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

Igor Ribeiro Braga

Séries Temporais de Receitas Orçamentárias da Prefeitura Municipal de João Pessoa e Predição de Valores Orçamentários

Belo Horizonte
2019

Igor Ribeiro Braga

Séries Temporais de Receitas Orçamentárias da Prefeitura Municipal de João Pessoa e Predição de Valores Orçamentários

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.

Belo Horizonte
2019

SUMÁRIO

1. Introdução

1.1. Contextualização

O acompanhamento da evolução de receitas na esfera pública é de interesse de toda a população. Este trabalho tem como intuito fazer uma análise sobre os valores recebidos pela Prefeitura do Município de João Pessoa no período de 2003 a 2019. A análise será feita utilizando uma visão estatística utilizando séries temporais e posteriormente será apresentado uma proposta de modelo para predição de valores orçamentários utilizando técnicas aprendidas durante o curso.

1.2. O problema proposto

A motivação desta análise é expor como o orçamento de uma prefeitura municipal pode ser composta e quais os valores são recebidos pela entidade. Neste caso a entidade escolhida foi a Prefeitura Municipal de João Pessoa, capital do estado da Paraíba. Além da análise dos dados, será feito uma análise preditiva utilizando uma série temporal das receitas orçamentárias baseados nos dados obtidos pelo Tribunal de Contas do Estado da Paraíba.

Proponho analisar os dados respondendo a algumas perguntas, como por exemplo: Qual foi o ano em que o município recebeu mais receitas? Qual mês se destacou mais no recebimento das receitas? Houve meses em que o recebimento das receitas foram consideradas anormais? Quais seriam os prováveis motivos? Respondendo a perguntas assim, podemos nos aprofundar mais no *dataset* escolhido.

Na próxima etapa será feito uma análise da série temporal das receitas do orçamento da prefeitura e a predição utilizando as estratégias estudadas no curso aplicando o modelo ARIMA (Autoregressive Integrated Moving Average).

2. Coleta de Dados

A fonte de dados utilizado foi adquirido pelo site do Tribunal de Contas do Estado da Paraíba, gerado pelo sistema SAGRES[1].

O *dataset*[2] de Receitas Orçamentárias da Esfera Municipal possui todos os recebimentos de entidades públicas do estado da Paraíba desde o mês de Janeiro de 2003 até o mês atual, sendo atualizado mensalmente pelo Tribunal. Para este estudo, foi filtrado apenas os dados referentes a Prefeitura Municipal de João Pessoa, direcionando assim, o estudo para uma entidade. A descrição dos dados é disponibilizada no site do TCE-PB e pode ser conferida também abaixo.

Nome da coluna/campo	**Descrição**	**Tipo**
cd_ugestora	Identificador da unidade gestora.	Numérico inteiro
de_ugestora	Nome da unidade gestora.	Texto
dt_ano	Ano.	Numérico inteiro
cd_receitaorcug	Código da receita/unidade gestora.	Texto
de_receitaorcug	Descrição da receita.	Texto
tp_atualizacaoreceita	Código para a atualização/lançamento da receita.	Numérico inteiro
de_atualizacaoreceita	Descrição da atualização	Texto
vl_lancamentoorc	Valor do lançamento, sendo o ponto(.) o separador decimal	Numérico monetário
dt_mesano	mês e ano, na forma MMYYYY	Texto

Tabela 1: Descrição das colunas presentes do dataset

3. Processamento/Tratamento de Dados

O *dataset* original, contendo todas as entidades públicas, possui **1.432.689 registros** com dados de receitas recebidas pelas entidades do ano de 2003 até Julho de 2019, mês de estudo deste trabalho. Alguns passos serão demonstrados neste documento, mas todo o código está disponibilizado em um Jupyter Notebook, onde toda a análise foi realizada. Para ter acesso ao Notebook, acesso o link disponibilizado na seção 7 deste documento.

```
# Definindo tipos das colunas antes de ler o dataset

columns_types = {'cd_ugestora': str, 'dt_mesano': str, 'dt_ano': int}

# Carrega os dados usando pandas

receitas = pd.read_csv('datasets/TCE-PB-SAGRES-Receita_Orcamentaria_Esfera_Munici-
pal.txt',sep='|',dtype=columns_types,encoding='utf-8')
```

Tabela 2: Leitura do dataset utilizando Pandas

Para que possamos focar apenas na Prefeitura Municipal de João Pessoa, é utilizado um filtro informando o valor da unidade gestora, no caso sendo o valor da ***cd_unidade_gestora*** igual a **201095**. Assim, nosso *dataset* fica com **7.749 registros**.

```
# Obtém todos os dados da Prefeitura Municipal de João Pessoa (cd_ugestora=201095)

pref_mun_joao_pessoa = receitas[receitas['cd_ugestora'] == '201095']
```

Tabela 3: Filtro no dataset obtendo apenas os dados da unidade gestora de interesse

Alguns registros demonstraram conter dados nulos na coluna de**_receitaorcug**. Essa coluna é uma descrição do tipo de receita recebido pela entidade. Decidi preencher esses dados com o texto ‘Sem descrição’, substituindo-os em todos os registros.

```
# Preenchendo os valores nulos

pref_mun_joao_pessoa.de_receitaorcug.fillna('Sem descrição', inplace=True)
```

Tabela 4: Substituição dos valores nulos

A outra verificação foi validar se as informações de mês, na coluna **dt_mesano**, estavam de acordo com o padrão, pois em análise exploratória notei que algumas entradas para outras entidades possuíam, o campo **dt_mesano** valores como *132011*, como a entrada exemplo abaixo:

```
201016|Prefeitura Municipal de Areial|2011|19909900|OUTRAS RECEITAS|2|Estorno de lançamento de receita|4.83|132011
```

Tabela 5: Exemplo de informação de data incoerente

Para me certificar que os dados que possuo não apresentam problemas, separei o mês do ano desta coluna com o comando abaixo:

```
# Certificando se possuem meses incoerentes

pref_mun_joao_pessoa.dt_mesano.str.slice(0,2).value_counts()

07    683
12    673
06    671
05    660
04    651
08    638
10    637
01    636
03    635
11    630
09    626
02    609
Name: dt_mesano, dtype: int64
```

Tabela 6: Validando os valores dos meses da coluna dt_mesano

Para a entidade escolhida não há registros com esse tipo de problema como evidenciado acima.

