PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
NÚCLEO DE EDUCAÇÃO A DISTÂNCIA
Pós-graduação *Lato Sensu* em Inteligência Artificial e Aprendizado de Máquina

Thalita de Melo Soares

UTILIZAÇÃO DE ALGORTIMOS DE MACHINE LEARNING PARA PREVISÃO DO ÍNDICE IPCA

Belo Horizonte
2022

Thalita de Melo Soares

UTILIZAÇÃO DE ALGORTIMOS DE MACHINE LEARNING PARA PREVISÃO DO ÍNDICE IPCA

Trabalho de Conclusão de Curso apresentado ao Curso de Especialização em Inteligência Artificial e Aprendizado de Máquina como requisito parcial à obtenção do título de especialista.

Belo Horizonte
2022

ÍNDICE DE FIGURAS

SUMÁRIO

1. Introdução

1.1. Contextualização

A predição de índices econômicos é de suma importância para o país, porque é por meio deles que o governo pode se preparar e traçar planos para o futuro da nação. Dessa forma, um índice muito importante para a economia é o Índice Nacional de Preços ao Consumidor Amplo (IPCA), pois com ele é possível calcular a inflação mensal do país.

De acordo com o (IBGE, 2022), o objetivo do IPCA é “medir a inflação de um conjunto de produtos e serviços comercializados no varejo, referentes ao consumo pessoal das famílias, cujo rendimento varia entre 1 e 40 salários mínimos, qualquer que seja a fonte de rendimentos.”

Com a alta dos preços e as crises globais dos últimos anos, ficou cada vez mais difícil prever essas importantes variações na economia. Sandroni (1999) explica que a inflação nada mais é do que um aumento de preços, que resulta em uma perda de poder aquisitivo da moeda.

Segundo BRITO, JÚNIOR, *et al.*, 2014, a mineração de dados consiste na utilização de algoritmos para extrair certos padrões e prever algum resultado a partir de dados. Dessa forma esse projeto tem o objetivo de utilizar aprendizado de máquina juntamente com mineração de dados para prever o valor do IPCA e por consequência da inflação para os próximos 12 meses.

1.2. O problema proposto

Segundo a CNN (2022), “os brasileiros têm sentido o impacto da inflação desde o começo do ano e isso tem impactado o consumo de alimentos e itens de abastecimento doméstico, de acordo com a pesquisa Radar, da Federação Brasileira de Bancos (FEBRABAN, 2022).”

Além dos preços dos alimentos, vemos aumento nos preços dos combustíveis e gás de cozinha. Sem contar a crise que vem assolando o país há alguns anos. Por isso é importante ter uma previsão dos próximos anos para que o Governo e a população em si possam ter um planejamento financeiro.

Os dados do IPCA são disponibilizados mensalmente pelo próprio governo na página do Instituto Brasileiro de Geografia e Estatística (IBGE).

O objetivo da análise é determinar a eficiência dos modelos de *Machine Learning* ao preverem os dados para a inflação e criar uma predição do IPCA para os próximos 12 meses. A análise abrangerá todo o território nacional. Estarão sendo utilizados dados do período de janeiro de 1994 a agosto de 2022.

2. Coleta de Dados

Os dados foram coletados do site do IBGE, através do link, no dia 07 de setembro de 2022. Os dados se encontram no seguinte padrão:

Nome da coluna/campo	Descrição	Tipo
Ano	Ano referente ao período do IPCA	float64
Mês	Mês referente	object
Índice	Valor do índice para o mês	float64
Variação no mês	Variação comparada ao mês anterio	float64
Variação 3 meses	Variação dos últimos 3 meses	float64
Variação 6 meses	Variação dos últimos 6 meses	object
Variação no ano	Variação no ano	float64
Variação 12 meses	Variação dos últimos 12 meses	float64

3. Processamento/Tratamento de Dados

A linguagem utilizada foi o Python devido a maior quantidade de bibliotecas e modelos já criados para análise de dados. Ao lidar com previsões podemos encontrar problemas como *overforecast,* erro por excesso de dados, ou o *underforecast,* erro de previsão por falta de dados (Serrão, 2003).

