Universidade de Brasília

Faculdade de Direito

O JULGAMENTO EM LISTA NAS AÇÕES DIRETAS DE INCONSTITUCIONALIDADE

Pedro Ian Ramalho Luz de Castro

Brasília

2022

Universidade de Brasília

Faculdade de Direito

O JULGAMENTO EM LISTA NAS AÇÕES DIRETAS DE INCONSTITUCIONALIDADE

Pedro Ian Ramalho Luz de Castro

Trabalho Final apresentado como pré-requisito para a obtenção do título de Mestre em Direito pela Universidade de Brasília.

Orientador: Prof. Dr. Alexandre Araújo Costa

Brasília

2022

Castro, Pedro Ian Ramalho Luz de

O JULGAMENTO EM LISTA NAS AÇÕES DIRETAS DE INCONSTITUCIONALIDADE/ Pedro Ian Ramalho Luz de Castro – Brasília, 2022.

Dissertação de Mestrado – Universidade de Brasília, Faculdade de Direito, 2022.

Orientador: Prof. Dr. Alexandre Araújo Costa.

1 - Controle Concentrado de Constitucionalidade. 2 - Poder Judiciário. 3- Jurimetria. 4- Ações Diretas de Inconstitucionalidade.

O JULGAMENTO EM LISTA NAS AÇÕES DIRETAS DE INCONSTITUCIONALIDADE

PEDRO IAN RAMALHO LUZ DE CASTRO

Trabalho Final apresentado à Faculdade de Direito da Universidade de Brasília para obtenção do título de Mestre em Direito e apresentado e aprovado pela seguinte banca examinadora:

Prof. Dr. Alexandre Araújo Costa
(Universidade de Brasília)
Orientador

Prof. Dr. Lucio Remuzat Rennó Junior
(Universidade de Brasília)
Membro da Banca

Prof. Dr. Henrique Augusto Figueiredo Fulgêncio
(Escola da Advocacia-Geral da União)
Membro da Banca

Prof. Dr. Henrique Araújo Costa
(Universidade de Brasília)
Membro da Banca - Substituto

AGRADECIMENTOS

A elaboração deste trabalho simplesmente não teria sido possível sem o suporte da minha família. Quero agradecer primeiramente à minha mãe e madrinha, por todo o amor e pelo apoio que deram à minha carreira acadêmica e ao meu desenvolvimento pessoal. Agradeço por tudo, até mesmo por todas as vezes que perguntaram quando esta dissertação ficaria pronta. Sem vocês, ela teria demorado ainda mais. Agradeço imensamente à minha amada esposa, Letícia, por ser minha parceira de vida e por todo o incentivo que me deu na elaboração desta dissertação. Seu amor deixa tudo mais leve.

Agradeço ao meu orientador, prof. Alexandre Araújo Costa, que primeiro me encorajou a combinar a programação e o estudo do judiciário. Ser seu orientando foi um divisor de águas na minha carreira e, sobretudo, um enorme privilégio. Não poderia ter tido um orientador melhor.

Agradeço ao prof. Lucio Rennó, por ter acreditado em mim e por ter me recebido tão bem na ciência política, que agora é minha casa. Muito obrigado por tudo.

Por fim, agradeço ao contribuinte, que custeou meus estudos.

RESUMO

O presente trabalho aborda o julgamento em lista de Ações Diretas de Inconstitucionalidade, objetivando compreender a forma como essa técnica tem sido usada pelo plenário do Supremo Tribunal Federal. Objetiva-se preencher uma possível lacuna na literatura, visto que não há um levantamento quantitativo das ações de controle concentrado julgadas em sistema de lista. Para tanto, faz-se uso de técnicas de coleta e raspagem de dados para se construir uma base de dados original com todos os acórdãos de ADIs publicados pela Corte até o final de 2019. Essa base, contendo 2136 ações, é então usada para calcular o grau de semelhança entre documentos diferentes, bem como um índice de singularidade de cada ação por forma de julgamento.

A pesquisa tem natureza empírica e mescla abordagens quantitativas e qualitativas. Devido à vasta quantidade de documentos analisada, faz-se extenso uso de técnicas automatizadas de classificação de documentos e análise de conteúdo típicas da ciência da computação. Esses métodos são combinados com o estudo do registro audiovisual do Supremo Tribunal Federal durante o ano de 2018 por meio da TV Justiça.

Registrou-se a duração de diferentes eventos no julgamento em lista de 77 ADIs no ano de 2018. A duração mediana e média dos julgamentos foi de 39 e 146 segundos, respectivamente. Observou-se um descompasso entre a extensão dos acórdãos e a duração das sessões do plenário.

Ainda, foi utilizado um algoritmo de *clustering* hierárquico aglomerante (*hierarchical agglomerative clustering*), conjuntamente com a técnica da semelhança do cosseno, para criar um índice de singularidade associado à forma de julgamento da ADI. A fim de melhor explorar as diferenças entre diferentes modos de julgamento, foram usados um algoritmo de alocação latente de Dirichlet e um classificador bayesiano ingênuo.

Os resultados encontrados indicam que, ao contrário do que se esperava, ações julgadas em lista não são substancialmente mais diferentes entre si que ações julgadas tradicionalmente. Contudo, observou-se uma taxa de unanimidade muito maior para ações julgadas em lista: 90% das ADIs julgadas em lista foram decididas à unanimidade, contra 63 e 69% das ações julgadas tradicionalmente e no plenário virtual, respectivamente.

Palavras-Chave: Controle Concentrado de Constitucionalidade, Poder Judiciário, Jurimetria, Ações Diretas de Inconstitucionalidade.

ABSTRACT

This paper discusses the expediting of abstract constitutional review cases trials ("*julgamento em lista*" of ADIs), aiming to understand how this technique has been used by the Federal Supreme Court *en banc* sessions. The goal is to fill a perceived gap in the literature, since there is no quantitative survey of concentrated judicial review cases judged "*em lista*". To this end, data collection and scraping techniques are used to build an original database with all ADI judgments published by the Court until the end of 2019. This database, containing 2136 cases, is then used to calculate the degree of similarity between different documents, as well as an index of uniqueness of each case grouped by mode of trial.

The research is empirical in nature and mixes quantitative and qualitative approaches. Due to the vast quantity of documents analyzed, extensive use is made of automated document classification and content analysis techniques typical of computer science. These methods are combined with the study of the audiovisual registry of the Federal Supreme Court during the year 2018 through *TV Justiça*.

The duration of different events in the trial was recorded in 77 ADIs judged "*em lista*" in 2018. The median and mean duration of the trials were 39 and 146 seconds, respectively. There was a mismatch between the length of the judgments and the length of the judicial opinions.

Furthermore, a hierarchical agglomerative clustering algorithm was used, together with the cosine similarity technique, to create a uniqueness index associated with the form of judgment of the ADI. In order to better explore the differences between different judgment methods, a Dirichlet latent allocation algorithm and a naive Bayesian classifier were used.

The results found indicate that, contrary to what was expected, actions judged "*em lista*" are not substantially more different from each other than actions judged traditionally. However, a much higher unanimity rate was observed for actions judged "*em lista*": 90% of the ADIs judged "*em lista*" were decided unanimously, against 63 and 69% of the actions judged traditionally and in the virtual *en banc* sessions, respectively.

Keywords: Abstract Constitutional Review, Judiciary Branch, Jurimetrics, Ações Diretas de Inconstitucionalidade.

Sumário

Lista de Siglas

ADI	Ação Direta de Inconstitucionalidade
COVID-19	Doença Causada pelo Coronavírus
HAC	*Hierarchical Agglomerative Clustering* (clustering hierárquico aglomerante)
HC	*Habeas Corpus*
ICMS	Imposto Sobre Circulação de Mercadorias e Serviços
LDA	*Latent Dirichlet Allocation* (modelo de alocação latente de Dirichlet)
LSI	*Latent Semantic Indexing* (modelo de indexação semântica latente)
OCR	*Optical Character Recognition* (reconhecimento óptico de caracteres)
RE	Recurso Extraordinário
RPPS	Regime Próprio de Previdência Social
SCOTUS	*Supreme Court of the United States* (Suprema Corte dos Estados Unidos)
STF	Supremo Tribunal Federal
SVM	*Support-Vector Machine* (máquina de vetores de suporte)
TJ-SP	Tribunal de Justiça do Estado de São Paulo

Índice de Tabelas

Índice de Figuras

Introdução

Em 19 de Agosto de 2015, ao final da 21ª sessão ordinária de 2015 do Supremo Tribunal Federal (STF), o então presidente da Suprema Corte, ministro Ricardo Lewandowski, começou a designar listas de processos para julgamento. A prática tornou-se corriqueira: ao final das sessões plenárias, listas compostas de processos de menor complexidade, como embargos de declaração e agravos regimentais, eram apregoadas para julgamentos rápidos, comumente durando poucos minutos.

Uma a uma, o plenário do Supremo Tribunal Federal analisa rapidamente diversas listagens de julgamento, sem suscitar maior debate entre os ministros, até o momento em que o ministro Lewandowski chamou para julgamento a lista número 1 de relatoria do ministro Luís Roberto Barroso, contendo a ADI 5.075/DF. De imediato, o conteúdo da lista provocou estranheza. Tratava-se de uma Ação Direta de Inconstitucionalidade (ADI), modalidade de controle concentrado que, por sua natureza, não se amolda bem à apreciação célere de um julgamento em lista. Questionado pelo ministro Marco Aurélio de Mello se era mesmo uma ADI que estava sendo pautada, o relator, ministro Barroso, explica que a Corte firmou entendimento de que casos que tratassem de impugnação de legislação estadual e que fossem mera reprodução de jurisprudência passariam a ser julgados em listas[1]. A ADI 5.075/DF foi então julgada procedente, por unanimidade, e a lei estadual foi declarada inconstitucional. O julgamento inteiro durou 87 segundos.

Longe de ser uma idiossincrasia do STF, o uso de listas de julgamento é um lugar-comum na realidade forense brasileira, sendo usadas por todo o sistema Judiciário como uma técnica de gestão do acervo processual[2]. Apesar de sua abrangência, a prática aparenta não ter sido objeto de maior curiosidade doutrinária, dado que não é discutida em tratados de processo civil e há parcos artigos jurídicos sobre o tema[3].

A sessão plenária de 19 de Agosto de 2015 é, assim, notável não por conter julgamentos em lista, mas por ter sido o primeiro caso em que uma Ação Direta de Inconstitucionalidade foi apreciada em lista. Apesar de constituir uma fatia relativamente pequena do número total de decisões tomadas pela Suprema Corte[4], o controle

[1] portal.stf.jus.br/noticias/verNoticiaDetalhe.asp?idConteudo=298029
[2] Confira-se, a título de exemplo, o artigo 290 do Tribunal de Justiça do Estado do Pará, bem como o artigo 143 do regimento do Tribunal de Contas da União.
[3] A primeira menção ao uso do julgamento em lista em um periódico jurídico parece ter ocorrido em 2004, na revista Migalhas, em um artigo intitulado "Súmula vinculante e Julgamentos por lista". Cf. www.migalhas.com.br/depeso/7081/sumula-vinculante-e-julgamentos-por-lista.
[4] FALCÃO, Joaquim; CERDEIRA, Pablo; ARGUELHES, Diego, I Relatório do Supremo em Números - O múltiplo Supremo, **Revista de Direito Administrativo**, v. 262, p. 399, 2013.

concentrado de constitucionalidade por via de ação é a função constitucional do Tribunal por excelência e ocupa boa parte da pauta de julgamentos do Plenário. É a importância das ADIs dentro da dinâmica de atuação da Corte que torna o uso da técnica do julgamento em lista um objeto de estudo digno de nota[5].

Objeto

A fim de entender a função desempenhada por essa ferramenta no funcionamento da Corte, é necessário compreender como a jurisdição constitucional brasileira evoluiu ao longo do tempo. O século XX presenciou uma contínua expansão do poder judiciário em todo o mundo, expansão esta que se acelerou ainda mais no período pós-Guerra[6]. No caso do Brasil, esse alargamento da jurisdição constitucional concentrada e abstrata remonta, ao menos, à emenda constitucional nº 16, de 1965, que primeiro deu ao STF a competência para julgar a Representação de Inconstitucionalidade, germe da ADI[7]. Esse alargamento da jurisdição constitucional se perpetua até hoje[8].

Ao longo das últimas décadas, diversas alterações foram feitas na estrutura do controle de constitucionalidade, a fim de evitar um suposto colapso do Judiciário frente à incapacidade de lidar com a quantidade de demandas processuais. Como bem salientam Costa, Carvalho e Farias[9], esse conjunto de reformas é caracterizado pelo aumento das faculdades de *concentração* e *seletividade* da Corte. Isto é, essas reformas concentraram o poder do STF, possibilitando que suas decisões se aplicassem a múltiplos processos, e, simultaneamente, criaram mecanismos para limitar o acesso ao controle de constitucionalidade, diminuindo assim a quantidade de processos cujo mérito deveria ser julgado pelo Tribunal[10].

---

[5] Ademais, ADIs são sobre-representadas em processos de relevância política. No levantamento feito por Kapiszewski dos casos mais importantes apreciados pelo STF entre 1985 e 2005, 30 dos 55 casos selecionados são ADIs. KAPISZEWSKI, Diana, Power Broker, Policy Maker, or Rights Protector?, *in*: HELMKE, Gretchen; RIOS-FIGUEROA, Julio (Orgs.), **Courts in Latin America**, Cambridge: Cambridge University Press, 2011, p. 154–186..

[6] TATE, C. Neal; VALLINDER, Torbjörn (Orgs.), **The global expansion of judicial power**, New York: New York University Press, 1995.

[7] COSTA, Alexandre Araújo; CARVALHO, Alexandre Douglas Zaidan de; FARIAS, Felipe Justino de, Controle de constitucionalidade no Brasil: eficácia das políticas de concentração e seletividade, **Revista Direito GV**, v. 12, n. 1, p. 155–187, 2016.

[8] BARROSO, Luís Roberto, **O controle de constitucionalidade no direito brasileiro: exposição sistemática da doutrina e análise crítica da jurisprudência**, 7a edição revista e atualizada. São Paulo, SP: Editora Saraiva, 2016.

[9] COSTA; CARVALHO; FARIAS, Controle de constitucionalidade no Brasil.

[10] *Ibid*.

A recente ampliação do instituto do julgamento em lista para contemplar ADIs pode, então, ser entendida como mais um movimento dentro desse processo de expansão da jurisdição constitucional, possibilitando julgamentos rápidos e, assim, maior controle sobre a forma como o Tribunal aloca o tempo escasso de seus ministros.

Contudo, a técnica do julgamento em lista, criada para lidar rapidamente com blocos de processos de baixa complexidade, parece, ao menos à primeira vista, inadequada para a deliberação necessária ao controle concentrado de constitucionalidade. O caráter deliberativo dos tribunais constitucionais é, afinal, uma das principais justificativas para a realização do controle de constitucionalidade pelo Judiciário: na tradicional defesa teórica do *judicial review* associada a Dworkin[11], os tribunais, por serem "fóruns de princípio", estariam mais bem equipados para a deliberação racional de princípios constitucionais.

Essa tensão aparente entre o caráter deliberativo de uma corte constitucional e o rito abreviado das listas é, inclusive, muito bem trabalhada por um dos próprios ministros da Corte, Luís Roberto Barroso. Em um artigo intitulado "Como salvar o sistema de repercussão geral: transparência, eficiência e realismo na escolha do que o Supremo Tribunal Federal vai julgar", Barroso caracteriza o julgamento em lista como "mecanismo sumário em que não há debate, e no qual dezenas de casos podem ser julgados por vez" e propõe expandir o sistema de repercussão geral a fim de libertar "o Tribunal da necessidade de proferir uma enxurrada de decisões monocráticas e julgamentos em lista, que não são nada além de uma ficção de justiça, uma forma de obscurecer um juízo inevitavelmente discricionário, com um verniz pretensamente técnico que não se sustenta e apenas desgasta a Corte"[12]. Essas limitações, inerentes ao modelo do julgamento em lista, ajudariam a explicar por que, desde o julgamento da ADI 5.075/DF, o plenário do STF escolheu julgar diversas ADIs da forma tradicional[13].

O que determina, então, se uma ADI será ou não pautada em lista? O exame do Regimento Interno do STF revela poucos critérios. Assim dispõe o art. 21-B do Regimento, o único a mencionar o julgamento em lista:

---

[11] DWORKIN, Ronald, **A matter of principle**, 9th print. Cambridge, Mass: Harvard Univ. Press, 2000.
[12] BARROSO, Luís Roberto; MONTEDONIO REGO, Frederico, Como Salvar o Sistema de Repercussão Geral: Transparência, Eficiência e Realismo na Escolha do que o Supremo Tribunal Federal Vai Julgar, **Revista Brasileira de Políticas Públicas**, v. 7, n. 3, 2018.
[13] Chamaremos de *tradicional* o julgamento presencial convencional, com deliberação ampla e plena.

> Todos os processos de competência do Tribunal poderão, **a critério do relator ou do ministro vistor com a concordância do relator**, ser submetidos a julgamento em listas de processos em ambiente presencial ou eletrônico, observadas as respectivas competências das Turmas ou do Plenário.
> §1º Serão julgados preferencialmente em ambiente eletrônico os seguintes processos:
> I – agravos internos, agravos regimentais e embargos de declaração;
> II – medidas cautelares em ações de controle concentrado;
> III – referendo de medidas cautelares e de tutelas provisórias;
> IV – demais classes processuais, inclusive recursos com repercussão geral reconhecida, cuja matéria discutida tenha jurisprudência dominante no âmbito do STF.

Surpreende que o referido artigo tenha sido incluído apenas com a Emenda Regimental 52/2019, e posteriormente modificado pela Emenda Regimental 53/2020 a fim de abarcar todo tipo de processo, quase quatro anos após o primeiro julgamento em lista de uma ADI, em 2015. Assim, de 19 de Agosto de 2015 até a publicação da ER 52, em 19 de junho de 2019, não existia no Regimento qualquer menção à possibilidade de julgamento de ADI na sistemática de lista.

O regimento tampouco estipula normas explícitas que determinem quais processos podem ser pautados em lista, deixando a decisão inteiramente ao Relator. Apesar de não estabelecer parâmetros para o julgamento em lista presencial, o primeiro parágrafo do artigo 21-B sugere critérios para o julgamento em lista em ambiente eletrônico. As ADIs estariam, então, abarcadas pelo inciso IV, "demais classes processuais cuja matéria discutida tenha jurisprudência dominante no âmbito do STF.". Dessa forma, em tese, seriam julgadas em lista por via eletrônica apenas as ADIs que tratassem de questões já enfrentadas reiteradas vezes pela Corte.