4. Análise e Exploração dos Dados

Inicialmente para a exploração dos dados, sugiro responder a perguntas provenientes da natureza do *dataset*. Quais os tipos de receitas que mais aparecem? Quais os anos que mais geraram receita? Quais foram os meses? Houve meses que se apresentam como *outliers*?

Iniciando pelos tipos de receitas que mais aparecem no *dataset*, utilizei um método personalizado que exibe um gráfico que utiliza o método **value_counts** do Pandas. Assim, abaixo estão os 5 tipos de receitas que mais se repetem no dataset:

Figura 1: 5 tipos de receitas que mais aparecem no dataset

Como mostrado, acima estão alguns impostos bem conhecidos como Cota do IPVA (Imposto sobre a propriedade de Veículos Automotores), IPTU (Imposto Predial e Territorial Urbano) e cota de ITR (Imposto Territorial Rural) estariam bem presentes, sendo os principais contribuidores das receitas dos municípios assim como aportes da União.

Realizando agrupamento com a soma das receitas por ano, obtive um gráfico como abaixo com os totais anuais das receitas:

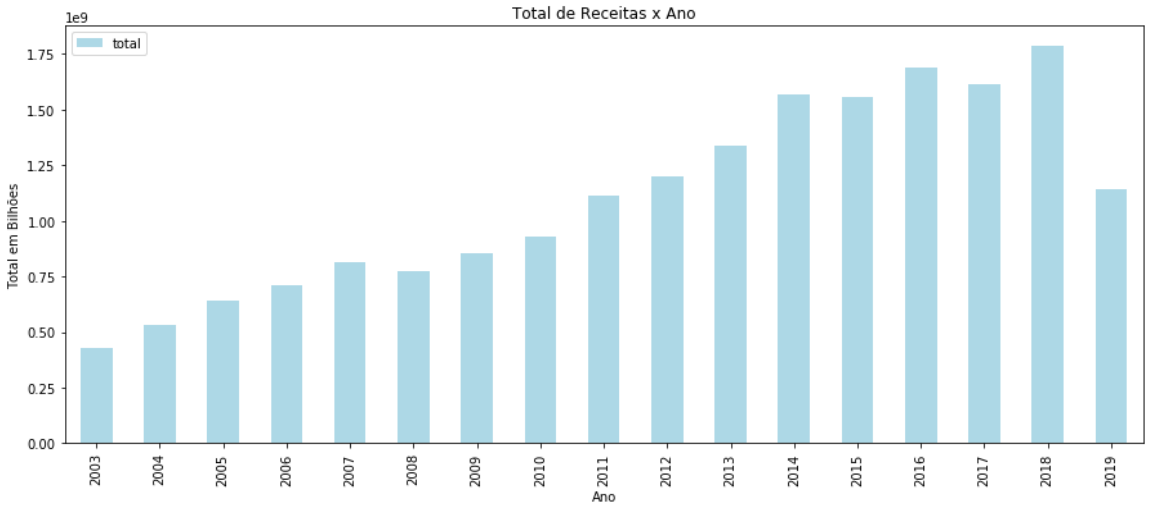

Figura 2: Totais das Receitas por ano

O gráfico se comporta numa crescente, em que o ano de 2018 fica em destaque logo seguido do ano de 2016 e 2014. O ano de 2019 está bem abaixo pois esse estudo foi realizado em Julho de 2019 e os dados para esse ano ainda serão gerados.

O valor total recebido em 2018 foi **R$ 1.787.107.723,33**, quase 1,8 bilhão de reais.

4.2 Outliers

Irei agora analisar o *dataset* com a finalidade de identificar possíveis valores muito distantes dos demais dados. Esses tipos de dados são considerados *outliers* e atrapalham a criação de modelos de predição de valores.

Um gráfico bloxplot com esse agrupamento mostra que os dados se comportam dentro de um intervalo não considerado *outlier*.