Após escolher a linguagem foi necessário escolher um ambiente de desenvolvimento, para tal algumas opções foram analisadas, entre elas podemos citar o *Colaboratory* da Google, e o *Jupyter* dentro do *Anaconda*. O *Jupyter* foi

escolhido por ser uma ferramenta capaz de ser executada sem a necessidade de estar conectado a internet e por possuir interface amigável para o desenvolvimento.

Para utilizar o *Jupyter Notebook* foi utilizado o *Anaconda*, uma distribuição *open-source* do Python e R. Com o Anaconda podemos criar ambientes virtuais (*envs*) de forma fácil e prática, e dessa forma é possível gerenciar o ambiente de maneira mais eficiente, controlando as versões do Python e das bibliotecas de forma independente. Sendo assim o Python que está instalado na máquina pode estar em uma versão e o que for utilizado na *env* em outra. Para utilizar a *env* no *kernell* do *jupyter* é necessário instalar a biblioteca do Jupyter dentro da *env.*

A *env* foi criada com o *Python* na versão 3.8, pois muitas bibliotecas utilizadas funcionam até essa versão, após a criação foi preciso configurar as bibliotecas e instalar as bibliotecas iniciais.

Para a analise inicial precisaremos apenas dos indices referentes ao mês, dessa forma após importar o arquivo csv foi preciso limpar as colunas que não utilizaremos. Outra mudança teve que ocorrer na forma como os meses estão formatados, para a analise o ano e o mês devem estar na mesma célula e em formato de data, por isso foi preciso alterar tal informação.

Por se tratar de uma planilha com uma formatação pensada para o excel, o mesmo foi utilizado para fazer esse pré-processamento dos dados. Para se adaptar as normas padrões do Python as colunas tiveram seus nomes alterados.

ANO	MÊS	NÚMERO ÍNDICE (DEZ 93 = 100)	VARIAÇÃO (%)				
			NO MÊS	3 MESES	6 MESES	NO ANO	12 MESES
2019	JAN	5116,93	0,32	0,26	1,10	0,32	3,78
	FEV	5138,93	0,43	0,90	1,63	0,75	3,89
	MAR	5177,47	0,75	1,51	1,90	1,51	4,58
	ABR	5206,98	0,57	1,76	2,02	2,09	4,94
	MAI	5213,75	0,13	1,46	2,37	2,22	4,66
	JUN	5214,27	0,01	0,71	2,23	2,23	3,37
	JUL	5224,18	0,19	0,33	2,10	2,42	3,22
	AGO	5229,93	0,11	0,31	1,77	2,54	3,43
	SET	5227,84	-0,04	0,26	0,97	2,49	2,89
	OUT	5233,07	0,10	0,17	0,50	2,60	2,54
	NOV	5259,76	0,51	0,57	0,88	3,12	3,27
	DEZ	5320,25	1,15	1,77	2,03	4,31	4,31
2020	JAN	5331,42	0,21	1,88	2,05	0,21	4,19
	FEV	5344,75	0,25	1,62	2,20	0,46	4,01
	MAR	5348,49	0,07	0,53	2,31	0,53	3,30

Figura 1 - Planilha Original

	ANO	MES	INDICE	MES.1	3	6	ANO.1	12
1	1994.0	JAN	141.31	41.31	162.13	533.33	41.31	2693.84
2	1994.0	FEV	198.22	40.27	171.24	568.17	98.22	3035.71
3	1994.0	MAR	282.96	42.75	182.96	602.93	182.96	3417.39
4	1994.0	ABR	403.73	42.68	185.71	648.92	303.73	3828.49
5	1994.0	MAI	581.49	44.03	193.36	695.71	481.49	4331.19
6	1994.0	JUN	857.29	47.43	202.97	757.29	757.29	4922.60
7	1994.0	JUL	915.93	6.84	126.87	548.17	815.93	4005.08
8	1994.0	AGO	932.97	1.86	60.44	370.67	832.97	3044.89
9	1994.0	SET	947.24	1.53	10.49	234.76	847.24	2253.15
10	1994.0	OUT	972.06	2.62	6.13	140.77	872.06	1703.17
11	1994.0	NOV	999.37	2.81	7.12	71.86	899.37	1267.54
12	1994.0	DEZ	1016.46	1.71	7.31	18.57	916.46	916.46
14	1995.0	JAN	1033.74	1.70	6.35	12.86	1.70	631.54