Outro aspecto digno de nota é que o regimento delega ao relator a decisão de julgar ou não um determinado processo em lista. Essa decisão se insere dentro de um movimento maior no Direito brasileiro de gradual ampliação dos poderes do relator. Tal como a expansão da jurisdição constitucional como um todo, a concentração de poderes na figura do relator é apresentada como um processo de modernização, a fim de tornar os julgamentos mais rápidos e efetivos e, assim, melhor gerir o acervo processual da Corte.

Contudo, como apontam Hartmann e Arguelhes[14] em um levantamento empírico das mudanças legislativas que aumentaram os poderes do relator no âmbito do STF, o argumento da efetividade nem sempre encontra respaldo nos dados sobre os processos judiciais.

Hipótese

Quais fatores influenciam a decisão da Corte de julgar algumas ADIs em lista e outras não? A existência, ainda que tardia, de um critério explícito no regimento, a saber, a presença de uma jurisprudência dominante, sugere que a técnica do julgamento em lista é reservada para ADIs repetitivas, que enfrentam assuntos que, apesar de pacificados pela Corte, são levados reiteradamente a sua apreciação.

Assim, seriam julgados em lista processos semelhantes a outros julgados anteriormente, com conteúdo repetitivo. O julgamento em lista, no âmbito das ADIs, seria uma forma de racionalizar o uso do tempo do Tribunal nas escassas sessões de plenário.

Como testar, empiricamente, essa hipótese? Uma primeira dificuldade a ser enfrentada em uma análise desse porte é a identificação dos casos que foram julgados em lista. Essa informação é surpreendentemente difícil de ser obtida. Não há, nos acórdãos publicados, qualquer indicação explícita de que o caso tenha sido julgado em lista. Tampouco é possível usar a extensão dos acórdãos como parâmetro para determinar se determinada ADI foi de fato debatida em sessão plenária, visto que, como acórdãos não são transcrições das sessões, seu número de páginas ou mesmo a extensão dos votos dos ministros não são critérios suficientes para estimar a duração de um julgamento.

Ainda, o estudo esbarra na dificuldade de como medir a existência de casos semelhantes. Mesmo que o pesquisador se restrinja a decisões de mérito, o elevado número de ações julgadas pelo STF inviabiliza examinar cada ADI individualmente a fim de determinar se se trata ou não de uma questão já enfrentada antes pelo Tribunal.

Nesse sentido, o presente trabalho busca preencher uma lacuna na literatura, construindo uma base de dados única para avaliar empiricamente o uso da técnica do

[14] HARTMANN, Ivar Alberto Martins; FERREIRA, Lívia Da Silva, Ao relator, tudo: o impacto do aumento do poder do ministro relator no Supremo, **Revista Opinião Jurídica (Fortaleza)**, v. 13, n. 17, p. 268, 2016.

julgamento em lista no julgamento de ADIs e fazendo uso de análise automatizada de conteúdo textual para construir um índice de singularidade processual.

Literatura

Este trabalho está situado na interseção de duas literaturas distintas, uma substantiva e outra metodológica. Substantivamente, o presente estudo dialoga com pesquisas quantitativas sobre as ações de controle concentrado, tanto no campo do Direito como na ciência política.

Apesar de as análises empíricas ainda serem incipientes na produção acadêmica jurídica brasileira[15], as ações de controle concentrado têm sido objeto de múltiplas pesquisas nos últimos anos, sobretudo pelo prisma quantitativo. Desde os trabalhos pioneiros de Castro[16] e Vianna[17], que primeiro fizeram uso de dados agregados sobre ações de controle concentrado para compreender o comportamento judicial, desenvolveu-se uma rica literatura para estudar empiricamente a jurisdição constitucional brasileira, como pode ser verificado nos trabalhos de Costa e Costa[18], Benvindo e Costa[19] e Gomes Neto[20], entre outros.

No campo da ciência política, bem como nas demais ciências sociais de orientação primariamente quantitativa, as ADIs são praticamente o único tipo de ação utilizada no estudo da Suprema Corte[21]. Essas análises são de natureza empírica, mas, em um reflexo da literatura comparada sobre comportamento judicial, estão majoritariamente interessadas em medir a ideologia dos ministros, normalmente aplicando uma estimativa

[15] MACHADO, Maíra Rocha, **Pesquisar empiricamente o direito**, [s.l.]: Rede de Estudos Empíricos em Direito, 2017.
[16] CASTRO, Marcus Faro de, O Supremo Tribunal Federal e a Judicialização da Política, p. 19, 1997.
[17] VIANNA, Luiz Werneck (Org.), **A judicialização da política e das relações sociais no Brasil**, Rio de Janeiro: Editora Revan, 1999.
[18] COSTA, Alexandre Araújo; COSTA, Henrique Araújo, Evolução do perfil dos demandantes no controle concentrado de constitucionalidade realizado pelo STF por meio de ADIs e ADPFs, v. 49, n. 2, p. 47, 2018.
[19] BENVINDO, Juliano Zaiden; COSTA, Alexandre Araújo, A Quem Interessa o Controle Concentrado De Constitucionalidade? - O Descompasso entre Teoria e Prática na Defesa dos Direitos Fundamentais, p. 84, 2014.
[20] GOMES NETO, Jose Mario Wanderley *et al*, Litígios Esquecidos: Análise empírica dos processos de controle concentrado de constitucionalidade aguardando julgamento, **Revista de Estudos Empíricos em Direito**, v. 4, n. 2, 2017.
[21] NERY FERREIRA, Pedro Fernando Almeida; MUELLER, Bernardo, How judges think in the Brazilian Supreme Court: Estimating ideal points and identifying dimensions, **EconomiA**, v. 15, n. 3, p. 275–293, 2014.

de pontos ideais e modelos espaciais ao placar de votações dos ministros[22]. Outros métodos comuns para quantificar ideologia fazem uso de elementos textuais, como *wordscoring* e *wordfish*[23].

Contudo, não obstante a riqueza e diversidade dos trabalhos empíricos existentes sobre ações de controle concentrado, pouca atenção tem sido dedicada à aplicação da técnica do julgamento em lista no âmbito das ADIs. Isso se deve ao fato de pesquisas empíricas sobre o STF partirem de informações disponíveis publicamente em bases de dados mantidas pelo Tribunal, como descrito em Costa e Costa[24]. Assim, apesar de comumente elaborarem classificações próprias, essas pesquisas estão limitadas aos metadados já constantes nessas bases de dados, tais como o número do processo, nomes das partes, datas de ingresso e distribuição, ementa e ministro relator. Não há dados públicos de fácil acesso que revelem quais ADIs foram julgadas em lista. Essa informação não consta nas bases públicas da Corte e nem mesmo está mencionada nos acórdãos publicados. Dessa forma, a única forma de saber se determinada ação foi julgada em lista é coletar esse dado diretamente da pauta de julgamento do plenário para, então, manualmente verificar se a ação foi incluída em uma listagem de relatoria de algum dos ministros.

Metodologicamente, esta dissertação se relaciona com a aplicação de métodos e técnicas de processamento de linguagem natural às ciências sociais, inserindo-se dentro do campo de pesquisa denominado *text as data*, ou texto como dado[25]. Originalmente

---

[22] MARTIN, Andrew D.; QUINN, Kevin M., Dynamic Ideal Point Estimation via Markov Chain Monte Carlo for the U.S. Supreme Court, 1953–1999, **Political Analysis**, v. 10, n. 2, p. 134–153, 2002; DESPOSATO, Scott W.; INGRAM, Matthew C.; LANNES, Osmar P., Power, Composition, and Decision Making: The Behavioral Consequences of Institutional Reform on Brazil's *Supremo Tribunal Federal*, **Journal of Law, Economics, and Organization**, v. 31, n. 3, p. 534–567, 2015; NERY FERREIRA; MUELLER, How judges think in the Brazilian Supreme Court; JALORETTO, M.F.; MUELLER, B.P.M., O Procedimento de Escolha dos Ministros do Supremo Tribunal Federal – Uma Análise Empírica, **Economic Analysis of Law Review**, v. 2, n. 1, p. 170–187, 2011.

[23] De forma simplificada, essas técnicas objetivam estimar como determinado documento se compara a certos pontos de referência. Por exemplo, é possível definir duas coletâneas de textos como referências para "esquerda" e "direita" e depois estimar em qual ponto determinado texto está no espectro entre os dois extremos. GRIMMER, Justin; STEWART, Brandon M., Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts, **Political Analysis**, v. 21, n. 3, p. 267–297, 2013; SLAPIN, Jonathan B.; PROKSCH, Sven-Oliver, A Scaling Model for Estimating Time-Series Party Positions from Texts, **American Journal of Political Science**, v. 52, n. 3, p. 705–722, 2008; LAVER, Michael; BENOIT, Kenneth; GARRY, John, Extracting Policy Positions from Political Texts Using Words as Data, **American Political Science Review**, v. 97, n. 02, 2003.

[24] COSTA; COSTA, Evolução do perfil dos demandantes no controle concentrado de constitucionalidade realizado pelo STF por meio de ADIs e ADPFs.

[25] MOREIRA, Davi; IZUMI, Maurício, O texto como dado: desafios e oportunidades para as ciências sociais, **Revista Brasileira de Informação Bibliográfica em Ciências Sociais - BIB**, v. 86, p. 138–174, 2018.

advindas da ciência da computação e da linguística, técnicas de análise automatizada de conteúdo têm ganhado cada vez mais espaço nas ciências sociais, particularmente na ciência política. Devido ao fato de a linguagem natural ser o meio pelo qual a política e o Direito se concretizam no mundo[26], o uso de texto no estudo desses domínios é fundamental, tanto em pesquisas qualitativas quanto quantitativas.

Contudo, a vasta quantidade e o volume de documentos produzidos na prática jurídica e política introduzem diversas dificuldades, particularmente quando se trata do estudo do poder judiciário. O número elevado de ações julgadas pelo STF inviabiliza o exame individual de todas os casos. Uma seleção por amostragem de alguns processos de interesse poderia introduzir vieses indesejados na análise. Como a atenção e tempo dos pesquisadores são recursos escassos, é muitas vezes impossível traçar um panorama completo de um objeto de estudo e, simultaneamente, usar o texto. Isto é, os pesquisadores são forçados a escolher entre duas opções: ou usam o texto dos processos, mas se limitam a uma amostra pequena de casos que se pretendem representativos, ou abrangem a totalidade dos casos, mas ignoram o conteúdo dos processos.

Nesse contexto, a análise automatizada de conteúdo possibilitada por técnicas de mineração de texto abre diversas novas fronteiras de pesquisa, o que leva alguns autores, como Grimmer e Stewart[27], a afirmar que elas "tornam possível o que antes era impossível: a análise sistemática de vastas coleções de texto sem que seja necessário obter grande financiamento para a pesquisa" (a fim de contratar mais pesquisadores). Grimmer e Stewart[28] têm o cuidado de ressaltar que, por óbvio, a análise computadorizada de conteúdo não é um substituto para a leitura cuidadosa, mas antes uma ferramenta para ampliar as capacidades interpretativas do pesquisador. Métodos computadorizados para a análise de texto são, afinal, modelos e vale a célebre colocação de Box[29]: todos os modelos estão errados, mas alguns são úteis.

O uso de métodos computacionais de análise de texto no estudo do poder judiciário ainda é incipiente, mas há cada vez mais trabalhos aplicando essas inovações em contextos jurídicos, particularmente no contexto norte-americano. O campo de pesquisa sobre o comportamento judicial é, afinal, dominado pela produção acadêmica dos Estados Unidos: os principais modelos teóricos para explicar o comportamento

---

[26] GRIMMER; STEWART, Text as Data.
[27] *Ibid*.
[28] *Ibid*.
[29] BOX, George E. P., Science and Statistics, **Journal of the American Statistical Association**, v. 71, n. 356, p. 791–799, 1976.

judicial foram desenvolvidos lá, fazendo uso de conceitos originalmente criados para explicar a realidade do poder judiciário americano[30].

Evans *et al.*, Hausladen *et al.*, e Kaufman *et al.*[31] são três bons exemplos de artigos recentes que aplicam técnicas de análise automatizada de conteúdo ao estudo do judiciário norte-americano. Evans *et al*.[32] avaliam a performance de métodos de *wordscoring* para medir a ideologia da Suprema Corte dos Estados Unidos (SCOTUS), bem como o desempenho de um classificador bayesiano ingênuo (*Naïve Bayes classifier*). O artigo obtém resultados encorajadores e os autores observam que todos os métodos testados tiveram acurácia adequada ao classificar a ideologia de manifestações de *amicus curiae* nos casos de ação afirmativa *Regents of the University of California v. Bakke* e *Grutter v. Bollinger*.

Hausladen *et al*.[33], em uma abordagem metodologicamente semelhante à empregada no presente trabalho, testam a aplicabilidade de diversos métodos supervisionados de aprendizado de máquina ao estudo do poder judiciário. Especificamente, os autores fazem uma análise comparada do desempenho de um conjunto de classificadores estatísticos (regressão logística, classificador passivo agressivo, classificador de Ridge e máquina de vetores de suporte) na tarefa de classificar a ideologia de diversas decisões dos Tribunais de Recursos. Os modelos são treinados com uma amostra pequena de decisões já previamente classificadas por humanos e são então encarregados de classificar o restante das decisões.

Finalmente, Kaufman *et al*. [34] fazem uso de árvores de decisão aumentadas com o algoritmo *AdaBoost* para construir modelos preditivos dos resultados da SCOTUS. Os autores alimentam o modelo com os votos dos ministros, bem como metadados relativos a cada caso, e obtêm resultados superiores a de outros modelos preditivos, obtendo o resultado correto em mais de 75% dos casos.

---

[30] SILVA, Jeferson Mariano, **Jurisdição constitucional em Espanha (1981-1992) e Brasil (1988-1997)**, 2016.
[31] EVANS, Michael *et al*, Recounting the Courts? Applying Automated Content Analysis to Enhance Empirical Legal Research: Automated Content Analysis to Enhance Empirical Legal Research, **Journal of Empirical Legal Studies**, v. 4, n. 4, p. 1007–1039, 2007; HAUSLADEN, Carina I.; SCHUBERT, Marcel H.; ASH, Elliott, Text classification of ideological direction in judicial opinions, **International Review of Law and Economics**, v. 62, p. 105903, 2020; KAUFMAN, Aaron Russell; KRAFT, Peter; SEN, Maya, Improving Supreme Court Forecasting Using Boosted Decision Trees, **Political Analysis**, v. 27, n. 3, p. 381–387, 2019.
[32] EVANS *et al*, Recounting the Courts?
[33] HAUSLADEN; SCHUBERT; ASH, Text classification of ideological direction in judicial opinions.
[34] KAUFMAN; KRAFT; SEN, Improving Supreme Court Forecasting Using Boosted Decision Trees.

Apesar dessa nova literatura que aplica métodos de aprendizado de máquina ao comportamento judicial focar-se no contexto norte-americano, há cada vez mais interesse em usar essas técnicas no estudo de tribunais de outros países. Liebman *et al*.[35] usam modelagem de tópicos, uma técnica de aprendizado de máquina não supervisionada que usa a escolha de palavras a fim de inferir categorias de tópico em um *corpus*[36], para investigar processos administrativos na província de Henan, China. Sulea *et al*.[37] aplicam um classificador SVM (*support-vector machine*s, ou máquina de vetores de suporte) para prever o resultado de casos da Corte de Cassação da França.

No caso do judiciário brasileiro, dois trabalhos promissores são Stemler[38] e Coelho[39]. Stemler[40] criou mecanismos para identificar precedentes judiciais, medindo o desempenho de diferentes algoritmos, como *Latent Semantic Indexing* (LSI) e *Latent Dirichlet Allocation* (LDA). A ferramenta desenvolvida atingiu desempenho notável, agrupando corretamente mais de 90% dos precedentes judiciais similares a temas de Incidente de Resolução de Demandas Repetitivas e identificando novos temas com precisão superior a 70%[41]. Coelho[42], por sua vez, em uma abordagem semelhante à empregada em Hausladen *et al*.[43], avalia o desempenho de diversos métodos supervisionados de classificação com uma base de dados de processos do TJ-SP.

Metodologia

A presente dissertação busca investigar como o STF tem aplicado a técnica do julgamento em lista às ADIs. O objetivo não é elaborar uma doutrina sobre o julgamento em lista, mas antes compreender, empiricamente, o papel que essa modalidade de julgamento desempenha no controle concentrado de constitucionalidade. A saber, o trabalho almeja responder às seguintes perguntas:

---

[35] LIEBMAN, Benjamin L. *et al*, Mass Digitization of Chinese Court Decisions: How to Use Text as Data in the Field of Chinese Law, **SSRN Electronic Journal**, 2017.
[36] QUINN, Kevin M. *et al*, How to Analyze Political Attention with Minimal Assumptions and Costs, **American Journal of Political Science**, v. 54, n. 1, p. 209–228, 2010.
[37] SULEA, Octavia-Maria *et al*, Exploring the Use of Text Classification in the Legal Domain, **arXiv:1710.09306 [cs]**, 2017.
[38] STEMLER, Igor Tadeu Silva Viana, **Identificação de Precedentes Judiciais por Agrupamento Utilizando Processamento de Linguagem Natural**, Mestrado, Universidade de Brasília, 2019.
[39] COELHO, Gustavo Rodrigues, **Utilizando Text Mining na Taxonomia Processual**, 2018.
[40] STEMLER, **Identificação de Precedentes Judiciais por Agrupamento Utilizando Processamento de Linguagem Natural**.
[41] *Ibid*.
[42] COELHO, **Utilizando Text Mining na Taxonomia Processual**.
[43] HAUSLADEN; SCHUBERT; ASH, Text classification of ideological direction in judicial opinions.

1) Quantas ADIs, proporcionalmente, são julgadas em lista?
2) O STF tem feito maior uso dessa técnica de julgamento desde 2015?
3) As ADIs julgadas em lista tratam mesmo de conteúdo repetitivo?

A fim de responder a essas questões, o trabalho se vale de uma abordagem mista, mesclando métodos quantitativos e qualitativos.

Primeiramente, a pesquisa faz uso de uma nova base de dados, contendo a íntegra dos acórdãos de todas as ADIs com decisão de mérito publicados até 31 de dezembro de 2019. Essa base foi produzida por meio de um processo de raspagem do repositório de jurisprudência do Tribunal. Na sequência, realizou-se um segundo processo de raspagem, dessa vez sobre a pauta do plenário do STF, para determinar quais ações foram julgadas em lista.