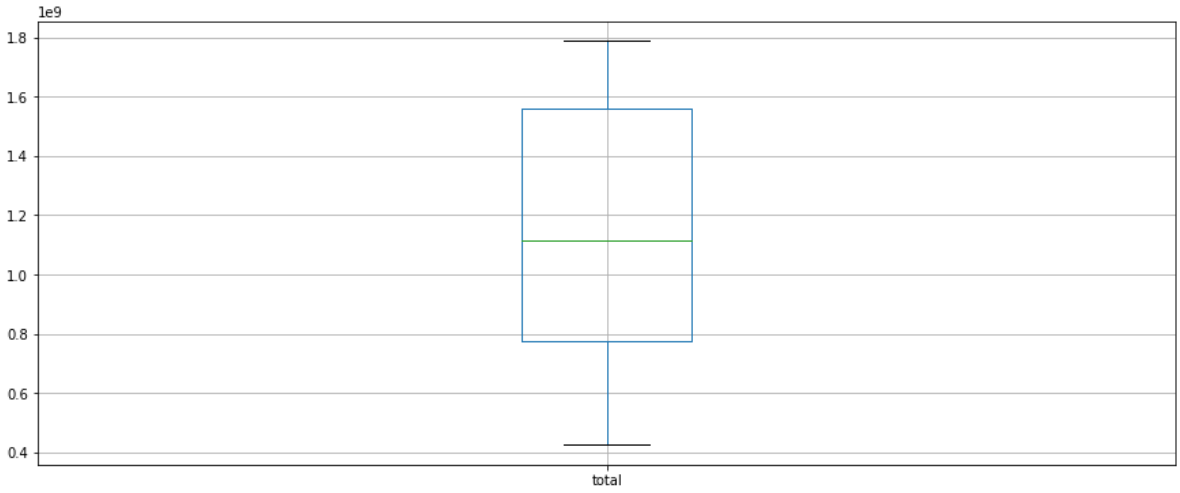

Figura 3: Boxplot dos totais recebidos por ano

Mas como seria o comportamento dos dados se o agrupamento desses valores fosse por mês, em vez de anos?

Mudando o agrupamento dessa forma teremos muito mais pontos a serem informados no gráfico em comparação ao agrupamento em anos. Assim, o próximo gráfico será exibido utilizando linhas em vez de barras.

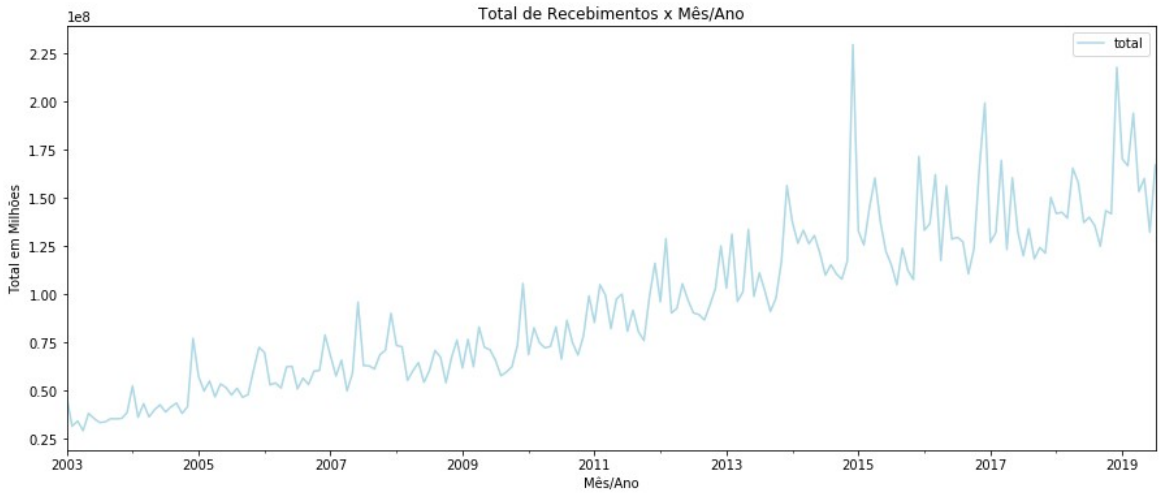

Figura 4: Totais das Receitas por mês

Aqui fica mais nítido que ocorrem diferenças mais abruptas nas receitas recebidas por mês. Pode ser observado, por exemplo, que o final de 2014 teve um valor bem diferente dos demais, chegando a ser um pouco maior que o maior valor de 2018. Vamos observar o gráfico bloxplot do *dataset* na visão das receitas por mês.

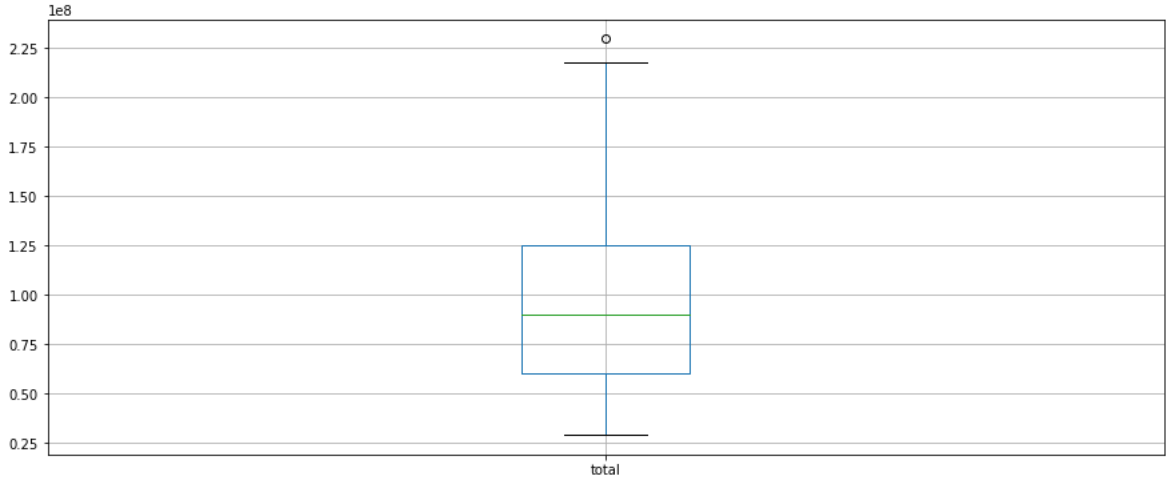

Figura 5: Aparecimento de valor considerado outlier no dataset das receitas por mês

Agora, um valor que se mostra fora dos limites é mostrado. Inicialmente é considerado um valor de *outlier* em relação aos demais do dataset, todos eles desde Janeiro de 2003 até Julho de 2019. Ao identificar o valor vi que se trata dos recebimento de dezembro de 2014, um total no valor de **R$ 229.477.921,05**.