Figura 2 - *Dataframe* após alterações

4. Análise e Exploração dos Dados

Para gerar uma visualização ainda mais detalhada do *dataframe* foi utilizada a biblioteca *pandas-profiling*. Para a previsão do índice IPCA nós precisaremos apenas das colunas ‘ANO, MES, INDICE’. Dessa forma, podemos descartar as outras colunas do *dataframe*.

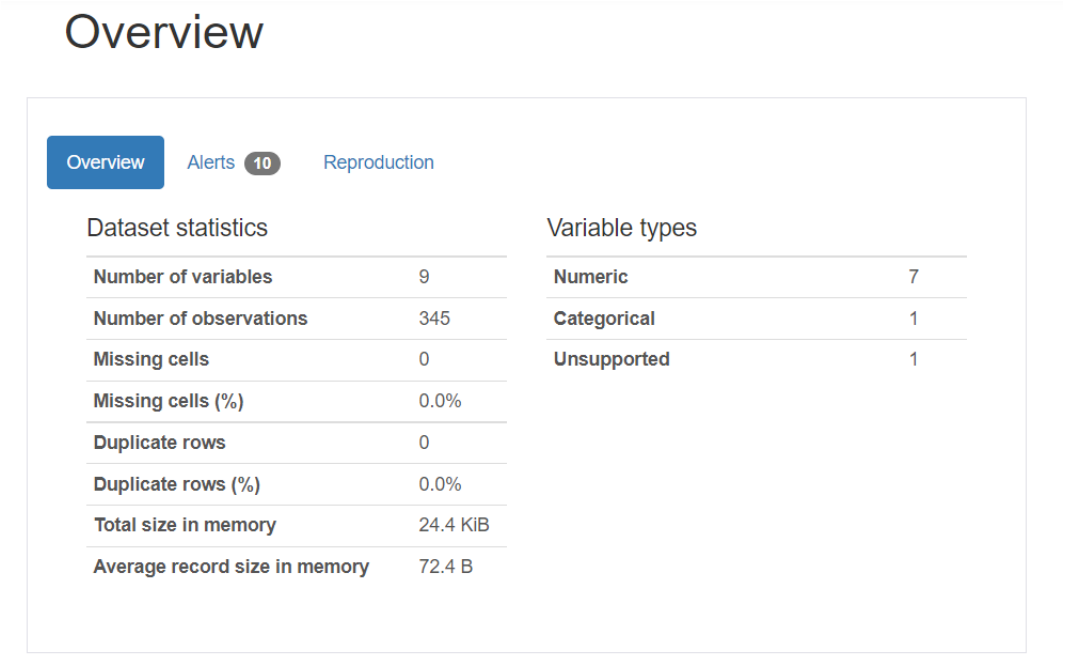

Figura 3 - *Overview* do *dataset*

5. Criação de Modelos de Machine Learning

Para realizar as previsões foi utilizada a biblioteca *prophet*, ela foi desenvolvida pelo Facebook e consistem em um pacote para R e Python que realiza previsão de séries temporais (Taylor e Lethan, 2017). Este pacote pode ser útil para conjuntos onde os dados apresentam um longo período e com forte sazonalidade (SILVA, OLIVEIRA, *et al.*, 2022).

Utilizado o *prophet* e o *statsmodel*, com um modelo linear de regressão foi possível criar uma previsão para os próximos 12 meses. Para utilizar essa biblioteca foi preciso renomear as colunas para ‘ds’ e ‘y’, que é a nomenclatura padrão para o modelo do *prophet*, foi preciso também separar o *dataframe* em teste e treino, para isso foi utilizada a biblioteca *sklearn*.