Em seguida, extraiu-se o texto dos acórdãos por meio de um *script* programado em Python[44]. O texto extraído foi então consolidado em uma única base de dados, contendo os metadados de cada processo coletados na raspagem, tais como ementa, relator, decisão e legislação citada.

Uma vez construído o banco de dados, utilizou-se um algoritmo de agrupamento (ou *clustering*), Agrupamento Hierárquico Aglomerante (*hierarchical agglomerative clustering*), em combinação com o método da semelhança do cosseno, para criar um índice de singularidade associado a tipos de ADI. Quanto mais próximo de 1 for o índice de singularidade, mais única é a ADI. Por meio desse índice, foi possível quantificar se ADIs julgadas em lista são mais ou menos únicas que processos julgados em rito tradicional. Ainda, para melhor explorar as diferenças entre diferentes técnicas de julgamento, foi usado um algoritmo de alocação latente de Dirichlet e um classificador bayesiano ingênuo.

A taxa de singularidade associada a diferentes modos de julgamento de ADI é utilizada como uma medida da repetitividade do conteúdo. A intuição que justifica essa escolha é que, se ações julgadas em lista tratam sempre dos mesmos temas e citam a

[44] Acórdãos mais antigos, cujo texto ainda não estava digitalizado, foram digitalizados automaticamente por meio do uso do pacote *pytesseract*. O código utilizado neste projeto está disponível em: github.com/prldc/lista.adi/

mesma jurisprudência, é esperado que sejam mais semelhantes entre si e possuam maior taxa de singularidade.

Finalmente, a fim de validar as conclusões obtidas por meio de aprendizado de máquina, o trabalho também se valeu de uma análise qualitativa das sessões dos julgamentos em lista. Essa análise foi realizada por meio das gravações das sessões do plenário de 2018, disponíveis publicamente no canal de YouTube do Tribunal. Durante essa fase, registraram-se dados diversos sobre a natureza e duração dos julgamentos em lista, bem como o comportamento dos ministros durante as sessões. O marco temporal escolhido para esta pesquisa vai do advento da Constituição de 1988 até dezembro de 2019, data da última ADI constante do banco de dados utilizado na pesquisa. O intervalo foi escolhido de modo a não abranger o período de calamidade pública causado pelo novo coronavírus (COVID-19), que alterou o formato do trabalho do Tribunal durante a pandemia.

O presente trabalho é de natureza empírica e busca aproximar o Direito de métodos e técnicas utilizados nas demais ciências sociais. Devido ao objeto de estudo, os julgamentos em lista, ser ainda subexplorado, o objetivo deste ensaio é primariamente exploratório, buscando assim levantar dados primários sobre seu tema de pesquisa. Os resultados dos modelos utilizados, tendo em vista o caráter ainda incipiente das técnicas de aprendizagem de máquina e a complexidade dos textos jurídicos, não devem ser tomados como verdades absolutas. Outros modelos igualmente técnicos poderiam ser utilizados, justificadamente, neste trabalho, e eles possivelmente apresentariam outros resultados. Assim, o principal valor desta pesquisa está nos dados brutos coletados, que são, até onde vai o conhecimento deste autor, inéditos.

A análise em tela procederá da seguinte forma. No primeiro capítulo, discutiremos brevemente a teoria sobre a análise automatizada de conteúdo, bem como conceitos elementares de processamento de linguagem natural, a fim de fundamentar os métodos e técnicas empregados no trabalho. Em um segundo momento, discutiremos o processo de coleta de dados e análise qualitativa dos julgamentos em lista. Ainda, elucidaremos algumas decisões tomadas na construção das bases de dados. No capítulo subsequente, discutiremos os resultados encontrados pelo modelo empregado. Finalmente, teceremos conclusões sobre o projeto.

***

Capítulo 1 – Fundamentação teórica

Este capítulo apresenta os conceitos elementares de processamento de linguagem natural que embasam o presente trabalho. Inicialmente, teço considerações sobre a relação entre o Direito, o texto e a linguagem natural. Em seguida, explico como o conteúdo de textos pode ser analisado quantitativamente, e como a mineração de texto, aqui entendida como o processo de extração automatizada de padrões a partir de informações textuais[45], tem sido usado modernamente nas ciências sociais. Na sequência, esclareço alguns conceitos basilares da teoria da informação e de aprendizado de máquina que fundamentam a metodologia empregada nesta pesquisa. Finalmente, são discutidos diferentes algoritmos de categorização de documentos utilizados na literatura.

1.1 Linguagem natural

O Direito é inerentemente textual[46]. Enquanto fenômeno social e político, o Direito pode se materializar no mundo de diversas formas, mas o texto é sempre a base para o seu estudo e aplicação. Especificamente, textos jurídicos são um tipo de texto escrito em linguagem natural[47], isso é, textos cuja audiência principal são outras pessoas[48]. Em direto contraste com a notação matemática ou o código de máquina, por exemplo, a linguagem natural é um artefato cultural utilizado primariamente para a comunicação entre humanos[49] e, assim, não obedece a um sistema rígido de regras lógicas, tal como a notação formal.

Em seu cerne, um texto escrito em linguagem natural é uma sequência de dados não estruturados[50]. Por dados não estruturados, entende-se dados que não possuem uma estrutura semântica explícita que pode ser prontamente compreendida por um

---

[45] DE AZEVEDO SOARES, Fabio, **Mineração de Textos na Coleta Inteligente de Dados na Web**, MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA, PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO, Rio de Janeiro, Brazil, 2008.
[46] LIVERMORE, Michael A.; ROCKMORE, Daniel N. (Orgs.), **Law as data: computation, text, & the future of legal analysis**, Santa Fe: SFI Press, 2019.
[47] ASH, Elliott; CHEN, Daniel L.; GALLETTA, Sergio, Measuring Judicial Sentiment: Methods and Application to US Circuit Courts, **Economica**, v. 89, n. 354, p. 362–376, 2022.
[48] JURAFSKY, Dan; MARTIN, James H., **Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition**, Upper Saddle River, N.J: Prentice Hall, 2000.
[49] BIRD, Steven, Natural Language Processing with Python, p. 504, 2009.
[50] BENGFORT, Benjamin; BILBRO, Rebecca; OJEDA, Tony, Applied Text Analysis with Python, p. 332, 2018.

computador[51]. Evidentemente, a linguagem natural possui várias complexidades e ambiguidades contextuais que tornam difícil sua interpretação por máquinas.[52]

"Não estruturado", contudo, não quer dizer aleatório, como bem colocado por Bengfort *et al.*[53]. Textos possuem diversas propriedades linguísticas, tais como sintaxe, morfologia e semântica, que permitem que sejam adequadamente compreendidos por outras pessoas. Assim, um fragmento de linguagem natural pode ser processado de forma tal que essas propriedades latentes se tornem explícitas, a fim de que seja interpretado por um computador. Uma vez definido o universo de análise, nós impomos seleções e abstrações para que os textos sejam convertidos em dados estruturados[54].

1.2 Processamento de linguagem natural

O processamento de linguagem natural consiste no conjunto de técnicas computacionais para traçar associações entre linguagens formais e naturais[55]. Essas técnicas compreendem desde as tarefas mais simples, como o cálculo da frequência de palavras, às mais complexas, como os sofisticados modelos de inteligência artificial que buscam interpretar a fala humana. Dentro do domínio do Direito, o uso do processamento de linguagem natural permite que textos jurídicos sejam processados e interpretados por modelos computacionais.

Nem todo processamento de dados textuais é, contudo, processamento de linguagem natural. Como afirmado por Jurafsky[56], o que distingue o processamento de linguagem das demais formas de processamento de dados é o seu uso de conhecimento da linguagem. A fim de ilustrar essa posição, Jurafsky faz referência ao comando de Unix *wc*, utilizado para contar o número de *bytes*, linhas e palavras em um arquivo de texto. Ao calcular a quantidade de linhas ou *bytes*, *wc* faz uso de um simples processamento de dados, mas, para que possa contar o número de palavras, o programa precisa *possuir um*

---

[51] MANNING, Christopher; RAGHAVAN, Prabhakar; SCHUETZE, Hinrich, Introduction to Information Retrieval, p. 581, 2009.
[52] BIRD, Natural Language Processing with Python.
[53] BENGFORT; BILBRO; OJEDA, Applied Text Analysis with Python.
[54] BENOIT, Ken, Text as Data: An Overview, *in*: CURINI, Luigi; FRANZESE, Robert (Eds.), **The SAGE Handbook of Research Methods in Political Science and International Relations**, 1 Oliver's Yard, 55 City Road London EC1Y 1SP: SAGE Publications Ltd, 2020, p. 461–497.
[55] BENGFORT; BILBRO; OJEDA, Applied Text Analysis with Python.
[56] JURAFSKY; MARTIN, **Speech and language processing**.

modelo que indique o que é uma palavra[57]. É a presença de um modelo de linguagem que permite que o texto seja tratado como dado.

Segundo Bengfort *et al.*[58], modelos de linguagem servem-se de uma hipótese simples, fundamental para que seja possível a aplicação de aprendizado de máquina a um texto: a previsibilidade da linguagem. Apesar de sua complexidade inerente, estruturas textuais escritas em linguagem natural obedecem a padrões estatísticos. Nem todo texto, contudo, é igualmente previsível. Tomemos, a título de exemplo, a frase "onde há fumaça, há ___". Ela é dita de baixa entropia, visto que um modelo de linguagem adequadamente treinado apontaria, corretamente, alta probabilidade de a frase ser completada pela expressão ("fogo"), dado que a expressão é popular. Em direto contraste, frases de alta entropia, como "cuidado com o ___", podem ser satisfatoriamente completadas por toda sorte de palavra ("cão", "buraco", etc.), de modo que apenas o início da frase não fornece contexto suficiente para uma previsão acertada.

Modelos computacionais inferem relações entre unidades de linguagem natural (ou *tokens*) por meio de uma análise contextual. Formalmente, como definido por Bengfort *et al.,* esses modelos usam o contexto em que uma determinada frase aparece para definir um espaço de decisão em que apenas algumas possibilidades existem[59]. Em casos de alta entropia, os modelos computacionais não necessariamente possuem o contexto necessário para extrair corretamente o sentido de determinada expressão e, por isso, pode ser necessário suplementar o modelo previamente com um conjunto de possibilidades para que o ele possa computar adequadamente as probabilidades[60].

De acordo com Jurafsky[61], uma das principais inovações dos últimos 50 anos de pesquisa em processamento de linguagem é o uso de novos modelos formais, extraídos da matemática, linguística e ciência da computação, para capturar sintaxe, ambiguidade e interpretação. Assim, é possível extrair o sentido de textos cada vez mais complexos.

1.3 Expressões regulares

Na definição de Jurafsky e Martin, uma expressão regular (*regex*, do inglês *regular expression*) é uma fórmula em uma linguagem própria que é usada para

[57] *Ibid*.
[58] BENGFORT; BILBRO; OJEDA, Applied Text Analysis with Python.
[59] *Ibid*.
[60] *Ibid*.
[61] JURAFSKY; MARTIN, **Speech and language processing**.

especificar certas classes de strings[62]. Uma *string*, por sua vez, é uma sequência de caracteres, normalmente alfanuméricos[63].

Expressões regulares são fundamentais para efetuar buscas em textos. Esse tipo de busca requer um padrão que queremos encontrar e um *corpus* de documentos no qual o procuramos[64]. Por meio do uso de padrões em *regex*, é possível executar consultas (*queries*) extremamente complexas. Por exemplo, a expressão regular "^a..r$" retornará uma *string* de quatro letras que comece com "a" e termine com "r" (eg. "amar", "amor", "agir", mas não "andar").

O uso de expressões regulares é essencial para extrair de um texto informações como um número de telefone ou código postal, pois são sequências textuais que obedecem a padrões previsíveis, sem com isso selecionar também todo tipo de sequência numérica. Da mesma forma, por meio do uso de expressões regulares é possível identificar todo RE ou ADI citados em um determinado julgamento, visto que esses processos são sempre referidos por uma nomenclatura específica e constante ao longo de todo o texto.

1.4 Análise quantitativa de texto nas ciências sociais

Nos últimos anos, tem-se observado uma expansão do uso de métodos computacionais nas ciências sociais, motivada pelo crescimento da disponibilidade de dados que capturam o comportamento humano[65]. O acesso a esses dados, que podem tomar a forma de textos, áudios e vídeos, tem possibilitado a investigação de perguntas de pesquisa que antes eram impossíveis.

Em uma revisão do estado da arte da pesquisa computacional com texto nas ciências sociais (ou *text as data*), O'Connor, Bamman e Smith[66] destacam diversas aplicações de métodos de mineração de texto em outras disciplinas. Na economia,

---

[62] *Ibid*.
[63] *Ibid*.
[64] *Ibid*.
[65] O'CONNOR, Brendan; BAMMAN, David; SMITH, Noah A, Computational Text Analysis for Social Science: Model Assumptions and Complexity, p. 8, 2011.
[66] *Ibid*.

Tetlock[67], Federal Reserve Bank of San Francisco *et al.*[68], e Shah, Isah e Zulkernine[69] usam análise de sentimento, técnica que busca associar determinadas emoções à escolha de palavras em um texto, para estimar como a opinião pública afeta o desempenho dos mercados de ações. Askitas e Zimmermann[70], por sua vez, demonstram forte correlação entre a ocorrência de determinadas pesquisas em buscadores e a taxa de desemprego. Na bibliometria, há vasta literatura sobre como elementos textuais afetam as citações de um artigo[71]. Na ciência política, Quinn *et al.*[72] usam modelagem de tópicos para examinar a pauta legislativa do Senado americano de 1997 a 2004. Black *et al.*[73] fazem uso do conteúdo emocional dos discursos de ministros da Suprema Corte dos Estados Unidos para examinar como a escolha de vocabulário afeta o voto dos ministros.

Todas essas pesquisas seriam impossíveis sem a aplicação de técnicas modernas de mineração de texto, o que leva O'Connor, Bamman e Smith[74] a afirmarem que a análise automatizada de conteúdo, que mescla técnicas desenvolvidas no processamento de linguagem natural, recuperação de informação, mineração de texto e aprendizado de máquina, deveria ser entendida como um tipo de metodologia quantitativa própria das ciências sociais.

A figura 1, traduzida de Grimmer e Stewart[75], detalha as diferentes técnicas de análise automatizada de conteúdo próprias de um trabalho de *text as data*. Como se pode verificar, o primeiro passo é a coleta de textos. Esses textos são então higienizados e preprocessados para facilitar o seu processamento pela máquina. Há diversos métodos de modelagem disponíveis para um pesquisador, como agrupamento (clustering), modelagem de tópicos e métodos de dicionário. A escolha por uma técnica ou outra vai depender do objetivo da pesquisa.

---

[67] TETLOCK, Paul C., Giving Content to Investor Sentiment: The Role of Media in the Stock Market, **The Journal of Finance**, v. 62, n. 3, p. 1139–1168, 2007.
[68] FEDERAL RESERVE BANK OF SAN FRANCISCO *et al*, Measuring News Sentiment, **Federal Reserve Bank of San Francisco, Working Paper Series**, p. 01–49, 2020.
[69] SHAH, Dev; ISAH, Haruna; ZULKERNINE, Farhana, Predicting the Effects of News Sentiments on the Stock Market, *in*: **2018 IEEE International Conference on Big Data (Big Data)**, Seattle, WA, USA: IEEE, 2018, p. 4705–4708.
[70] ASKITAS, Nikolaos; ZIMMERMANN, Klaus F, Google Econometrics and Unemployment Forecasting, **Applied Economics Quarterly**, v. 55, n. 2, p. 107–120, 2009.
[71] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[72] QUINN *et al*, How to Analyze Political Attention with Minimal Assumptions and Costs.
[73] BLACK, Ryan C. *et al*, Emotions, Oral Arguments, and Supreme Court Decision Making, **The Journal of Politics**, v. 73, n. 2, p. 572–581, 2011.
[74] O'CONNOR; BAMMAN; SMITH, Computational Text Analysis for Social Science: Model Assumptions and Complexity.
[75] GRIMMER; STEWART, Text as Data.

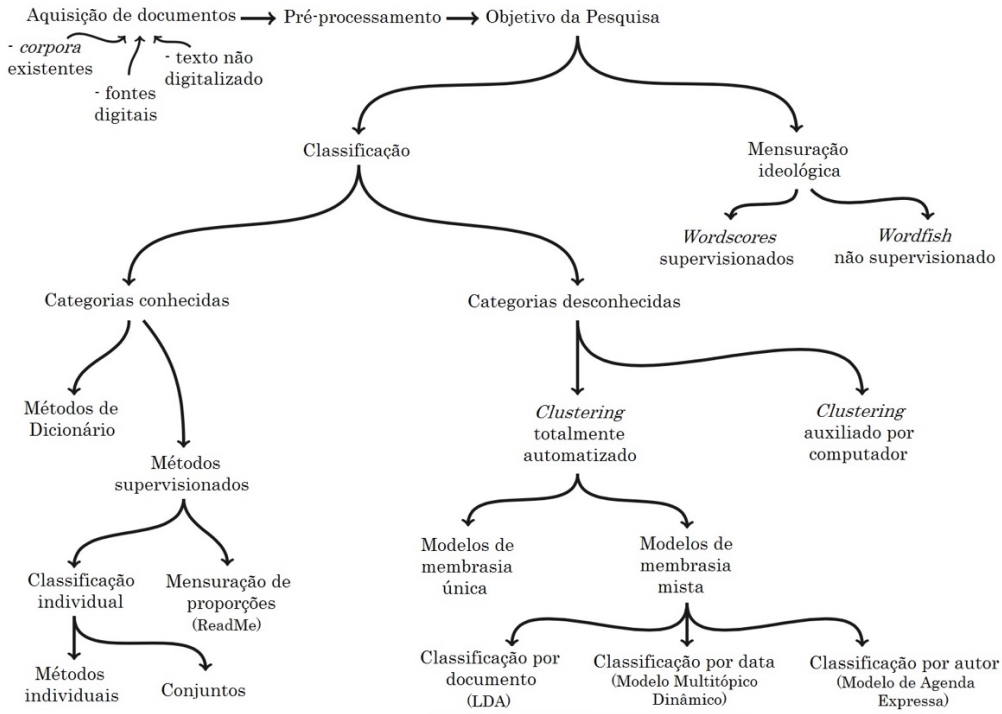


Figura 1: Fluxograma ilustrando o processo de confeção de um trabalho de *text as data*. Traduzido de Grimmer e Stewart (2013).

