 1.642e+03 mi

6. Previsões (Forecasting)

Foram ajustados vários modelos de suavização aos dados existentes. O objetivo por trás disso é prever o que acontece a seguir.

Qual é a nossa melhor estimativa para o valor do próximo mês? Para os próximos seis meses?

Vamos procurar estender nossos modelos para o futuro. Primeiro, dividiremos os dados conhecidos em conjuntos de treinamento e teste e avaliaremos o desempenho de um modelo treinado nos dados de teste conhecidos.

6.1 Base de testes

.6. Interpretação dos Resultados

Nessa seção você deve interpretar os resultados obtidos na análise e exploração de dados e também interpretar os resultados da aplicação dos algoritmos de Machine Learning, descobrindo insights importantes para responder o problema proposto.

.7. Apresentação dos Resultados

Nessa seção você deve apresentar os resultados obtidos. Apresente gráficos, *dashboards*, conte a sua história de forma bastante criativa. Aqui você pode utilizar os modelos de Canvas propostos por Dourard (clique [aqui](#)) ou por Vasandani (clique [aqui](#)).

.8. Links

Aqui você deve disponibilizar os links para o vídeo com sua apresentação de 5 minutos e para o repositório contendo os dados utilizados no projeto, scripts criados, etc.

Link para o site do FGTS: <https://www.fgts.gov.br/>

Link para o vídeo: [youtube.com/...](youtube.com/)

Link para o repositório: [github.com/...](github.com/)

REFERÊNCIAS

MORETTIN, Pedro A.; TOLOI, Clélia M. C. **Análise de séries temporais**. Blucher, 2006.

MONTGOMERY, Douglas C.; JONHSON, Llyod R.; GARDNER, Emily A. **Forecasting and time series analysis**. Springer, 2015.

OLIVEIRA, F. F. **Análise de Séries Temporais Financeiras com a Plataforma Economatica. Tese (Doutorado em Engenharia Elétrica) - Escola Politécnica, Universidade de São Paulo, São Paulo, 2019.**

MÉDIA MÓVEL. In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2022. Disponível em: <https://pt.wikipedia.org/wiki/M%C3%A9dia_m%C3%B3vel#M%C3%A9dia_m%C3%B3vel_simples>. Acesso em: 9 dez. 2022.

HUANG, D. X.; ZHOU, X. D.; ZHANG, Y. F. Financial time series forecasting using support vector machines. *Neurocomputing*, v. 51, p. 321-339, 2003.

Jain, M., & Kumar, P. (2021). Double Exponential Smoothing: A Comprehensive Study. *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*, 10(3), 59-63.

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: principles and practice*. OTexts.

SOBRENOME DO AUTOR, Nome do autor. **Título do livro ou artigo**. Cidade: Editora, ano.

.APÊNDICE

Programação/Scripts

Cole aqui seus scripts em Python e/ou R.

Gráficos

Cole aqui workflows (KNIME), gráficos e figuras que você tenha gerado e não colocou no texto principal.

Tabelas

Cole aqui tabelas de dados que você tenha gerado e não colocou no texto principal.