Mas será que este é o único valor que está fora dos limites? Como esse gráfico se comportaria se analisarmos os dados contidos em um único ano individual-

mente? Dentro de um mesmo ano não existem mais valores fora dos intervalos de máximo e mínimo? Abaixo este gráfico é mostrado.

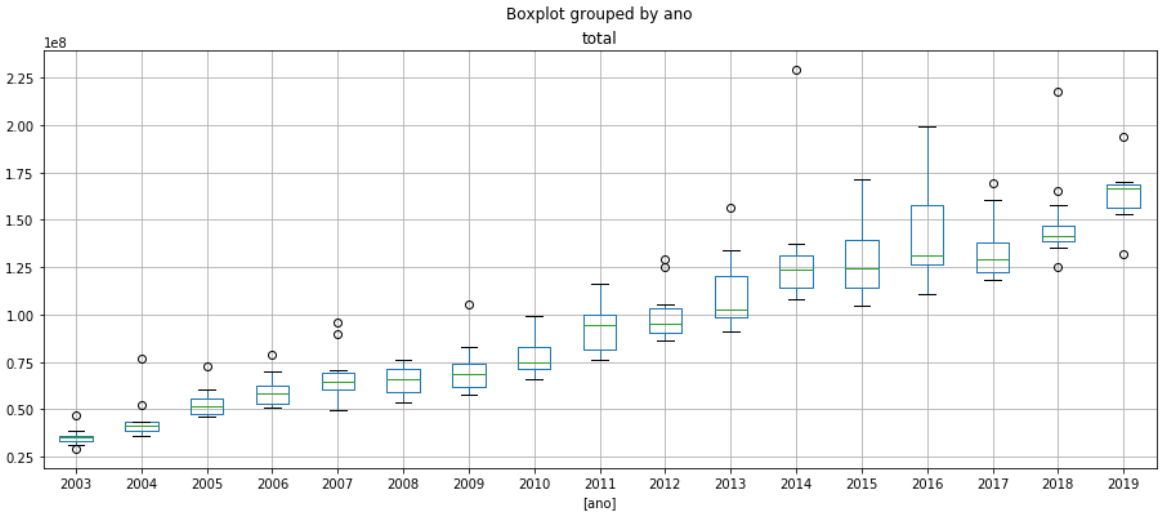

Figura 6: Bloxplot dos dados contidos e agrupados por ano

De fato vários outros pontos são identificados. Este gráfico mostra que ao analisarmos os dados agrupados por ano individualmente podemos identificar mais valores que estão fora dos valores de máximo e mínimo da distribuição autocontido no ano. Representa também como esses valores oscilam vertiginosamente dentro dos anos.

Mas como será essa distribuição visualizando os meses dentro dos anos?

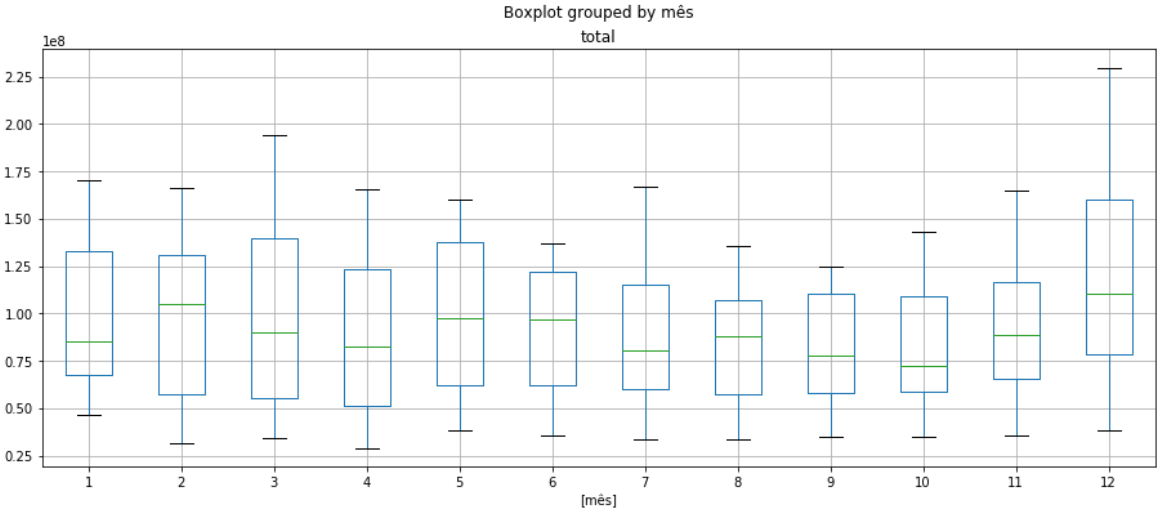

Figura 7: Visualizando a distribuição das receitas dentro de cada mês com passar dos anos

A leitura que podemos ter do gráfico é que, dentro de cada mês no passar dos anos, os valores oscilam e possuem comportamento aceitável, tendo valores de

quartis, de máximas e mínimas variadas. Acredito que isso se explique pela natureza do *dataset*. Alguns impostos que impactam nas receitas do município como IPVA e IPTU possuem calendários e formas de pagamentos flexíveis, o que pode acontecer que em determinado ano a maioria das pessoas escolheram vencimentos ou formas de pagamentos que se acumularam em determinados meses que afetam o comportamento da série. Ou recebimentos de receitas com valores impactantes venham de forma assíncrona, arbitrária, como foi o caso de recebimento de royalties relativos ao pré-sal.