Para Wilkerson e Casas[76], há quatro fases distintas de um trabalho de *text as data*. Primeiramente, deve-se coletar dados e definir um universo de análise (um *corpus*), depois converter o texto para dados quantitativos, analisá-los e finalmente validá-los. No que segue, detalharemos cada uma dessas etapas.

1.4.1 Raspagem de dados e *crawling*

A primeira etapa de um projeto de análise quantitativa de texto é a coleta de matéria-prima, o objeto textual não estruturado que será estudado. Esse objeto pode ser adquirido de várias fontes diferentes, como bancos de dados já prontos ou livros publicados. Contudo, para certos projetos de pesquisa, não há bases de dados prontas; elas precisam ser construídas por meio de um processo de raspagem, isto é, a coleta

[76] WILKERSON, John; CASAS, Andreu, Large-Scale Computerized Text Analysis in Political Science: Opportunities and Challenges, **Annual Review of Political Science**, v. 20, n. 1, p. 529–544, 2017.

automatizada de dados[77]. No caso de textos disponíveis na internet, essa coleta pode ser feita por *web crawlers*[78].

Segundo Manning, Raghavan e Schuetze[79], *web crawling* é o processo por meio do qual páginas na *web* são coletadas para que sejam indexadas. O objetivo do *crawler* é coletar o maior número possível de páginas de rede, bem como a estrutura de hiperligações que as conectam[80]. Para De Azevedo Soares[81], um *crawler* pode ser definido como um robô que percorre a *web* de forma automatizada e metódica a fim de coletar dados.

Uma vez definidos os parâmetros de operação do *crawler*, ele pode acessar páginas públicas, tais como o *site* do Supremo Tribunal Federal, e baixar arquivos, realizar pesquisas e anotar metadados em um banco de dados.

Neste trabalho, esse processo se deu pela programação de dois *crawlers* na linguagem Python, usando o pacote **scrapy** e a ferramenta de interpretação de JavaScript **splash**. O primeiro *crawler* raspou o banco de jurisprudência do Supremo Tribunal Federal[82] a fim de identificar todas as ADIs julgadas na história do Tribunal com acórdão publicado, bem como obter o respectivo acórdão. O segundo *crawler* raspou a pauta do plenário do Tribunal[83] para identificar as ADIs que foram julgadas em listas.

1.4.2 Construção do *corpus*

Usar texto como dado significa estruturá-lo para que possa ser objeto de análises quantitativas. O primeiro passo é definir um *corpus*, o que, de acordo com Benoit, envolve delimitar uma amostra de documentos disponível, dentro de todo o espaço amostral de documentos que poderiam ter sido selecionados[84]. Todos os níveis de análise de texto giram em torno de uma única base de dados textual, o *corpus*. Bengfort *et al.* definem *corpus* como a coleção de documentos escritos que contém linguagem natural[85].

---

[77] MITCHELL, Ryan, Web Scraping with Python, p. 306, 2018.
[78] DE AZEVEDO SOARES, **Mineração de Textos na Coleta Inteligente de Dados na Web**.
[79] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[80] *Ibid*.
[81] DE AZEVEDO SOARES, **Mineração de Textos na Coleta Inteligente de Dados na Web**.
[82] Disponível em: jurisprudencia.stf.jus.br
[83] Disponível em: portal.stf.jus.br/pauta/pesquisarCalendario.asp
[84] BENOIT, Text as Data.
[85] BENGFORT; BILBRO; OJEDA, Applied Text Analysis with Python.

Um *corpus* pode ser entendido como uma coletânea de documentos individuais, que variam em tamanho e natureza[86]. Esses documentos, por sua vez, são divididos em unidades menores de análise, como parágrafos e frases[87]. Finalmente, com o processo de *tokenização*, essas frases são subdivididas em *tokens*, a unidade de análise textual.

Definido um universo de análise, o próximo passo é preparar o texto para que possa alimentar algoritmos de aprendizado de máquina[88]. O pré-processamento consiste no conjunto de ações tomadas para facilitar a conversão do texto em dados estruturados[89].

1.4.3 Representações quantitativas de documentos

Para que um texto possa alimentar um algoritmo de aprendizado de máquina, é necessário representar o texto não processado como números que possam ser computados[90]. Segundo Gentzkow[91], normalmente são feitas três simplificações: reduzir o texto a documentos individuais, reduzir a quantidade de elementos linguísticos que analisamos e limitar o escopo. O resultado é mapear o texto cru D para um arranjo numérico (*numerical array*) C. Cada entrada de C será um vetor numérico em que cada elemento representa uma unidade menor de linguagem. Segundo Wilkerson e Casas, o objetivo final é normalmente criar uma matriz de documentos e termos (*term-document matrix*) em que cada entrada é um documento e cada coluna é uma característica daquele documento[92]. Essa matriz será então objeto de análises estatísticas.

Benoit afirma que, como essas entradas e colunas não estão em ordem, as características que eram originalmente palavras em sequência são normalmente armazenadas em um objeto sem representação de ordem[93]. Essa abordagem é denominada "***bag-of-words***" (saco de palavras), em que cada texto é representado como um conjunto de palavras, medidas nos termos de sua frequência, mas sem atentar para a ordem.

---

[86] *Ibid*.
[87] *Ibid*.
[88] DE AZEVEDO SOARES, **Mineração de Textos na Coleta Inteligente de Dados na Web**.
[89] *Ibid*.
[90] HVITFELDT, Emil; SILGE, Julia, **Supervised machine learning for text analysis in R**, First edition. Boca Raton: CRC Press, 2022.
[91] GENTZKOW, Matthew; KELLY, Bryan; TADDY, Matt, Text as Data, **Journal of Economic Literature**, v. 57, n. 3, p. 535–574, 2019.
[92] WILKERSON; CASAS, Large-Scale Computerized Text Analysis in Political Science.
[93] BENOIT, Text as Data.

1.4.3.1 Tokenização

Dada uma sequência de caracteres, tokenização é o ato de subdividir texto em pedaços, chamados *tokens*[94]. Apesar de frequentemente esses pedaços coincidirem com palavras, Manning, Raghavan e Schuetze [95] ressaltam que isso não é sempre o caso. Na definição dos autores, um *token* é uma sequência de caracteres de um determinado documento que é agrupada junto como uma unidade semântica útil[96]. Como afirmado por Hvitfeld e Sigel, muitas linguagens, como o chinês, não usam espaçamento entre palavras, então nesse caso os *tokens* não serão palavras[97]. Como o *token* é a unidade de análise semântica, ele teoricamente pode ser uma frase, um parágrafo, uma letra ou um n-grama[98].

Por n-grama, entende-se uma sequência contínua de n itens em uma sequência de texto, normalmente um conjunto de palavras. Por exemplo, a frase "João pega a bola" pode ser dividida em 3 bigramas: {joão, pega}, {pega, a}, {a bola}. Uma vantagem de usar n-gramas em vez de palavras como tokens é que isso preserva a relação sequencial entre palavras. Contudo, isso pode aumentar consideravelmente a complexidade computacional da análise.

O processo de tokenização é normalmente feito usando os espaçamentos entre as palavras e os sinais de pontuação como delimitadores entre *tokens*. Como bem colocado por De Azevedo Soares[99], a pontuação introduz algumas ambiguidades que podem ser complexas para o computador. A título de exemplo, a pontuação pode demarcar uma frase, mas também pode aparecer em abreviações. É comum, nesses casos, substituir as abreviações mais comuns pelas expressões inteiras, ainda na fase de pré-processamento do *corpus*.

Segundo Benoit, é considerado uma boa prática remover pontuação após a tokenização, a fim de retirar do universo de análise tudo aquilo que não adiciona informação ao modelo[100]. Ainda, é normalmente recomendado remover as ditas *stopwords*, *tokens* que não possuem valor semântico, tais como preposições, conjunções, artigos e demais termos que de alguma forma poluam a análise[101]

---

[94] *Ibid*.
[95] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[96] *Ibid*.
[97] HVITFELDT; SILGE, **Supervised machine learning for text analysis in R**.
[98] *Ibid*.
[99] DE AZEVEDO SOARES, **Mineração de Textos na Coleta Inteligente de Dados na Web**.
[100] BENOIT, Text as Data.
[101] *Ibid*.; DE AZEVEDO SOARES, **Mineração de Textos na Coleta Inteligente de Dados na Web**.

De acordo com Manning, Raghavan e Schuetze, a estratégia geral para se identificar a lista de *stop words* a serem removidas, denominada *stop list*, é organizar os termos pela frequência que aparecem no documento[102]. Essa abordagem parte do pressuposto que, normalmente, os termos mais frequentes não adicionarão valor informacional, visto que o que distingue dois objetos é justamente o que eles não têm em comum. Ou, colocado de outra forma, o fato de dois documentos possuírem alta incidência de termos como "e", "de" e "para" possui baixa relevância para sua classificação, enquanto a presença de termos como "norma", "inciso" e "lei" em um documento, mas não em outro, indica que o primeiro provavelmente se trata de um texto jurídico.

Não há regras estritas para se confeccionar uma *stop list*, e a escolha dos termos a serem removidos acaba sendo uma decisão do pesquisador, construída de acordo com o seu objeto de pesquisa. Como apontam Hvitfeld e Silge[103], o contexto é sempre importante para modelagem de texto, de modo que é necessário considerar as informações introduzidas pela incorporação de um termo no modelo. Pronomes, por exemplo, podem introduzir ruído em muitas análises, mas são importantíssimos para se estudar viés de gênero. Bender *et al*.[104] apontam como termos de discurso de ódio eram comumente removidos de listas para treinar modelos de inteligência artificial (para que não reproduzam discurso ofensivo), mas, por óbvio, termos ofensivos podem ser extremamente úteis em uma análise de discurso de ódio ou da experiência de minorias na internet. Ainda, segundo Bender *et al*., quando filtramos termos ditos ofensivos, podemos suprimir o discurso de populações marginalizadas que estejam usando termos considerados ofensivos em outros contextos, o que pode introduzir vieses estruturais no modelo ao suprimir vozes de minorias[105].

Como bem salientado por Manning, Raghavan e Schuetze[106], historicamente, pesquisas com recuperação de informação removiam muitas stop words (200 a 300 termos), depois gradualmente passaram a remover menos (de 7 a 12) e agora há várias pesquisas que não fazem uso algum de *stop words*.

---

[102] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[103] HVITFELDT; SILGE, **Supervised machine learning for text analysis in R**.
[104] BENDER, Emily M. *et al*, On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?, *in*: **Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency**, Virtual Event Canada: ACM, 2021, p. 610–623.
[105] *Ibid*.
[106] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.

1.4.3.2 Lematização e stemming

Documentos podem conter diferentes variações de uma mesma palavra. Por exemplo, "tenho", "tem" e "temos" são conjugações no presente de um mesmo verbo, "ter". Da mesma forma, "rapidamente", "rápido" e "rapidíssimo" remetem a um mesmo conceito de rapidez. *Stemming* e lematização são diferentes formas de lidar com variações de um mesmo termo, reduzindo palavras a seu radical. *Stemming* refere-se a um método mais simples, que simplesmente remove as terminações de palavras, almejando reduzi-las a seu radical[107]. Essa abordagem faz uso de regras rígidas e heurísticas, que sabidamente irão falhar em alguns casos[108]. A fim de ilustrar essa limitação, tomemos os termos ‘caso’ e ‘casa’. Apesar de ambos compartilharem de um mesmo radical, ‘cas’, eles não possuem sentidos semelhantes, de modo que um processo de *stemming* apressado poderia introduzir ruídos na pesquisa. Enquanto técnica, o *stemming* é relativamente simplório, mas é rápido e sua menor precisão não costuma afetar os resultados de forma significativa, sendo, portanto, segundo Grimme e Stewart, o método mais comum[109].

Em direto contraste com o *stemming,* a lematização, como colocado por Hvitfeld e Silge, faz uso de informações contextuais e conhecimento de linguística para reduzir palavras ao seu *lema*, ou forma canônica, evitando assim uma dependência excessiva em heurísticas[110]. Contudo, a lematização precisa de informações sobre a sintaxe de um texto para funcionar[111].

Ambos os métodos mencionados variam de acordo com o idioma do texto[112]. Mesmo usando uma heurística de *stemming*, as regras para obter um radical em inglês seriam diferentes das em português.

1.4.3.3 Vetorização e matriz de frequência de termos

A última etapa da representação quantitativa de um texto é sua conversão em uma representação numérica. Para que possamos aplicar modelagem textual, precisamos representar o texto matematicamente usando características dos *tokens*. De acordo com

---

[107] *Ibid*.
[108] *Ibid*.
[109] *GRIMMER; STEWART, Text as Data.*
[110] HVITFELDT; SILGE, **Supervised machine learning for text analysis in R**.
[111] *Ibid*.
[112] GRIMMER; STEWART, Text as Data.

Bengfort, Bilbro e Ojeda, esse processo é denominado extração de características, ou vetorização[113].

Um método simples para seleção de características é a análise da frequência de *tokens* nos documentos (*frequency-based feature selection*)[114].

Na definição de Grimmer e Stewart, cada documento *i* (*i* = 1, ..., N) é representado como um vetor contendo a frequência de cada *token m*, $W_i = (W_{i1}, W_{i2}, W_{i3}, ..., W_{im})$, em que cada termo W_{im} conta o número de vezes que o m-ésimo *token* aparece no i-ésimo documento[115]. O resultado é uma matriz de frequência de termos denominada matriz de frequência termo-documento (*document-frequency matrix*).

Segundo Grimmer e Stewart[116], essa última etapa pode parecer radical por eliminar muita informação. Sem a ordem, a frase "o cachorro mordeu o homem" é equivalente a "o homem mordeu o cachorro". Contudo, de acordo com esses autores, essa representação é, talvez surpreendentemente, suficiente para se extrair propriedades substantivas de um texto[117]. Existem outras formas de representar documentos que usam a posição das palavras, mas a mais comum é usar o modelo de saco de palavras, que é suficiente para os propósitos deste trabalho. A figura 2, baseada em um exemplo semelhante de Bengfort, Bilbro e Ojeda, ilustra a representação numérica dos textos.

[113] BENGFORT; BILBRO; OJEDA, Applied Text Analysis with Python.
[114] AGGARWAL, Charu C.; ZHAI, ChengXiang (Orgs.), **Mining Text Data**, Boston, MA: Springer US, 2012; MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[115] GRIMMER; STEWART, Text as Data.
[116] *Ibid*.
[117] *Ibid*.

Figura 2: Representação numérica dos textos. Cada entrada corresponde à frequência da repetição de um termo.

Na figura, cada entrada corresponde ao número de repetições de um termo no documento. Termos que não ocorrem em um dado texto, mas que constam de outros objetos de um *corpus* são preenchidos com o número 0.

1.4.3.4 Frequência de termos

Uma dificuldade introduzida pela medida de frequência de termos, como colocado por Manning, Raghavan e Schuetze, é que nem toda palavra é igualmente importante[118]. Inclusive, muitas vezes palavras que aparecem frequentemente são apenas ruído. A fim de combater esse problema, podemos introduzir pesos a cada termo. Nessa toada, nós podemos associar a cada termo em um documento um peso baseado na sua frequência em um documento, ou *term frequency*[119]. Isso é baseado na intuição de que termos que apareçam mais vezes em um texto tem uma importância maior.

$$\mathrm{tf}(t,\ d) = f_{t,\,d}$$

[118] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[119] *Ibid*.

Sendo *tf* a frequência *f* de um termo *t* no documento *d*.

Contudo, o uso da frequência de um termo incorre em um problema. Palavras comuns, aquelas frequentemente eliminadas *stop words*, aparecerão com alta frequência em múltiplos documentos. Por exemplo, o termo "parte" ou "processo" constará de virtualmente qualquer peça jurídica, mas isso não carrega muita informação para nosso modelo, é ruído. Para remediar esse problema, é utilizado uma ponderação adicional pelo inverso da frequência nos documentos, isto é, a quantidade de documentos de nosso *corpus* que contém o termo. Termos mais raros, que aparecem em poucos documentos, tendem a carregar maior informação que termos que constam de todos os documentos.

$$\mathrm{idf_t} = \log \frac{N}{\mathrm{df_t}}$$

Combinando as duas ponderações, chegamos à estatística de frequência do termo-inverso da frequência nos documentos, ou *tf-idf* (*term frequency-inverse document frequency*).

$$\text{tf-idf}_{t,d} = \mathrm{tf_{t,d}} \times \mathrm{idf_t}$$

O resultado é que a estatística de *tf-idf* irá atribuir um peso maior quando *t* ocorre muitas vezes em um número pequeno de documentos e menor quando o termo ocorre poucas vezes em um documento, ou ocorre em muitos ou todos os documentos[120]. O principal *insight* da estatística de *tf-idf*, de acordo com Bengfort, Bilbro e Ojeda, é que é mais provável que o sentido de um texto esteja refletido em termos mais raros[121].

Bengfort, Bilbro e Ojeda[122] dão o exemplo de termos de baseball. Palavras como "arremessador", "*strike*" e "taco" servem para distinguir que um texto é sobre baseball e não sobre outro esporte, enquanto palavras mais genéricas como "pontuação" e "jogo" não serviriam muito bem para diferenciar um texto de baseball de outros tipos de esporte.

[120] *Ibid*.
[121] BENGFORT; BILBRO; OJEDA, Applied Text Analysis with Python.
[122] *Ibid*.

1.4.3.5 Lei de Zipf

A lei de Zipf consiste na observação de que, em um dado *corpus*, a frequência de uma palavra tende a ser inversamente proporcional ao seu ranking de frequência[123]. A lei é atribuída ao linguista George Kingsley Zipf que, em 1932, observou que a frequência de uma palavra em um documento é uma função de potência de seu ranking de frequência, formulada como:

$$f(r) \propto \frac{1}{r^{\alpha}}$$

Em que *f* é a frequência da palavra, *r* é seu ranking da frequência e α é uma constante[124]. Assim, se o termo mais frequente ocorre *k* vezes, o segundo termo ocorrerá $k/2$, o terceiro termo $k/3$ e assim em diante[125]. A frequência, como bem observado por Manning, Raghavan e Schuetze, cairá rapidamente conforme o ranking aumentar[126].

A lei de Zipf é uma observação de um fenômeno reiterado, não um teorema matemático. Dito isso, o padrão se repete reiteradamente na linguística computacional[127]. Yu *et al.*[128] testaram a lei em um *corpus* contendo 50 línguas diferentes e o padrão é notável. Os resultados podem ser observados na figura 3.