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, random_state=42)
```

Figura 4 - Exemplo de separação dos dados

Para gerar as previsões foi criado o *dataframe* “df_previsoes” e o resultado da predição do modelo obteve um RMSE de 2097.985, com os seguintes resultados:

	ds	yhat	yhat_lower	yhat_upper
102	2019-12-01	5511.716967	5391.901097	5640.437273
103	2020-08-01	5684.654682	5561.713591	5812.855234
104	2020-09-01	5664.577381	5541.792602	5793.302818
105	2020-11-01	5685.221762	5568.000583	5799.706082
106	2021-01-01	5800.180735	5678.682541	5927.749865
107	2021-03-01	5817.864165	5692.392178	5937.035926
108	2021-06-01	5868.987490	5744.181638	5985.813954
109	2021-10-01	5864.496442	5740.110774	5983.627423
110	2021-11-01	5945.937478	5818.762537	6075.062917
111	2021-12-01	5949.377172	5834.040913	6081.707847
112	2022-02-01	5987.384788	5873.707608	6098.182437
113	2022-07-01	6061.267238	5939.490029	6187.717079

Figura 5 - Resultados do *dataframe* de treino

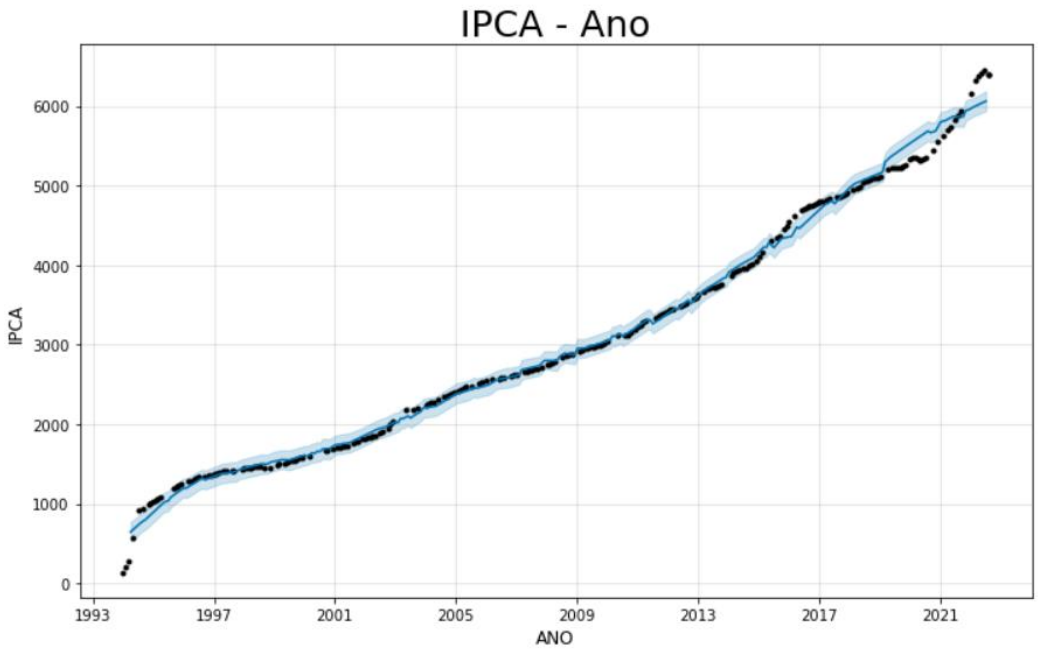

Figura 6 - Gráfico de dispersão

Por fim, para prever o indice para os próximos 12 meses foi utilizado o *make_future_dataframe* do proprio prophet para gerar o dataframe dos próximos 12 meses.

O outro modelo utilizado foi o modelo Auto-regressivo Integrado de médias móveis (ARIMA), juntamente com o módulo *statsmodel.* Segundo Box e Jenkins (1970) o ARIMA é um modelo cuja a escolha é baseada nos próprios dados, em um ciclo iterativo. Ele segiu praticamente os mesmos padrões do *prophet*, com algumas diferenças, no treino e teste. Para esse modelo devemos passar um inicio e um fim para o teste do modelo. O treino utilizando o ARIMA resultou em um RMSE de 64477,361.