Como a análise da série é levando em consideração os dados mensais dos totais por mês, vou considerar a figura 5, onde apenas um dos valores foi apontado como *outlier* relativo aos demais dados. Para o tratamento do dado e com o intuito de não impactar os valores da série, vou utilizar a média do mês de Dezembro do ano de 2013 (anterior) e do ano de 2015 (posterior). Assim, o valor de Dezembro de 2014 passará de **R$ 229.477.921,05** para **R$ 163.870.833,32**.

Assumindo esses valores como os aceitáveis para continuar a análise da série podemos cobrir as características que faltam para realizar a predição dos dados.

4.3 Tendência, Autocorrelação e Médias Móveis

Para a criação do modelo preditivo e validar a predição a ser realizada o *dataset* utilizado foi divido em 2 partes. Uma base de treino com 80% dos valores e uma base de teste com 20% restantes dos dados. Ao final, o gráfico com a predição terá os dados da base de teste para comparativo com os resultados obtidos.

Para que seja possível criar modelos preditivos, temos que garantir que a série seja considerada estacionária. Para isso temos que ter 3 elementos constantes, a média, a variância e a autocorrelação. A primeira verificação que devemos fazer é checar se a série possui tendência.

4.3.1 Tendência

Para identificar se a série possui tendência, podemos usar um gráfico chamado **seasonal_decompose** presente no **statsmodels**.

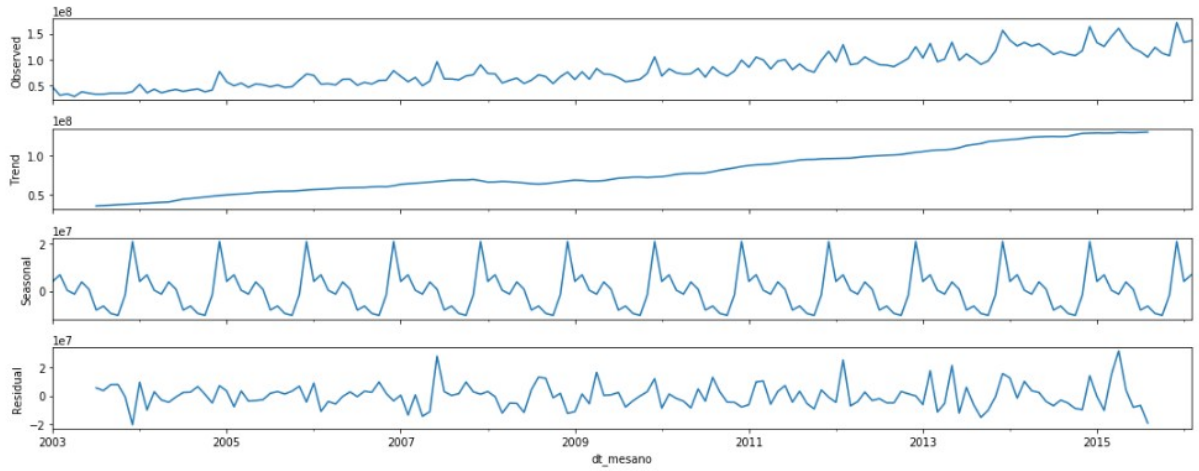

Figura 8: Confirmando o Trend (tendência) da série

Como pode-se confirmar olhando o gráfico *Trend*, a série possui uma tendência crescente, assim torna-se necessário aplicar técnicas para tentar anular essa tendência. Existem testes que podem ser aplicados para comprovar que, estatisticamente, a série é ou não estacionária. Aqui usei o teste de *Augmented Dickey-Fuller Test*, aqui caso o *p-value* calculado seja abaixo de 5%, pode se considerar que a série pode ser considerada estacionária.

Utilizando a série original de treinamento temos o resultado abaixo:

```
ADF Statistic: 0.670713
p-value: 0.989235
Critical Values:
        1%: -3.476
        5%: -2.882
        10%: -2.577
```

Tabela 7: Teste ADF da série original

Como o valor de *p* foi de 98% está comprovado que a série é não estacionária. Com o intuito de corrigir esta tendência apliquei uma diferenciação a série. Criando novamente o gráfico, podemos ver como fica a tendência da nova série.