---

[123] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[124] PIANTADOSI, Steven T., Zipf's word frequency law in natural language: A critical review and future directions, **Psychonomic Bulletin & Review**, v. 21, n. 5, p. 1112–1130, 2014.
[125] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[126] *Ibid*.
[127] PIANTADOSI, Zipf's word frequency law in natural language; MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[128] YU, Shuiyuan; XU, Chunshan; LIU, Haitao, Zipf's law in 50 languages: its structural pattern, linguistic interpretation, and cognitive motivation, **CoRR**, v. abs/1807.01855, 2018.

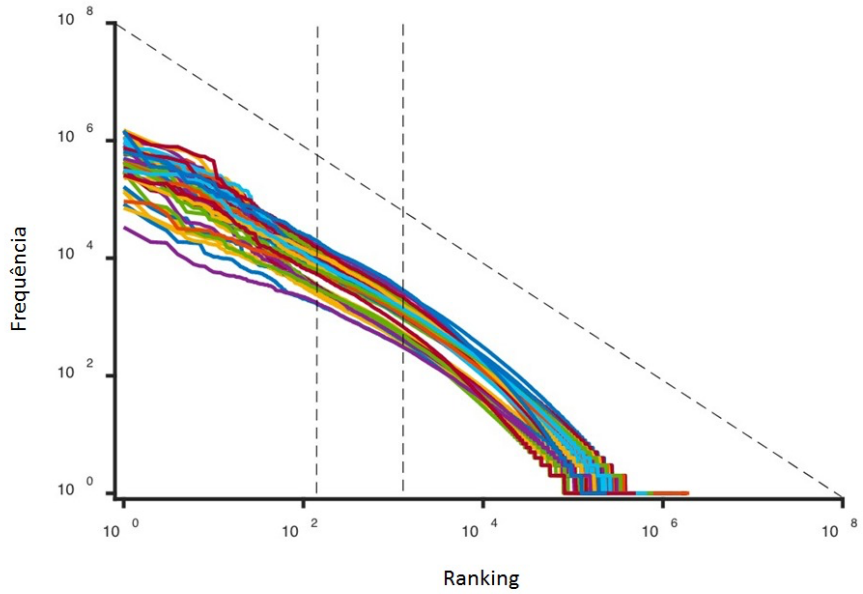


Figura 3: Curva representando a relação entre frequência e ranking de termos, onde observa-se uma relação inversamente proporcional, como preconizado pela Lei de Zipf. Traduzida de Yu et al (2018).

1.5 Algoritmos de categorização de documentos

De acordo com Grimmer e Stewart, a categorização textual é o processo de atribuir categorias a um determinado documento de texto[129]. Um modelo de categorização atribui categorias a documento baseado em alguns critérios.

Um dos usos cotidianos de categorização, comumente explorado como exemplo na literatura, é a criação de filtros de *spam*[130]. Ao receber um e-mail, provedores de e-mails tentam classificar os documentos automaticamente, fazendo uso de seu conteúdo, em uma de duas categorias: *spam* e não *spam*. Essa detecção não é feita pela mera presença ou ausência de determinado termo, mas por análises sofisticadas de frequência e estilo que permitem que provedores identifiquem *spam* corretamente na maioria das vezes.

Apesar de serem essencialmente sinônimas, a literatura tende a empregar "categorização" e "classificação" de forma pouco consistente. Categorização é

[129] GRIMMER; STEWART, Text as Data; MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[130] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.

normalmente usada como gênero do qual classificação é espécie, mas há quem use como sinônimos, como Jo[131]. Classificação sempre se refere a um tipo específico de categorização por aprendizado de máquina, o aprendizado supervisionado[132]. Supervisionado porque usa um conjunto de categorias previamente estabelecido. A filtragem de spam, por exemplo, é um tipo de aprendizado supervisionado. O algoritmo é treinado com centenas de milhares de exemplos de *spam*, previamente identificados por humanos, e então "aprende" a identificar *spam* sozinho[133]. Em direto contraste, o método do *clustering* agrupa documentos de acordo com critérios de coerência interna, mas sem que eles tenham sido associados previamente a classes.

Segundo Grimmer e Stewart[134], o emprego de um método ou outro depende dos objetivos da pesquisa. Em casos em que os pesquisadores conhecem previamente as categorias que querem identificar, como é frequente nas ciências sociais, a classificação é normalmente o método mais eficiente para agrupar documentos[135]. Já nos demais casos, em que a pesquisa objetiva descobrir novos padrões ocultos dentro do texto, métodos não supervisionados como o *clustering* serão preferíveis[136].

1.5.1 Aprendizado supervisionado

De acordo com Benoit, o aprendizado supervisionado parte do pressuposto de que o algoritmo de categorização será suprido com documentos previamente classificados por pesquisadores humanos, de modo que ele possa associar características desses textos a categorias e associar o resultado desse aprendizado a textos sobre os quais o pesquisador não tem conhecimento[137]. Isto é, há um conjunto de documentos d_t sobre o qual já se tem informação, o algoritmo irá extrair padrões dessa informação e tentar classificar um outro conjunto de documentos d_k cuja classificação o pesquisador desconhece. Retomando o exemplo do *spam*, o conteúdo dos e-mails previamente identificados como *spam* será usado para que o modelo possa identificar futuros e-mails de *spam*, cujo conteúdo é naturalmente desconhecido.

---

[131] JO, Taeho, **Text Mining**, Cham: Springer International Publishing, 2019.
[132] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[133] BENOIT, Text as Data.
[134] GRIMMER; STEWART, Text as Data.
[135] *Ibid*.
[136] *Ibid*.
[137] BENOIT, Text as Data.

Segundo Manning, Raghavan e Schuetze[138], em um problema de classificação, há uma descrição d ∈ *X*, sendo *X* um espaço de documentos, e um conjunto fixo de classes $C = \{c_1, c_2, ..., c_j\}$. Há um conjunto de treinamento *D* de documentos já classificados que é um subconjunto de $X \times C$[139]. Podemos então criar uma função de classificação γ que associe esses documentos às classes em C[140]:

$$\gamma: X \rightarrow C$$

Há diversos algoritmos de classificação que podem ser empregados para essa tarefa, tais como classificadores SVM (*support-vector machine*), classificadores de redes neurais e árvores de decisão[141]. A fim de elucidar o funcionamento do aprendizado supervisionado, passaremos a discutir os pormenores da operação de um modelo simples de classificação, o classificador Bayes ingênuo, ou *naive Bayes*.

1.5.1.1 Classificador de Bayes ingênuo

O classificador multinomial de Bayes ingênuo é um classificador probabilístico de texto extremamente simples. Como definido por Grimmer e Stewart[142], cada documento *d* é representado por um vetor, de dimensões *M*, em que cada elemento corresponde a frequência de palavras únicas em um texto, tal como já ilustrado na figura 2. Um subconjunto da amostra de documentos, o conjunto de treinamento (*training set*), é usado para alimentar o algoritmo, que então pode calcular a probabilidade de que um dos documentos restantes pertença a uma das categorias do *training set*. Assim, será possível computar a probabilidade de um documento *d* pertencer a uma categoria *k* dado um perfil de palavras únicas W_i[143]. Dado um conjunto de *k* classes em *C*, o classificador devolverá a classe $\hat{C}$ com maior probabilidade posterior, dado um determinado perfil de probabilidade.

---

[138] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[139] *Ibid*.
[140] *Ibid*.
[141] AGGARWAL; ZHAI (Orgs.), **Mining Text Data**.
[142] GRIMMER; STEWART, Text as Data.
[143] *Ibid*.

A estimativa de máxima verossimilhança de C_k poderá então ser inferida do conjunto de treinamento, contanto que ele seja representativo do *corpus*[144]. Do contrário, sendo um modelo supervisionado, não há como esperar bons resultados a partir de um conjunto de treinamento muito deficiente.

O classificador de Bayes ingênuo é dito "ingênuo" porque supõe que as probabilidades $P(W_i|C_k)$ são independentes para C_k, o que, segundo Benoit, é quase sempre falso[145]. Contudo, apesar de ser um modelo incorreto, ele pode ser surpreendentemente útil para classificar textos[146]. Alguns filtros de *spam* usam modelos baseados em um classificador de Bayes ingênuo[147].

1.5.2 Aprendizado não supervisionado

Diferentemente do aprendizado supervisionado, métodos de aprendizado não supervisionados não fazem uso de um conjunto de treinamento. Assim, uma classificação objetiva replicar uma distinção categórica feita por um supervisor humano, ao passo que métodos não supervisionados não usam essa categorização prévia para "ensinar" o modelo[148]. Em vez disso, diferenças nas características textuais são usadas com base em suas correlações internas, sem uso de conhecimento externo[149].

Como apontam Bengfort *et al.*[150], muitas vezes *corpora* não possuem classificações já prontas, então a única opção disponível ao pesquisador é ou ler e classificar tudo manualmente, o que é quase sempre proibitivo em termos de tempo e dinheiro, ou usar métodos não supervisionados.

A forma mais comum de aprendizado não supervisionado é o *clustering*, ou agrupamento[151]. De acordo com Aggarwal e Zhai, o problema que *clustering* tenta resolver é encontrar grupo de objetos similares nos dados[152]. Como afirmam Bengfort *et al.*[153], algoritmos de *clustering* tentam encontrar estruturas latentes ou temas em dados não rotulados, organizando os dados entre grupos dissimilares. Assim, a medida mais

---

144 *Ibid*.
145 BENOIT, Text as Data.
146 *Ibid*.
147 BENGFORT; BILBRO; OJEDA, Applied Text Analysis with Python.
148 MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
149 BENOIT, Text as Data.
150 BENGFORT; BILBRO; OJEDA, Applied Text Analysis with Python.
151 MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
152 AGGARWAL; ZHAI (Orgs.), **Mining Text Data**.
153 BENGFORT; BILBRO; OJEDA, Applied Text Analysis with Python.

importante para um algoritmo de *clustering* é a distância entre documentos[154]. A figura 4 ilustra esse processo.

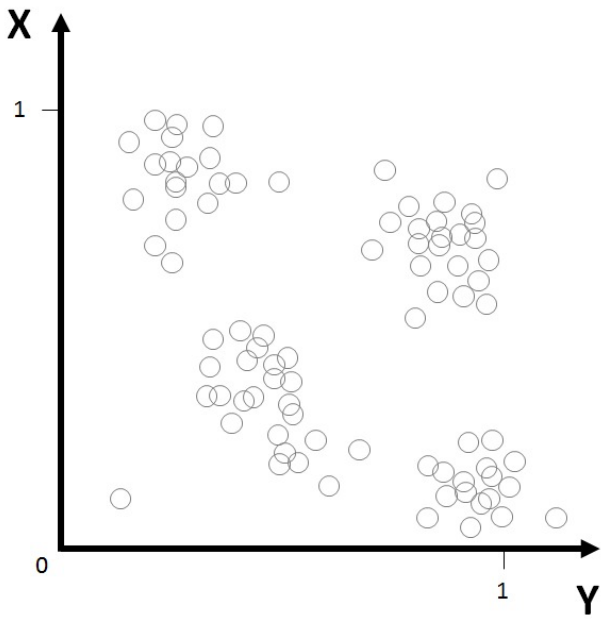


Figura 4: Representação gráfica de *clustering*, onde cada aglomerado de pontos representa um *cluster*.

Segundo Manning, Raghavan e Schuetze[155], o fundamento do algoritmo de *clustering* é a hipótese de que documentos no mesmo cluster se comportam de forma semelhante. A hipótese diz que, se um documento de um cluster é relevante para uma determinada pesquisa, então é provável que documentos no mesmo cluster também serão[156].

Ainda, Jo[157] distingue entre vários tipos de *clustering*. Para os propósitos deste trabalho, é relevante abordar quatro categorias: *clustering* duro e mole e *clustering* hierárquico e não hierárquico. Diz-se duro o *clustering* em que cada documento só pode integrar um cluster, enquanto no mole itens podem ocupar mais de um cluster[158]. Ainda,

[154] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[155] *Ibid*.
[156] *Ibid*.
[157] JO, **Text Mining**.
[158] *Ibid*.

clustering não hierárquico é aquele em que clusters se situam desordenados em um plano, sem nenhuma estrutura entre si. Em direto contraste, o *clustering* hierárquico organiza clusters em uma estrutura de árvore, com uma hierarquia entre eles[159].

1.5.2.1 Similaridade por cosseno

Segundo Aggarwal e Zhai, a computação da similaridade textual é um problema fundamental na recuperação da informação[160]. Uma vez que textos estejam representados quantitativamente como vetores numéricos, é possível usar medidas da distância entre vetores para medir a semelhança entre textos, como a distância euclidiana e o índice de distância de Jaccard[161].

Um método muito popular para computar a taxa de similaridade entre dois vetores de texto é o método da semelhança do cosseno. É possível usar o ângulo do cosseno entre dois vetores para medir o seu grau de semelhança, como demonstrado na figura 5. Uma vantagem dessa abordagem, em contraste com, por exemplo, a magnitude do vetor de diferença entre dois vetores de texto, é que ela não é afetada por diferenças no tamanho entre dois vetores[162].

[159] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval; JO, **Text Mining**.
[160] AGGARWAL; ZHAI (Orgs.), **Mining Text Data**.
[161] A distância de Jaccard define semelhança entre dois conjuntos como o quociente da sua interseção e de sua união BENGFORT; BILBRO; OJEDA, Applied Text Analysis with Python..
[162] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.

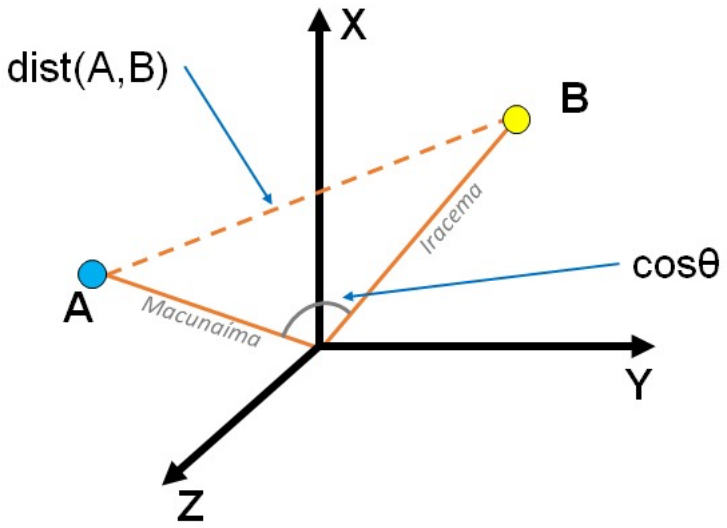


Figura 5: Representação do método da semelhança do cosseno. A distância entre os vetores representando as duas obras corresponde ao grau de semelhança.

Assim, para medir a taxa de similaridade entre dois documentos d_1 e d_2, representados por $\vec{V}(d_1)$ e $\vec{V}(d_2)$, é só aplicar a fórmula:

$$sim(d_1, d_2) = \frac{\vec{V}(d_1) \cdot \vec{V}(d_2)}{|\vec{V}(d_1)||\vec{V}(d_2)|}$$

Em que o numerador representa o produto escalar entre $\vec{V}(d_1)$ e $\vec{V}(d_2)$ e o denominador o produto de suas magnitudes[163].

O valor resultante do cálculo da medida de similaridade por cosseno variará entre 0 (caso sejam ortogonais) e 1 (caso sejam paralelos). Assim, quanto mais semelhante forem dois vetores, mais próximos estarão de uma relação de paralelismo e mais próximo o cosseno será zero[164]. A ilustração constante da figura 5 mostra a intuição por trás do método.

[163] *Ibid*.
[164] BENGFORT; BILBRO; OJEDA, Applied Text Analysis with Python.

Vale frisar que, como ressaltado por Aggarwal e Zhai[165], a medida da semelhança do cosseno é muitas vezes combinada com a ponderação pela estatística *tf-idf*. Assim é possível computar a semelhança entre dois vetores documento sem considerar todas as palavras como igualmente relevantes.

1.5.2.2 *Clustering* hierárquico aglomerante

O *clustering* hierárquico aglomerante (ou HAC, do inglês *hierarchical agglomerative clustering*) é uma modalidade de clustering hierárquico, isto é, aquele que produz uma hierarquia ordenada entre os grupos de documentos, caracterizada por juntar (aglomerar) cluster de acordo com critérios de semelhança[166]. O algoritmo HAC é dito "de baixo para cima", pois toma inicialmente todo cluster como isolado e vai agrupando grupos semelhantes até formar um cluster só com todos os documentos[167]. Confira-se o dendrograma da Figura 6, adaptado de Jo[168], para visualizar a ordem do processo.

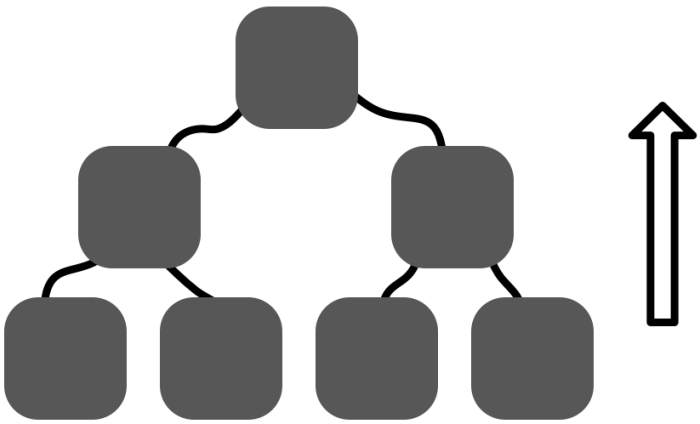


Figura 6: Ilustração da ordem *bottom-up* de um algoritmo de *clustering* aglomerante hierárquico.

[165] AGGARWAL; ZHAI (Orgs.), **Mining Text Data**.
[166] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[167] *Ibid*.
[168] JO, **Text Mining**.

Como salientado por Manning, Raghavan e Schuetze [169], o HAC não requer uma quantidade pré-determinada de *clusters*, mas, em algumas ocasiões, nós queremos formar grupos disjuntos, como em um *clustering* não hierárquico. Nesses casos, o dendrograma deve ser cortado em algum ponto, a fim de formar grupos. Os autores sugerem, entre outros, dois pontos de corte: cortar a partir de determinado grau de semelhança, que corresponderá a determinada altura; ou ainda pré-determinar um número k de *clusters*[170].