```
# ao utilizar dynamic false, o modelo utiliza o ultimo valor previsto para prever o próximo
df['forecast'] = results.predict(start = 90, end= 320, dynamic= False)
df[['INDICE', 'forecast']].plot(figsize=(12, 8))
```

Figura 7 - Treinamento do modelo

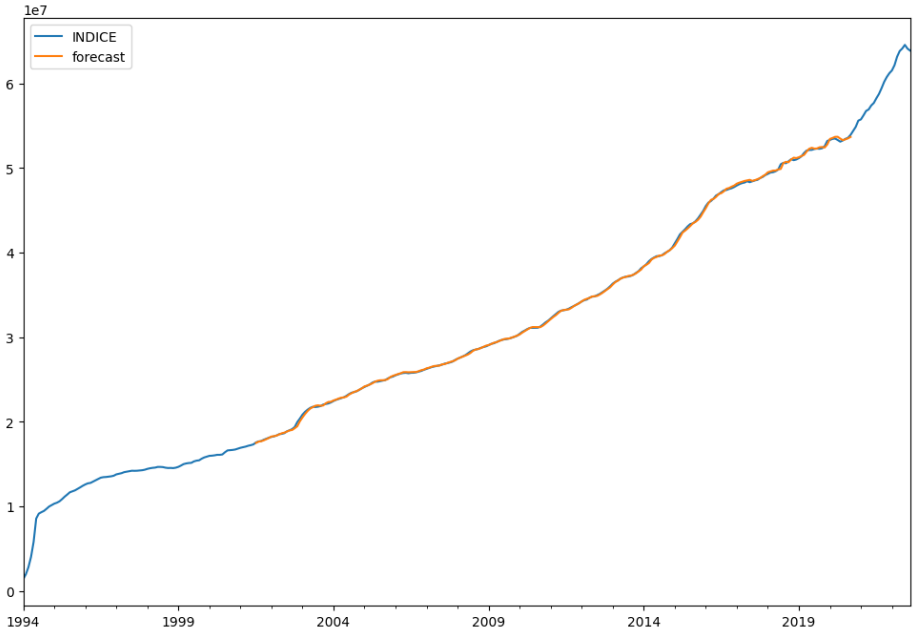

Figura 8 - Resultado do treino

6. Apresentação dos Resultados

Para criar a previsão foi utilizado o método *predict* do Pandas. O *prophet* gerou um *dataframe* onde ds representa as datas, *yhat* é a previsão, *what_lower* é o menor valor previsto e *yhat_upper* é o maior valor previsto.

	ds	yhat	yhat_lower	yhat_upper
344	2022-09-01	6119.359850	5995.906471	6234.380437
345	2022-10-01	6151.268366	6036.218566	6277.126695
346	2022-11-01	6181.011048	6063.027124	6287.093358
347	2022-12-01	6220.646416	6101.186967	6330.580985
348	2023-01-01	6245.345812	6131.684598	6362.064446
349	2023-02-01	6271.135062	6152.432701	6384.251843
350	2023-03-01	6316.878007	6198.031898	6434.103236
351	2023-04-01	6337.627356	6218.460380	6456.511874
352	2023-05-01	6355.462695	6244.928997	6470.731551
353	2023-06-01	6378.526526	6266.356293	6501.201222
354	2023-07-01	6388.164957	6270.299318	6504.878154
355	2023-08-01	6438.916875	6317.801808	6557.791547

Figura 9 - Previsão dos próximos 12 meses

A previsão com o método ARIMA, gerou um *dataframe* onde *forecast* é o valor previsto e o índice é a data prevista. Aqui podemos observar valores maiores do que os previstos pelo *prophet*, vários testes foram realizados com treinamento do modelo utilizando períodos diferentes, porém isso não afetou significativamente os resultados.