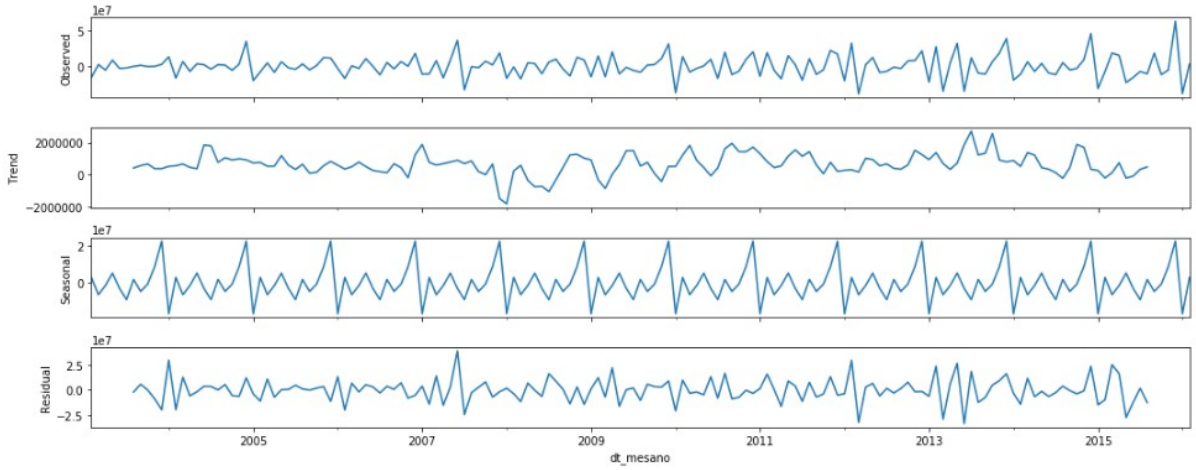

Figura 9: Tendência da série reduzida aplicando uma diferenciação

Re-executando o teste de Dickey-Fuller na série com uma diferenciação temos o seguinte valor de *p*.

```
ADF Statistic: -13.588397
p-value: 0.000000
Critical Values:
        1%: -3.476
        5%: -2.882
        10%: -2.577
```

Tabela 8: Teste ADF da série com 1 diferenciação

Com o valor de *p* sendo 0 posso considerar que uma diferenciação foi suficiente para estacionar a série.

4.3.2 Autocorrelação

A autocorrelação nos informará qual o grau de relação que cada dado tem com os demais dados passados da série. Isso ajudará a definir o valor *p* para ser utilizado no modelo preditivo.

Para isso, utilizarei o gráfico ACF (Autocorrelation Factor), que irá nos mostrar o número de *lags* que estão fora do intervalo de confiança. Esse valor será o nosso AR.

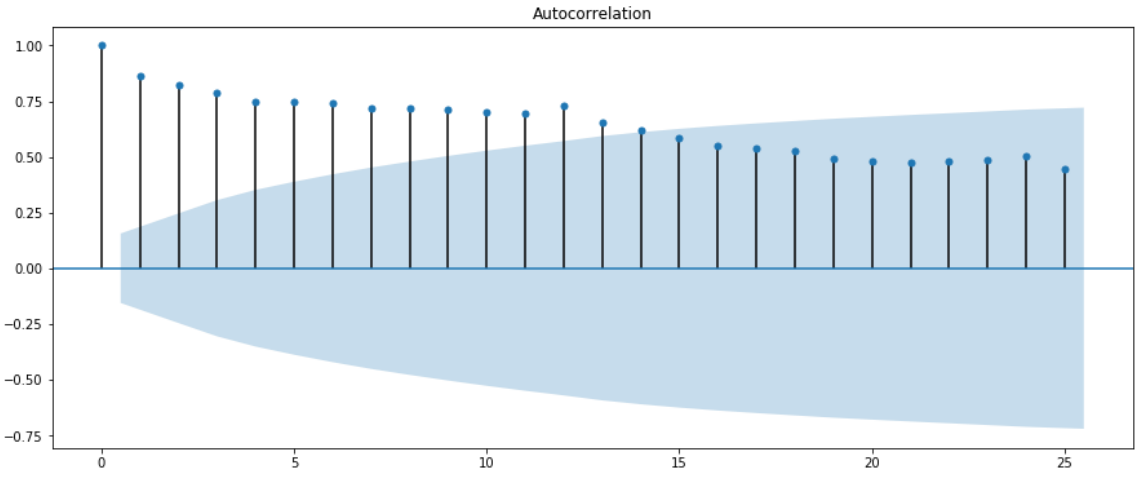

Figura 10: Gráfico ACF da série

Como o gráfico mostra, os valores ficam contidos no intervalo a partir do *lag* 15, então os valores a serem utilizados estará perto do valor *14*. Após alguns testes e verificando o AIC e BIC do modelo, decidi usar o valor *11*.

4.3.3 Médias Móveis

Para obter o valor das médias móveis e termos uma pista sobre o valor *q*, utilizaremos o gráfico PACF (Partial Autocorrelation Factor).

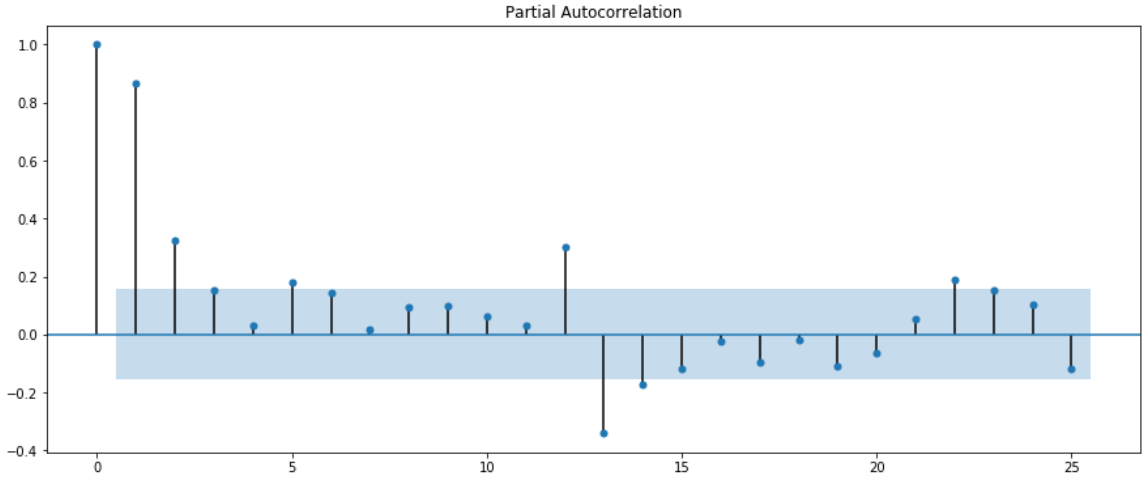

Figura 11: Gráfico PACF da série

Vemos aqui eu os três primeiros valores têm uma significância correlativa alta em relação aos demais. Então usarei o valor 3 inicialmente para definir o valor *q* do modelo preditivo.