Há diversos critérios para calcular a semelhança entre documentos no método HAC. Dentre os muitos mencionados por Manning, Raghavan e Schuetze [171], cumpre ressaltar aqui três: *single-link*, em que a similaridade entre dois *clusters* é a similaridade entre seus dois elementos mais próximos; *complete-link*, em que o critério é a similaridade entre seus dois membros mais diferentes; e o método de Ward, que busca minimizar a variância dentro de cada cluster.

1.5.2.3 Modelagem de tópicos

De acordo com Blei[172], algoritmos de modelagem de tópicos são métodos estatísticos que analisam as palavras de um texto original para encontrar temas que permeiam o texto e descobrir como esses temas são conectados uns com os outros. O primeiro modelo de tópicos, e o mais popular na literatura especializada, é o modelo de alocação latente de Dirichlet, ou LDA (do inglês, *latent Dirichlet allocation*).

Benoit[173] destaca que modelos de tópicos são modelos relativamente simples para descrever a relação entre grupos de palavras que ocorrem juntas ("tópicos") e sua relação com os documentos que as contém. Ainda, não há necessidade de supervisão humana, para além da definição prévia do número de tópicos.

Grimmer e Stewart[174] apontam que modelos LDA supõe que cada documento é uma mistura de tópicos, e, por isso, *topic models* são ditos modelos de membrasia mista, ou *mixed-membership models*, pois permitem que um mesmo documento contenha vários tópicos.

---

[169] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval.
[170] *Ibid*.
[171] *Ibid*.
[172] BLEI, David M., Probabilistic topic models, **Communications of the ACM**, v. 55, n. 4, p. 77–84, 2012.
[173] BENOIT, Text as Data.
[174] GRIMMER; STEWART, Text as Data.

Os resultados da modelagem de tópicos são diretamente afetados pelo valor iniciado (ou semente) escolhido. Isso é, os resultados do modelo vão variar de estocasticamente de acordo com parâmetros definidos pelo pesquisador, de modo que um mesmo pode fornecer tópicos diferentes. Devido à sua natureza aleatória, modelos LDA são primariamente exploratórios, sendo utilizados para se ter uma primeira aproximação dos conteúdos presentes em um *corpus*.

1.6 Validação

A última etapa da análise textual computadorizada é a validação, isso é, a avaliação humana da performance da máquina. Ela é especialmente importante quando se trata de métodos não supervisionados de aprendizado. Benoit[175] coloca a questão da seguinte forma: "tendo pulado o julgamento humano na fase de análise, devemos trazer de volta nosso julgamento na conclusão do processo para compreender os resultados. Se nosso bom senso indicar que há algo de errado, podemos decidir ajustar a máquina ou seus insumos e repetir o processo até conseguir resultados melhores".

Grimmer e Stewart ressaltam[176] que a validação pode ocorrer de várias formas, que o importante é evitar o uso cego de um método sem uma etapa de validação. Isso é mais grave com softwares comerciais de análise de texto, que muitas vezes produzem resultados que não podem ser validados.

No contexto deste trabalho, a validação dos resultados da análise computadorizada foi feita por meio de um trabalho qualitativo com os julgamentos em lista. Previamente à elaboração dos modelos, foi analisado o registro audiovisual de todos os julgamentos em lista de ADI pelo Supremo Tribunal Federal no ano de 2018. O conhecimento construído na fase qualitativa foi fundamental para se avaliar a performance *ex post* dos algoritmos de análise automatizada de conteúdo.

***

[175] BENOIT, Text as Data.
[176] GRIMMER; STEWART, Text as Data.

Capítulo 2 – Coleta de dados e análise qualitativa

Este capítulo examina o processo de construção da base de dados usada neste trabalho, bem como a análise qualitativa das sessões de julgamento do Supremo Tribunal Federal. Em um primeiro momento, discute-se a estratégia de coleta de dados utilizada no trabalho. Em seguida, detalha-se as escolhas tomadas na construção dos robôs de raspagem, bem como as decisões feitas no registro audiovisual das sessões do STF.

2.1 Processo de raspagem

Os dados primários foram coletados por raspagem, isto é, a aquisição automatizada de dados, comumente por meio de um programa, denominado *crawler*, que solicita informações para um servidor *web* e então analisa esses dados para obter o conteúdo desejado[177]. O *crawler* simula um navegador a fim de realizar várias solicitações ao sítio eletrônico e obter as informações almejadas.

Os dados coletados, por sua vez, foram obtidos por meio de uma consulta ao repositório de jurisprudência do Supremo Tribunal Federal[178], que disponibiliza em seu domínio eletrônico acórdãos publicados. O *crawler* em questão foi elaborado na linguagem Python, por meio do pacote *scrapy*, e foi capaz de realizar o download de arquivos em formato PDF de todas as 2136 ADIs publicadas no período entre 1989-04-14 e 2019-12-31. Ainda, o *crawler* extraiu os seguintes metadados dos processos:

- Número
- Data do julgamento
- Data da publicação
- Partes
- Relator
- Ementa
- Decisão
- Indexação
- Legislação

[177] MANNING; RAGHAVAN; SCHUETZE, Introduction to Information Retrieval; MITCHELL, Web Scraping with Python.
[178] jurisprudencia.stf.jus.br

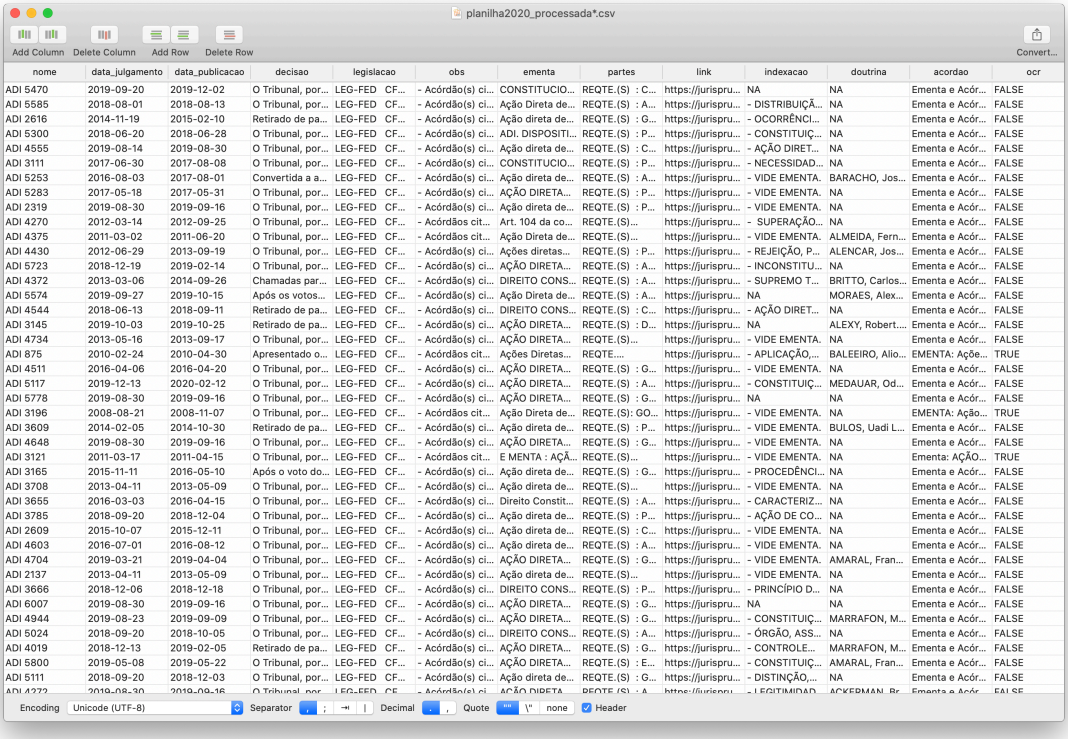
- Acórdãos no mesmo sentido
- Doutrina
- Observações

Finda essa primeira extração, elaborou-se um *crawler* adicional para identificar quais desses processos foram julgados em lista. Essa informação, apesar de não constar dos acórdãos, constava das pautas de julgamento do plenário do Tribunal. Para cada data em que há sessão, constam os processos que serão julgados no dia, bem como as listas da relatoria de cada Ministro. Como exposto anteriormente, essas listas são majoritariamente compostas de REs e HCs, mas contém também processos de controle concentrado.

Finalmente, a fim de descobrir quais ADIs foram julgadas em lista, construiu-se novo *crawler* para inspecionar cada lista, desde 2004, para identificar ADIs. Foram localizados 246 ADIs julgadas em lista, ou 11.51%, sendo que a primeira foi a referida ADI 5075, julgada em 19 de Agosto de 2015. Os processos em lista são 41.27% das ADIs julgadas desde 2015.

O levantamento descrito acima não teria sido possível sem o uso de técnicas automatizadas de análise de dados. O trabalho necessário para consultar, manualmente, todas as informações coletadas nessa pesquisa excederia em muito o razoável, mesmo para uma pesquisa empírica. Ainda, a coleta manual de informações estaria sujeita a erro humano, até pela natureza repetitiva da tarefa, que aumenta a probabilidade de erro por distração.

A figura 7 retrata o formato adotado na construção da base de dados:



nome	data_julgamento	data_publicacao	decisao	legislacao	obs	ementa	partes	link	indexacao	doutrina	acordao	ocr
ADI 5470	2019-09-20	2019-12-02	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	CONSTITUCIO...	REQTE.(S) : C...	https://jurispru...	NA	NA	Ementa e Acór...	FALSE
ADI 5585	2018-08-01	2018-08-13	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	Ação Direta de...	REQTE.(S) : A...	https://jurispru...	- DISTRIBUIÇÃ...	NA	Ementa e Acór...	FALSE
ADI 2616	2014-11-19	2015-02-10	Retirado de pa...	LEG-FED CF...	- Acórdão(s) ci...	Ação direta de...	REQTE.(S) : G...	https://jurispru...	- OCORRÊNCI...	NA	Ementa e Acór...	FALSE
ADI 5300	2018-06-20	2018-06-28	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	ADI. DISPOSITI...	REQTE.(S) : P...	https://jurispru...	- CONSTITUIÇ...	NA	Ementa e Acór...	FALSE
ADI 4555	2019-08-14	2019-08-30	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	Ação direta de...	REQTE.(S) : C...	https://jurispru...	- AÇÃO DIRET...	NA	Ementa e Acór...	FALSE
ADI 3111	2017-06-30	2017-08-08	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	CONSTITUCIO...	REQTE.(S) : P...	https://jurispru...	- NECESSIDAD...	NA	Ementa e Acór...	FALSE
ADI 5253	2016-08-03	2017-08-01	Convertida a a...	LEG-FED CF...	- Acórdão(s) ci...	Ação direta de...	REQTE.(S) : A...	https://jurispru...	- VIDE EMENTA.	BARACHO, Jos...	Ementa e Acór...	FALSE
ADI 5283	2017-05-18	2017-05-31	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S) : P...	https://jurispru...	- VIDE EMENTA.	NA	Ementa e Acór...	FALSE
ADI 2319	2019-08-30	2019-09-16	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	Ação direta de...	REQTE.(S) : P...	https://jurispru...	- VIDE EMENTA.	NA	Ementa e Acór...	FALSE
ADI 4270	2012-03-14	2012-09-25	O Tribunal, por...	LEG-FED CF...	- Acórdãos cit...	Art. 104 da co...	REQTE.(S)...	https://jurispru...	- SUPERAÇÃO...	NA	Ementa e Acór...	FALSE
ADI 4375	2011-03-02	2011-06-20	O Tribunal, por...	LEG-FED CF...	- Acórdãos cit...	Ação Direta de...	REQTE.(S)...	https://jurispru...	- VIDE EMENTA.	ALMEIDA, Fern...	Ementa e Acór...	FALSE
ADI 4430	2012-06-29	2013-09-19	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	Ações diretas...	REQTE.(S) : P...	https://jurispru...	- REJEIÇÃO, P...	ALENCAR, Jos...	Ementa e Acór...	FALSE
ADI 5723	2018-12-19	2019-02-14	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S) : A...	https://jurispru...	- INCONSTITU...	NA	Ementa e Acór...	FALSE
ADI 4372	2013-03-06	2014-09-26	Chamadas par...	LEG-FED CF...	- Acórdão(s) ci...	DIREITO CONS...	REQTE.(S) : A...	https://jurispru...	- SUPREMO T...	BRITTO, Carlos...	Ementa e Acór...	FALSE
ADI 5574	2019-09-27	2019-10-15	Após os votos...	LEG-FED CF...	- Acórdão(s) ci...	Ação Direta de...	REQTE.(S) : A...	https://jurispru...	NA	MORAES, Alex...	Ementa e Acór...	FALSE
ADI 4544	2018-06-13	2018-09-11	Retirado de pa...	LEG-FED CF...	- Acórdão(s) ci...	DIREITO CONS...	REQTE.(S) : C...	https://jurispru...	- AÇÃO DIRET...	NA	Ementa e Acór...	FALSE
ADI 3145	2019-10-03	2019-10-25	Retirado de pa...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S) : D...	https://jurispru...	NA	ALEXY, Robert....	Ementa e Acór...	FALSE
ADI 4734	2013-05-16	2013-09-17	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S)...	https://jurispru...	- VIDE EMENTA.	NA	Ementa e Acór...	FALSE
ADI 875	2010-02-24	2010-04-30	Apresentado o...	LEG-FED CF...	- Acórdãos cit...	Ações Diretas...	REQTE....	https://jurispru...	- APLICAÇÃO,...	BALEEIRO, Alio...	EMENTA: Ação...	TRUE
ADI 4511	2016-04-06	2016-04-20	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S) : G...	https://jurispru...	- VIDE EMENTA.	NA	Ementa e Acór...	FALSE
ADI 5117	2019-12-13	2020-02-12	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S) : A...	https://jurispru...	- CONSTITUIÇ...	MEDAUAR, Od...	Ementa e Acór...	FALSE
ADI 5778	2019-08-30	2019-09-16	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S) : G...	https://jurispru...	NA	NA	Ementa e Acór...	FALSE
ADI 3196	2008-08-21	2008-11-07	O Tribunal, por...	LEG-FED CF...	- Acórdãos cit...	Ação Direta de...	REQTE.(S): GO...	https://jurispru...	- VIDE EMENTA.	NA	EMENTA: Ação...	TRUE
ADI 3609	2014-02-05	2014-10-30	Retirado de pa...	LEG-FED CF...	- Acórdão(s) ci...	Ação direta de...	REQTE.(S) : P...	https://jurispru...	- VIDE EMENTA.	BULOS, Uadi L...	Ementa e Acór...	FALSE
ADI 4648	2019-08-30	2019-09-16	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S) : G...	https://jurispru...	- VIDE EMENTA.	NA	Ementa e Acór...	FALSE
ADI 3121	2011-03-17	2011-04-15	O Tribunal, por...	LEG-FED CF...	- Acórdãos cit...	E MENTA : AÇÃ...	REQTE.(S)...	https://jurispru...	- VIDE EMENTA.	NA	Ementa: AÇÃO...	TRUE
ADI 3165	2015-11-11	2016-05-10	Após o voto do...	LEG-FED CF...	- Acórdão(s) ci...	Ação direta de...	REQTE.(S) : G...	https://jurispru...	- PROCEDÊNCI...	NA	Ementa e Acór...	FALSE
ADI 3708	2013-04-11	2013-05-09	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	Ação direta de...	REQTE.(S)...	https://jurispru...	- VIDE EMENTA.	NA	Ementa e Acór...	FALSE
ADI 3655	2016-03-03	2016-04-15	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	Direito Constit...	REQTE.(S) : A...	https://jurispru...	- CARACTERIZ...	NA	Ementa e Acór...	FALSE
ADI 3785	2018-09-20	2018-12-04	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	Ação direta de...	REQTE.(S) : P...	https://jurispru...	- AÇÃO DE CO...	NA	Ementa e Acór...	FALSE
ADI 2609	2015-10-07	2015-12-11	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	Ação direta de...	REQTE.(S) : C...	https://jurispru...	- VIDE EMENTA.	NA	Ementa e Acór...	FALSE
ADI 4603	2016-07-01	2016-08-12	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	Ação direta de...	REQTE.(S) : A...	https://jurispru...	- VIDE EMENTA.	NA	Ementa e Acór...	FALSE
ADI 4704	2019-03-21	2019-04-04	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S) : G...	https://jurispru...	- VIDE EMENTA.	AMARAL, Fran...	Ementa e Acór...	FALSE
ADI 2137	2013-04-11	2013-05-09	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	Ação direta de...	REQTE.(S)...	https://jurispru...	- VIDE EMENTA.	NA	Ementa e Acór...	FALSE
ADI 3666	2018-12-06	2018-12-18	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	DIREITO CONS...	REQTE.(S) : P...	https://jurispru...	- PRINCÍPIO D...	NA	Ementa e Acór...	FALSE
ADI 6007	2019-08-30	2019-09-16	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S) : G...	https://jurispru...	NA	NA	Ementa e Acór...	FALSE
ADI 4944	2019-08-23	2019-09-09	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S) : G...	https://jurispru...	- CONSTITUIÇ...	MARRAFON, M...	Ementa e Acór...	FALSE
ADI 5024	2018-09-20	2018-10-05	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	DIREITO CONS...	REQTE.(S) : A...	https://jurispru...	- ÓRGÃO, ASS...	NA	Ementa e Acór...	FALSE
ADI 4019	2018-12-13	2019-02-05	Retirado de pa...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S) : G...	https://jurispru...	- CONTROLE...	MARRAFON, M...	Ementa e Acór...	FALSE
ADI 5800	2019-05-08	2019-05-22	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	AÇÃO DIRETA...	REQTE.(S) : E...	https://jurispru...	- CONSTITUIÇ...	AMARAL, Fran...	Ementa e Acór...	FALSE
ADI 5111	2018-09-20	2018-12-03	O Tribunal, por...	LEG-FED CF...	- Acórdão(s) ci...	Ação direta de...	REQTE.(S) : G...	https://jurispru...	- DISTINÇÃO,...	NA	Ementa e Acór...	FALSE

Figura 7: Base de dados construída durante o trabalho. Cada entrada representa uma ADI. As colunas representam metadados correspondentes a cada processo.

A íntegra do texto extraído dos acórdãos consta da coluna ‘acordao’. Cada documento é representado por uma entrada, e cada coluna representa um atributo da ADI.

A figura 8 compara o número de julgamentos em lista com os julgamentos tradicionais. Pode-se observar um uso crescente dos julgamentos em lista, até o advento do julgamento de ADI por plenário virtual. A figura 9 mostra a proporção de ADIs julgadas em lista, por ano.