```
2022-09-01    642150.400451
2022-10-01    646256.673868
2022-11-01    649773.575982
2022-12-01    654315.054753
2023-01-01    656573.062700
2023-02-01    660511.360196
2023-03-01    665605.455864
2023-04-01    668520.024051
2023-05-01    670721.582267
2023-06-01    673497.428847
2023-07-01    674034.989134
2023-08-01    674801.980570
Name: forecast, dtype: float64
```

Figura 7 - Previsão ARIMA

Podemos observar que ambos os modelos previram que o IPCA tende a subir, porém a previsão gerada pelo *prophet* resultou em valores mais baixos do que os do ARIMA. Outro ponto importante é que durante os treinos o *prophet* se mostrou mais consistente com valores mais próximos aos valores de treino.

7. Conclusão

Em busca de gerar previsões úteis para a economia brasileira especialistas buscam medidas para prever os principais índices do país. Dessa forma, um modelo eficiente de aprendizado de máquina pode ser uma ferramenta aliada na melhora dessas previsões e por consequência pode auxiliar na tomada de decisões por parte do governo e de empresas.

Técnicas como as utilizadas aqui podem trazer informações importantes a cerca do futuro da economia, assim podemos traçar planos e metas visando o bem-estar de nossa economia como um todo.

Sendo assim é possível concluir que uma previsão do IPCA por meio de aprendizagem de máquina não é precisa, porém nos dá uma visão dos padrões que o índice poderá apresentar nos próximos meses.

Outros modelos e bibliotecas foram testados como o *Catboost* e o *Linear Regression* do *Sklearn*, porém ambos mostraram dificuldades em trabalhar com séries temporais. Por fim, deixo como sugestão para futuros trabalhos o teste de outros métodos que incluam ou não séries temporais.

8. Links

Link para o repositório contendo os dados processados e os *notebooks*:

https://github.com/Weivak/ML-IPCA-Pred

Link para download dos dados:

https://www.ibge.gov.br/estatisticas/economicas/precos-e-custos/9256-indice-nacional-de-precos-ao-consumidor-amplo.html?=&t=downloads

REFERÊNCIAS

Box, G.E.P. e Jenkins, G.M. Time Series Analysis: Forecasting and Control. San Francisco: Holden Day, 1970.

BRITO, D. M. D. et al. Predição de desempenho de alunos do primeiro período baseado nas notas de ingresso utilizando métodos de aprendizagem de máquina. III Congresso Brasileiro de Informática na Educação. [S.l.]: [s.n.]. 2014.

CNN, 2022. Disponivel em: <https://www.cnnbrasil.com.br/business/pesquisa-diz-que-93-dos-brasileiros-sentiram-alta-dos-precos-desde-o-comeco-do-ano/>.

FEBRABAN, 2022. Disponivel em: <https://economia.uol.com.br/noticias/estadao-conteudo/2022/06/15/93-dos-brasileiros-sentiram-alta-dos-precos-desde-o-comeco-do-ano-diz-pesquisa.htm>.

IBGE, 2022. Disponivel em: <https://www.ibge.gov.br/estatisticas/economicas/precos-e-custos/9256-indice-nacional-de-precos-ao-consumidor-amplo.html?=&t=o-que-e>.

SANDRONI, Paulo, Novíssimo Dicionário de Economia, São Paulo/SP, Editora Best Seller, 1999.

Serrão, F.C.C. Modelo de previsão de carga de curto prazo utilizando redes neurais e lógica fuzzy. Master’s thesis, Pontífica Universidade Católica, Brasil, Rio de Janeiro, 2003.

SILVA, F. M. D. et al. Previsão de geração de energia elétrica renovável em curto prazo no estado do Ceará utilizando modelo de regressão prophet. Research, Society and Development, 18 mai. 2022.

Taylor, S. J., & Letham, B. ; Forecasting at scale. PeerJ Preprints.; 2017