5. Modelo de Predição Orçamentária

Definido os valores AR(*p), I(d)* e MA(*q)* para nosso modelo ARIMA, vamos instanciar o modelo e mostrar no gráfico como ficou a predição em relação aos dados de teste definidos anteriormente.

5.1 Criação do modelo ARIMA

No nosso cenário, o modelo ficou com os seguintes parâmetros:

```
# Criando o modelo

modelo = ARIMA(ts_treino, order=(11,1,3), freq=ts_treino.index.inferred_freq)
```

Tabela 9: Parâmetros do modelo ARIMA

Ajustando o modelo, obtive o seguinte resultado para os valores de AIC e BIC:

ARIMA Model Results

Dep. Variable:	D.total	**No. Observations:**	157
Model:	ARIMA(11, 1, 3)	**Log Likelihood**	-2753.222
Method:	css-mle	**S.D. of innovations**	9748544.576
Date:	Tue, 15 Oct 2019	**AIC**	5538.444
Time:	20:26:24	**BIC**	5587.343
Sample:	02-01-2003	**HQIC**	5558.304
	- 02-01-2016		

Figura 12: Resultados do modelo ARIMA(11,1,3) definido pelo estudo

Com o modelo ajustado, podemos obter informações sobre os resíduos do modelo. Assim, podemos averiguar o comportamento dos residuais e sua densidade.

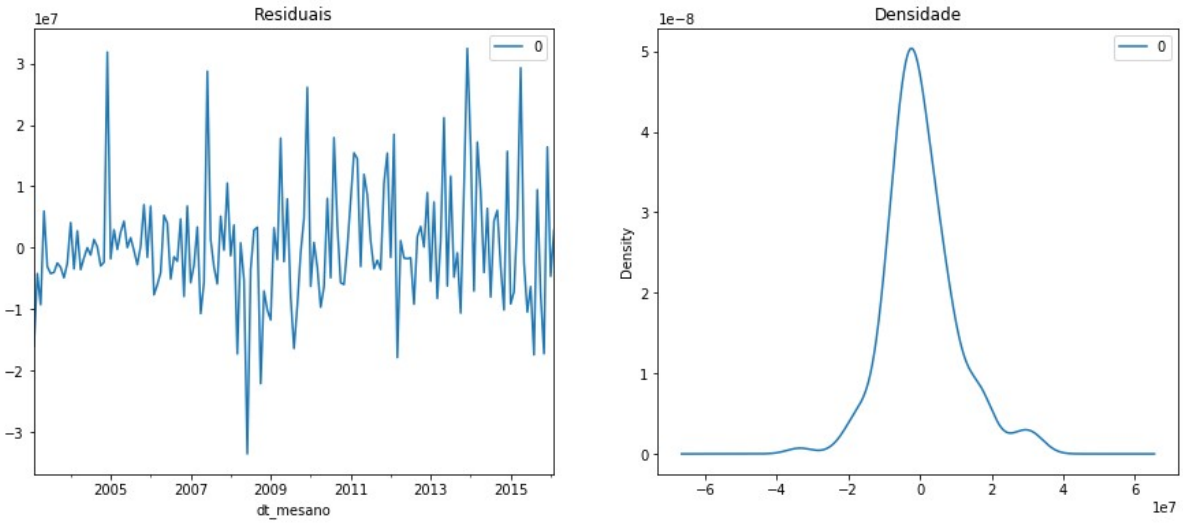

Figura 13: Dados residuais e sua densidade

Os resíduos ficam em torno de 0 com uma variação satisfatória, com os pontos tendendo a permanecer com média em torno de 0. E a densidade dos dados se comporta como uma distribuição normal.

Agora podemos ver como os dados do modelo e os dados reais de treino exibidos em um mesmo gráfico nos dará uma ideia do comportamento do nosso modelo.

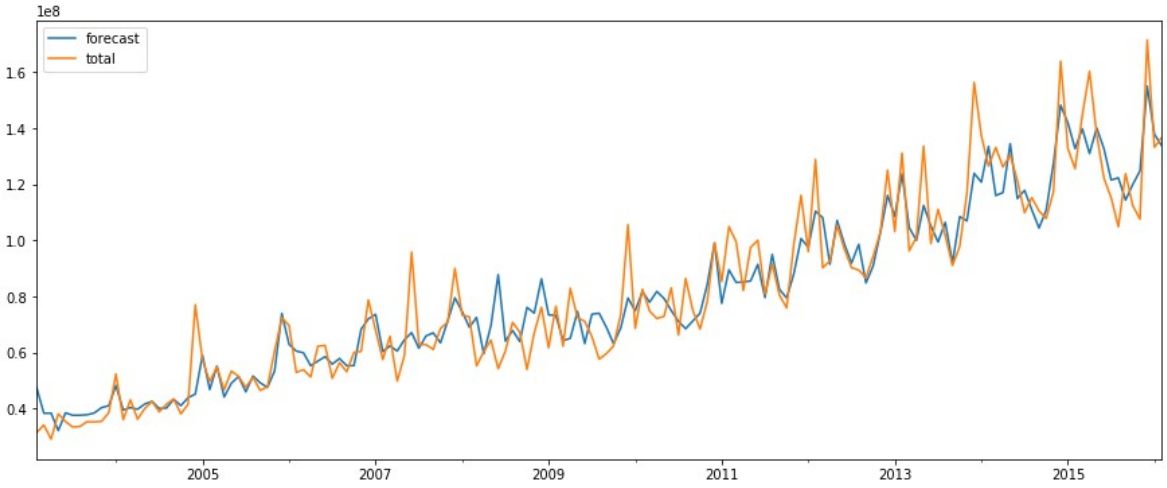

Figura 14: Dados reais x modelo ajustado

Podemos notar aqui que os dados do modelo se comporta de maneira similar aos dados reais, todas as variações de alta de baixa dos dados reais são acompanhadas pelo modelo, não assumindo exatamente os mesmos valores de pico e queda, mas sendo suficiente para ser utilizado de objeto deste estudo.