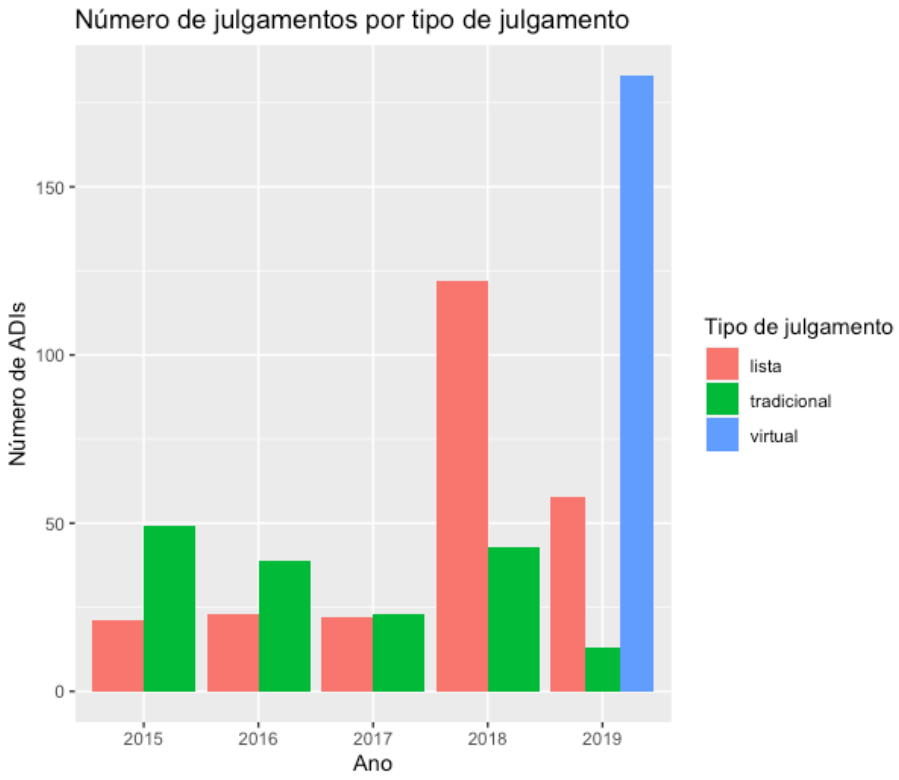


Figura 8: Número de ADIs por ano, agrupadas por modo de julgamento. Em vermelho observa-se o número de ADIs julgadas em lista. A cor azul representa os julgamentos em plenário virtual e a verde os julgamentos tradicionais.

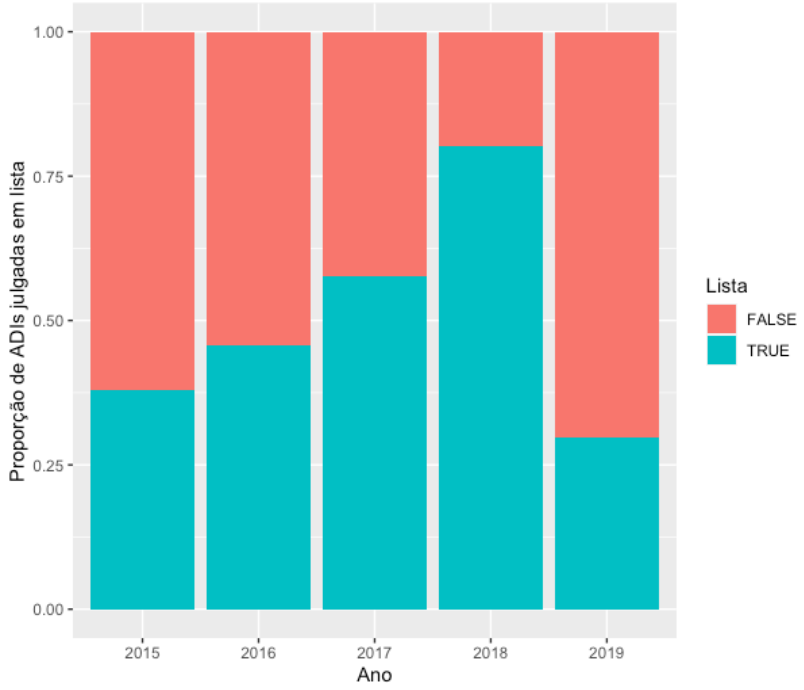


Figura 9: Proporção anual de ADIs julgadas por modo de julgamento. Em azul (TRUE) temos a proporção de ADIs julgadas em lista. Em vermelho (FALSE) temos os demais modos de julgamento.

Como se pode depreender da figura 9, a proporção de ADIs julgadas em lista subiu a cada ano, desde sua introdução em 2015, até 2019, quando surge a possibilidade de julgamento de ADI em plenário virtual. A natureza assíncrona e diferida do plenário virtual diminui a vantagem comparativa de se realizar um julgamento em lista, visto que o ambiente do plenário virtual pode ser ainda mais célere. É interessante, contudo, notar que o julgamento em lista não desapareceu após a introdução do julgamento de ADI em plenário virtual, o que sugere que essa modalidade oferece ao Relator ou ao plenário vantagens em relação à modalidade virtual.

Quais seriam essas vantagens? A análise qualitativa das sessões de julgamento em lista sugere um possível caminho. Em alguns dos julgamentos analisados, o julgamento se iniciou em rito abreviado, como os demais julgamentos em lista, mas converteu-se em um julgamento tradicional, com amplo debate, após manifestação de algum dos ministros. A título de exemplo, apesar de nominalmente terem sido julgadas em lista, as ADIs 4332 e 5109 tiveram duração de 20 minutos e 8 segundos e 35 minutos e 54 segundos, respectivamente. Assim, o julgamento em lista ofereceria em relação ao plenário virtual à possibilidade de conversão espontânea em julgamento tradicional. Contudo, a previsão

de pedido de destaque constante do parágrafo terceiro do artigo 21-B do Regimento Interno do Supremo Tribunal Federal, que prevê a possibilidade de pedido de destaque e respectivo encaminhamento para o julgamento presencial, aproxima as duas modalidades de julgamento.

2.1.1 Conversão em texto

Obtidos os documentos em formato PDF, é necessário extrair a íntegra do texto dos acórdãos para análise. Isso foi feito por meio de um *script* em Python, que obteve todo o texto de cada documento e armazenou a informação resultante em uma planilha em formato CSV, juntamente com os metadados obtidos pelo *crawler*. No caso de acórdãos mais antigos, originários de processos físicos que não estavam digitalizados, realizou-se um processo automatizado de reconhecimento óptico de caracteres (OCR, do inglês *optical character recognition*) por meio da biblioteca *pytesseract*. OCR, na definição de Jurafsky[179], consiste no reconhecimento automatizado de caracteres criados por máquinas ou por humanos. A planilha final indica os processos que foram digitalizados por meio de OCR[180]. 1305 dos 2136 processos analisados, ou 61.09%, precisaram de OCR.

Os textos extraídos dos acórdãos, bem como o texto constante dos campos "ementa" e "decisão", foram então limpos, isto é, tiveram alguns artefatos removidos, e em sequência foram normalizados em caixa baixa. Extraído o texto dos acórdãos, utilizou-se de várias expressões regulares para obter informações adicionais sobre os processos. Especificamente, foi possível identificar outras ADIs citadas no acórdão, ADIs contra leis e atos estaduais e ainda ADIs julgadas em plenário virtual. Finalmente, foi possível descobrir quais decisões foram unânimes.

2.2 Análise qualitativa

A fim de investigar empiricamente os padrões envolvidos nos julgamentos em lista, a análise computadorizada foi combinada com o levantamento de dados sobre as sessões de julgamentos por meio do registro audiovisual do Tribunal, com o objetivo de

[179] JURAFSKY; MARTIN, **Speech and language processing**.
[180] {O processo de OCR naturalmente introduz alguns artefatos ao converter as páginas xerocadas de cada acórdão em texto. Isso introduz alguns ruídos na análise de semelhança entre documentos, mas não realmente afeta os resultados, devido ao tipo de modelo empregado.

identificar o tempo dedicado em plenário para o debate entre ministros e a tomada de decisões. A anotação cuidadosa da duração das sessões permitiu que se observasse um descompasso entre o elemento oral dos julgamentos e seu correspondente registro escrito, bem como comparar as diferenças em abordagens presentes em um julgamento em lista e um comum. A ADI 4562 é um exemplo particularmente ilustrativo desse descompasso. A Ação Direta possui um acórdão cuja íntegra tem 26 páginas, sendo 9 dedicadas ao voto do relator. A ação, por sua vez, foi julgada em 17 segundos em 17 de outubro de 2018[181], conjuntamente com 4 outras ADIs.

Para o presente trabalho, foram analisadas 77 ADIs julgadas em lista em 2018. Exclui-se da amostra ADIs julgadas em bloco, substituindo-se o bloco por uma ação representativa. Isto é, se em um mesmo momento foram julgadas 5 ações conjuntamente, elas são registradas com o número de uma única ação. Essa decisão busca evitar a contaminação dos resultados por casos repetidos, visto que se 3 casos são julgados em 20 minutos, seria um erro considerar que o plenário gastou uma hora em seu julgamento. Foram identificados julgamentos em lista de ações de controle concentrado em 15 das 76 das sessões realizadas durante o ano.

Nas sessões em que houve julgamentos em sistema de lista, foram coletados dados sobre a duração dos seguintes elementos:

- Apresentação do processo pelo presidente da sessão
- Leitura do relatório
- Sustentações orais
- Leitura do voto do Relator
- Demais intervenções discursivas

Ao todo, observou-se julgamentos extremamente breves. Como se pode observar no *boxplot* constante da figura 10, a sessão mediana durou cerca de 39 segundos, menos de um minuto. Devido à presença de *outliers*, tal como a já referida ADI 5109, cujo julgamento durou 35 minutos e 54 segundos, a duração média das sessões foi substantivamente maior que a mediana, orbitando 146 segundos, ou 2 minutos e 26. Os

[181] Disponível em: https://www.youtube.com/watch?v=_w5h2FIuzq4, de 01:03:56 a 01:04:13.

julgamentos mais rápidos duraram 17 segundos, em um empate técnico das ADIs 4759, 1374, 3894, 4562, 5535 e 5723.

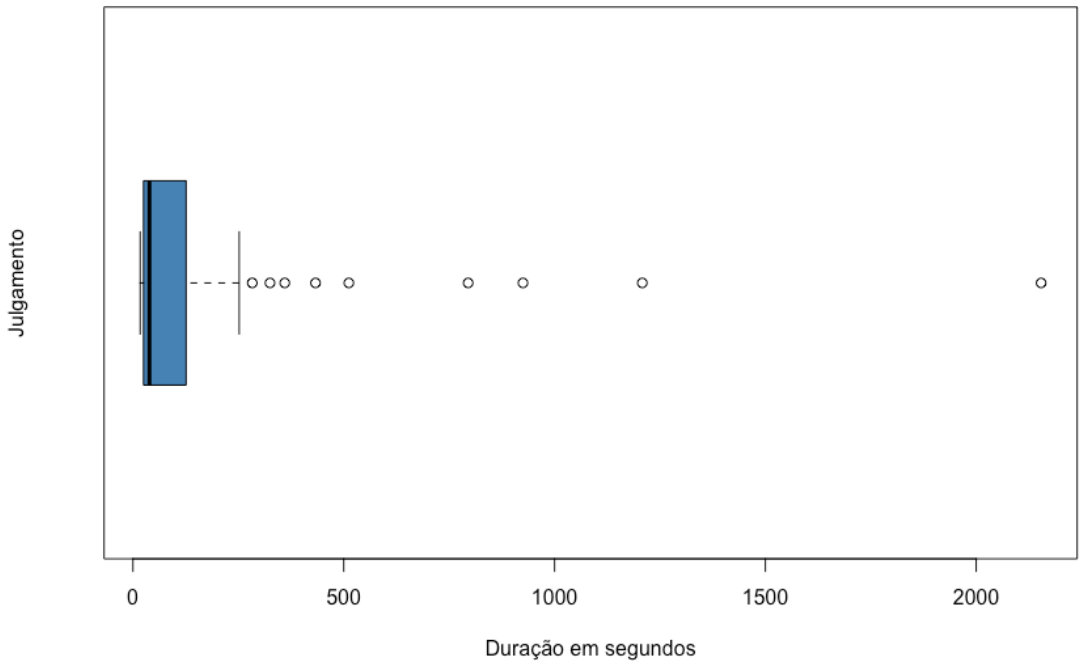


Figura 10: *Boxplot* da duração, em segundos, do julgamento de cada ADI. Cada ponto representa uma ADI. Observa-se uma concentração de processos próxima de 0, dado a curta duração dos julgamentos. A mediana, representada pelo risco preto, é de 39 segundos. Os últimos dois pontos são os *outliers*, correspondentes às ADIs 4332 e 5109, respectivamente.

A figura 11 refere-se à duração do voto do relator. Na maior parte dos julgamentos, 42 dos 77 (54.54%), não houve a leitura de voto. Nesses casos, o Ministro presidindo a sessão limita-se a afirmar se o caso foi julgado procedente ou não, e em seguida pergunta se há alguma divergência. Não havendo, o caso é decidido à unanimidade. Dado que a maioria dos casos é julgada sem a leitura de voto, o voto mediano dura 0 segundos (isto é, não há leitura).

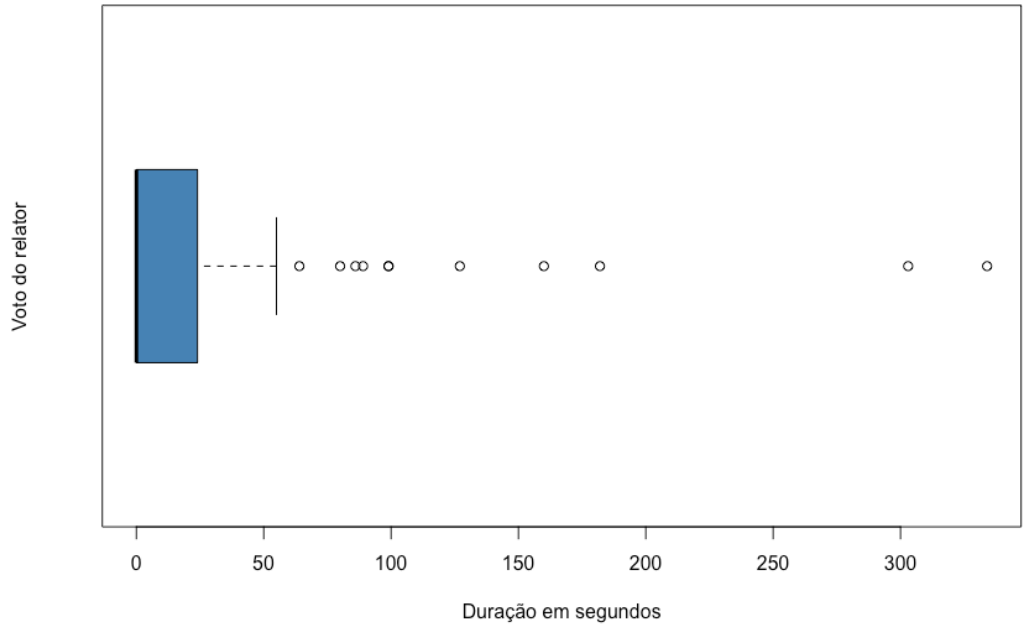


Figura 11*: Boxplot* da duração, em segundos, da leitura do voto do relator no julgamento de cada ADI. Cada ponto representa uma ADI. Observa-se uma concentração de processos próxima de 0, dado a curta duração dos julgamentos. A mediana, representada pelo risco preto, é de 0 segundos. Os últimos dois pontos são os *outliers*, correspondentes às ADIs 4332 e 5109, respectivamente.

A curta duração dos julgamentos, contudo, não significa que não exista debate. Como mencionado anteriormente, em vários casos observados o julgamento em lista se aproximou de um julgamento tradicional após a fala de algum dos ministros. É somente na ausência de qualquer manifestação que o julgamento em lista assume a sua forma mais abreviada. Naturalmente, é provável que só sejam pautados em lista casos incontroversos, de modo que a falta de divergências é, de certa forma, previsível.

***

Capítulo 3 – Resultados

O presente capítulo apresenta os resultados das análises quantitativas detalhadas no capítulo 1. Inicialmente, é introduzido o método do *clustering* hierárquico aglomerante, que é utilizado para não só para categorizar as ADIs coletadas, mas também para eliminar da amostra casos idênticos julgados simultaneamente. Em seguida, é discutida a taxa de singularidade, que se propõe a medir o quão único é um documento quando comparado a seus pares.

Ainda, é apresentado o resultado da aplicação de um modelo LDA de tópicos ao *corpus* de ADIs. Essa técnica é usada para que se possa traçar um panorama de temas tocados nas ADIs, distinguindo diferentes tópicos do *corpus*. Por ser um modelo de membrasia mista, o LDA permite que um mesmo documento pertença à múltiplos tópicos.

Finalmente, é discutido o resultado da aplicação de um classificador bayesiano ingênuo, que faz uso de uma técnica *hold-out* para tentar prever se um processo será julgado em lista ou não somente com base em seu texto.

3.1 Agrupamento Hierárquico Aglomerante

Há diferentes formas de se classificar e categorizar documentos. No primeiro capítulo, foram abordados alguns algoritmos possíveis para a realização dessa tarefa. Como o objetivo da pesquisa é analisar a semelhança de documentos, optou-se pelo método da semelhança do cosseno, ponderado pela estatística de frequência do termo-inverso da frequência nos documentos (*tf-idf*).

Anteriormente, tratou-se de como um texto extraído de cada acórdão pode representado como um vetor de k dimensões, como ilustrado na figura 2. Cada entrada desse vetor é ponderada pela estatística *tf-idf*, de modo a dar maior peso para palavras menos comuns. O valor do cosseno entre dois vetores será uma medida da semelhança entre dois documentos. Esse processo é ilustrado na figura 5 do primeiro capítulo.