5.2 Predição

Após todo o estudo, podemos agora realizar a predição dos dados em comparação aos dados da base de teste, validando assim nosso modelo.

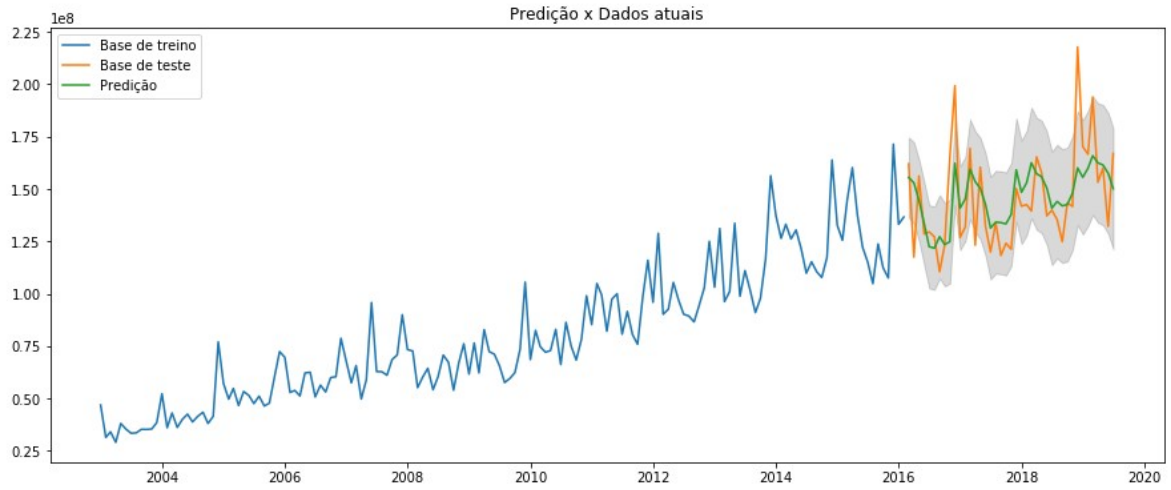

Figura 15: Predição x Dados atuais

Os dados obtidos neste gráfico mostra que a predição utilizando o modelo ARIMA(11,1,3) exibe pontos bem próximos dos reais, com exceção dos picos que estão presentes fora do intervalo de confiança do modelo, o comportamento da curva acompanha os dados reais da base de teste.

6. Conclusão

O estudo se séries temporais é um dos assuntos que mais me despertou interesse no curso de Big Data e Ciência de Dados, e a medida em que me aprofundava no assunto mais me despertava a importância da matemática estatística nesse universo. Me mostrou o quanto este mundo e o da informática devem e tendem a se fundir para atender as necessidades atuais sobre o estudo analítico dos dados gerados diariamente pelas pessoas e pelos sistemas.

Sobre este estudo, todas as etapas, desde a obtenção dos dados, o tratamento das informações e a análise do *dataset* para que possamos gerar informação, tem uma profunda importância nos passos futuros. A fase em que mais me fascinou foi a de análise da série e como obter os parâmetros para a predição de dados futuros de acordo com o modelo. Nesta etapa é onde se destacam os profissionais matemáticos que conseguem analisar os gráficos levando em consideração sua natureza e seu comportamento em uma linha do tempo. A análise não acontece friamente baseado-se apenas em números, mas sim em o que esses números representam em um ambiente computacionalmente vivo e evolutivo.

Como pontos de melhoria no estudo eu acredito que um tratamento mais minucioso nos **outliers** podem melhorar o desempenho do modelo assim como experimentar outros tipos de modelo além do ARIMA. Uma dificuldade que eu entendo que passei foi devido à natureza do *dataset*. Diferentemente de um *dataset* que poderia ter dados de temperaturas de um lugar onde a variância é conhecida e a identificação de **outliers** derivados de medições problemáticas ou defeitos de equipamentos, os dados de receitas orçamentárias de um município que são validadas por um órgão de competência como o Tribunal de Contas do Estado podem sofrer oscilações bem consideráveis, mas certamente justificáveis, como o recebimento de royalties do pré-sal ou aportes da União para resolver questões momentâneas, e para isso cabe um estudo histórico mais aprofundado e detalhado para definir o que realmente é **outlier** e como tratá-los.

7. Links

Documento do TCC: https://github.com/irbraga/data-science/tree/master/tcc-puc-minas/report

Jupyter Notebook: https://github.com/irbraga/data-science/tree/master/tcc-puc-minas/notebooks

Vídeo da apresentação: http://www.youtube.com

REFERÊNCIAS

[1] Tribunal de Contas do Estado da Paraíba http://tce.pb.gov.br/servicos/dados-abertos-do-sagres-tce-pb

[2] Link para o dataset: https://dados.tce.pb.gov.br/TCE-PB-SAGRES-Receita_Orcamentaria_Esfera_Municipal.txt.gz