Essa medida de semelhança de cosseno, que é também uma medida de distância, pode ser usada como insumo de um algoritmo de agrupamento, tal que documentos mais próximos sejam classificados em um mesmo grupo (ou *cluster*). Para a análise em tela, fez-se uso de um algoritmo hierárquico aglomerante usando o método de Ward. Esse algoritmo foi escolhido por sua eficiência e por exigir relativamente pouco poder

computacional se comparado a outras técnicas de classificação. Confira-se o seguinte dendrograma:

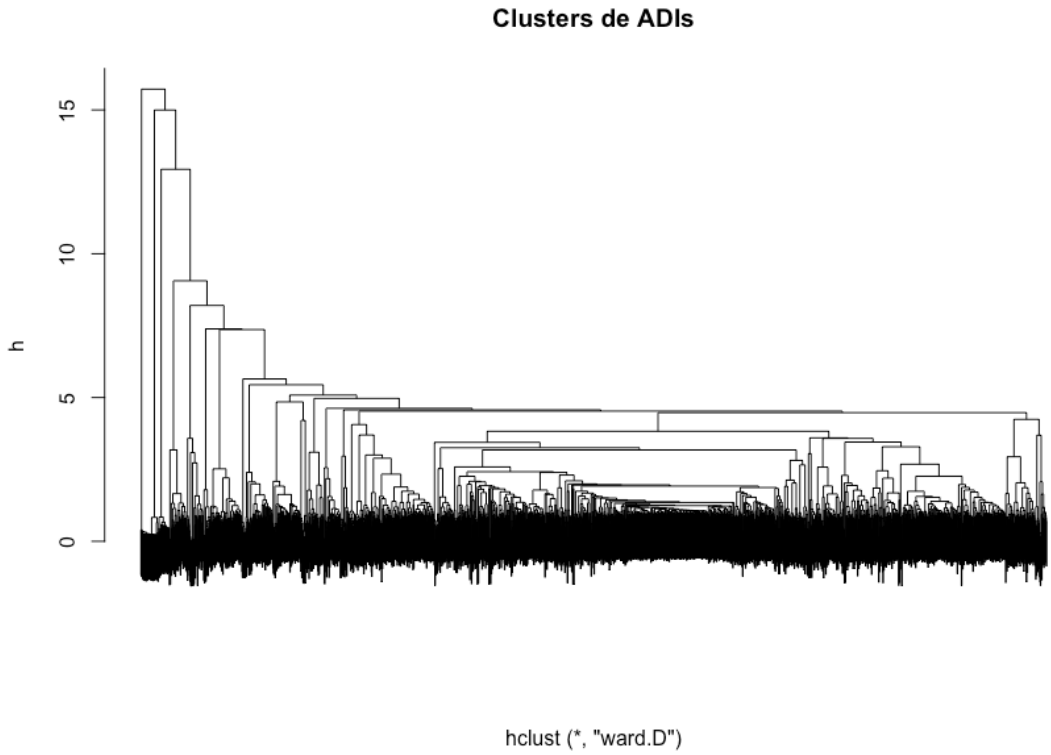


Figura 12: Dendrograma ilustrando a *clusterização* de ADIs. O eixo h representa uma medida associada à semelhança dos documentos quando ponderadas pela estatística *tf-idf*.

Na figura 12, os documentos são divididos em grupos cada vez menores conforme o valor de *h* diminui. O eixo *h* representa uma medida associada à semelhança dos documentos quando ponderadas pela estatística *tf-idf*. Um corte horizontal em dado valor de *h* irá criar um determinado número de *clusters*. Veja-se a figura 13:

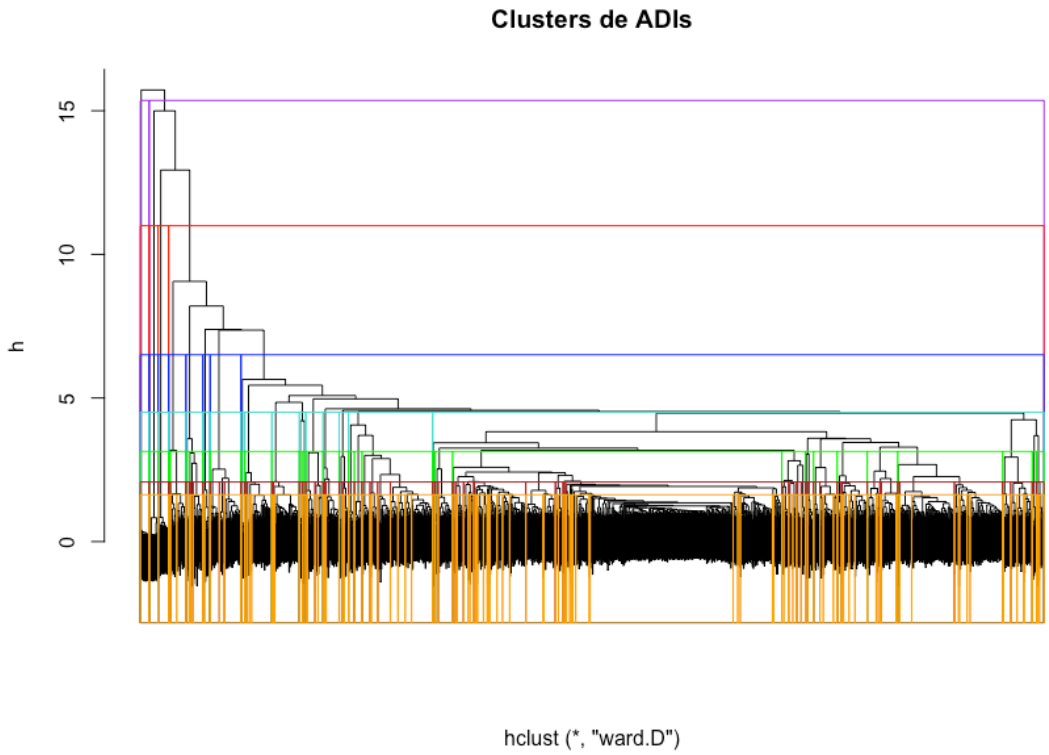


Figura 13: Dendrograma mostrando a divisão de ADIs em *clusters*. Os cortes usados foram para 2 (roxo), 4 (vermelho), 8 (azul), 16 (turquesa), 32 (verde), 64 (marrom) e 128 (laranja) clusters.

A figura 13 ilustra cortes para 2 (roxo), 4 (vermelho), 8 (azul), 16 (turquesa), 32 (verde), 64 (marrom) e 128 (laranja) *clusters*. A decisão do número de *clusters* é flexível, cabendo ao pesquisador optar pela altura do corte. Conforme *h* vai diminuindo, os *clusters* vão sendo subdivididos e mais parecidos serão os documentos. Um valor de *h* próximo de 0 resultará em um número de grupos próximo ou igual ao número de documentos, o que, por óbvio, é de pouca utilidade para o pesquisador, dado que o objetivo é justamente juntar documentos semelhantes. Contudo, os documentos agrupados em valores menores de *h* são mais próximos entre si do que os documentos em clusters maiores

O uso de valores baixos de *h*, formando grupos de alta semelhança entre si, pode ser usado para detectar documentos de conteúdo idêntico ou quase idêntico. No que segue, são criados 1709 *clusters*, uma redução de aproximadamente 20% no total de documentos. Esses *clusters* são combinados com a data de julgamento para identificar processos julgados em bloco. Documentos que estão no mesmo cluster e, simultaneamente, foram julgados no mesmo tratam do mesmo assunto e foram julgados

na mesma lista. Para cada grupo de ADIs julgadas em bloco, é escolhido uma para representar o grupo. O resto é excluído da amostra. Isso evita que um mesmo caso seja representado múltiplas vezes na pesquisa. Esse corte pode, em teoria, pegar casos muito semelhantes que não tenham sido julgados conjuntamente, mas essa hipótese é altamente improvável. Os documentos excluídos da amostra batem com casos julgados em conjunto identificados na análise qualitativa do capítulo 2.

Feita essa exclusão de documentos repetidos, o número de processos cai para 1801. Repete-se então a aplicação do algoritmo, a fim de gerar novo dendrograma:

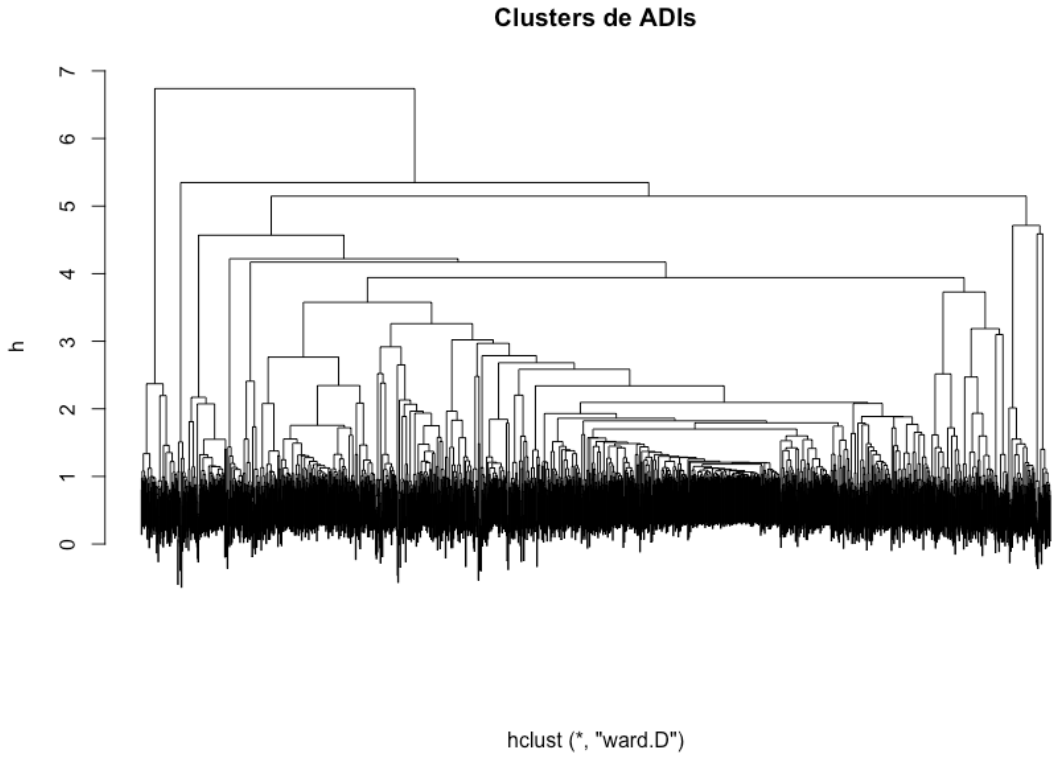


Figura 14: Dendrograma ilustrando a *clusterização* de ADIs após a remoção de processos repetidos. O eixo h representa uma medida associada à semelhança dos documentos quando ponderadas pela estatística *tf-idf*.

Pode-se notar que o dendrograma da figura 14 é distribuído de forma consideravelmente mais homogênea que o anterior. Isso se dá pela exclusão dos casos idênticos, que enviesam o agrupamento. Elaborado o novo dendrograma, é possível novamente dividi-lo em uma sequência de *clusters* (figura 15).

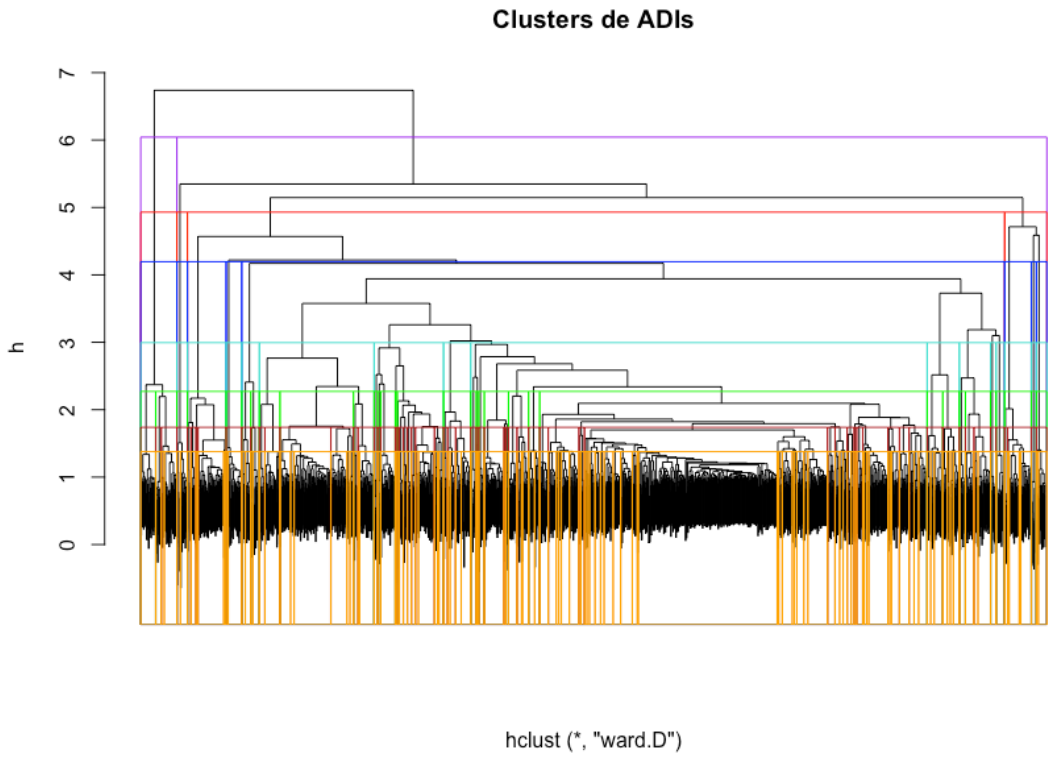


Figura 15: Dendrograma mostrando a divisão de ADIs em *clusters* após a remoção de processos repetidos. Os cortes usados foram para 2 (roxo), 4 (vermelho), 8 (azul), 16 (turquesa), 32 (verde), 64 (marrom) e 128 (laranja) clusters.

Apesar de imperfeitos, os grupos formados por esse processo podem ser utilizados para criar uma medida da singularidade de cada caso. Quanto mais amontada estiver determinada classe de processos, maior sua semelhança interna.

3.1.1. Taxa de singularidade

A intuição por trás da medida de singularidade proposta aqui é relativamente simples. Imagine que toda ADI julgada em lista tratasse de dois assuntos possíveis: 1) questões tributárias relativas ao ICMS ou; 2) questões relativas à previdência de servidores públicos. Nesse cenário, para um valor suficientemente baixo de *h*, naturalmente os casos julgados em lista estariam agrupados em dois, ou próximo de dois, *clusters*. Isso é, palavras-chave associadas às duas únicas questões julgadas em lista, tais

como ‘icms’ e ‘rpps’, tenderiam a aparecer primariamente em casos julgados em lista. Como essas palavras raramente ocorrem, ou ocorrem com menos frequência, em julgados que não tratem de direito tributário ou previdenciário, esses termos possuiriam alto peso ponderado no *tf-idf* e levariam o algoritmo hierárquico aglomerante a *clusterizar* esses casos julgados em lista juntos.

Por outro lado, se processos julgados em sistema de lista nada tiverem em comum, espera-se que não sejam agrupados juntos pelo algoritmo, visto que não terão termos em comum o suficiente. Isso permite que criemos a seguinte medida de singularidade:

$$\text{taxa de singularidade} = \frac{\text{número total de casos}_{tj}}{\text{número de } \textit{clusters} \text{ únicos}_{tj}}$$

Em que *tj* representa o tipo de julgamento. Para um determinado grau de clusterização, a taxa de singularidade medirá o grau de semelhança interno dos casos, agrupados por um de três modos de julgamento: lista, plenário virtual e tradicional. No que segue, calcula-se a taxa de singularidade para cada um dos três modos de julgamento em dois níveis de *clusterização*: 128 e 1621[182] *clusters*.

Tabela 1: Taxa de singularidade para cada modo de julgamento.

Tipo	Número total de processos	Número de clusters distintos [128]	Número de clusters distintos [1621]	Taxa de singularidade [128]	Taxa de singularidade [1621]
Lista	177	69	167	2.565217	1.05988
Tradicional	244	85	223	2.870588	1.09417
Virtual	137	70	134	1.957143	1.022388

[182] Uma redução de 10% em cima dos 1801 documentos.

A tabela 1 mostra os resultados obtidos. Os processos julgados em lista são ligeiramente mais singulares, isto é, menos semelhantes entre si, do que julgados tradicionais. Como julgamentos em lista eram, originalmente, usados para questões que repetidas vezes chegavam ao Supremo, o resultado é surpreendente.

Há diversas explicações possíveis para o fenômeno. É possível que a semelhança de termos não consiga capturar a questão jurídica que está sendo enfrentada, mas semelhanças superficiais, tal como a identidade Requerente. Infelizmente, nesta pesquisa não foi encontrada nenhuma resposta definitiva, de modo que qualquer hipótese aventada é, em algum grau, especulativa.

3.2 LDA

Em seguida, faz-se uso de uma modelagem de tópicos para traçar um panorama de possíveis temas enfrentados no *corpus* de documentos. O modelo LDA é, frise-se, estocástico, de modo que alterações no valor inicial (*seed*) do modelo irão afetar o resultado. O objetivo, assim, não é obter os *verdadeiros* tópicos de um conjunto de documentos, mas sim um conjunto temas e palavras que tendem aparecer junto no grupo. Isso é, trata-se de um método essencialmente exploratório.

A primeira decisão que cabe ao pesquisador é definir o número de tópicos a ser gerado. Essa decisão costuma ser balizada por diferentes estatísticas de qualidade de tópico. As figuras 16 (julgados tradicionais) e 17 (julgados em lista) trazem o resultado das estatísticas de Arun *et al*[183], Cao *et al*[184], Deveaud, Sanjuan e Bellot[185] e Griffith e Steyvers[186] para um conjunto de 9, 16, 25, 36 e 49 tópicos.

[183] ARUN, R. *et al*, On Finding the Natural Number of Topics with Latent Dirichlet Allocation: Some Observations, *in*: ZAKI, Mohammed J. *et al* (Orgs.), **Advances in Knowledge Discovery and Data Mining**, Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, v. 6118, p. 391–402.
[184] CAO, Juan *et al*, A density-based method for adaptive LDA model selection, **Neurocomputing**, v. 72, n. 7–9, p. 1775–1781, 2009.
[185] DEVEAUD, Romain; SANJUAN, Eric; BELLOT, Patrice, Accurate and effective latent concept modeling for ad hoc information retrieval, **Document numérique**, v. 17, n. 1, p. 61–84, 2014.
[186] GRIFFITHS, Thomas L.; STEYVERS, Mark, Finding scientific topics, **Proceedings of the National Academy of Sciences**, v. 101, n. suppl_1, p. 5228–5235, 2004.

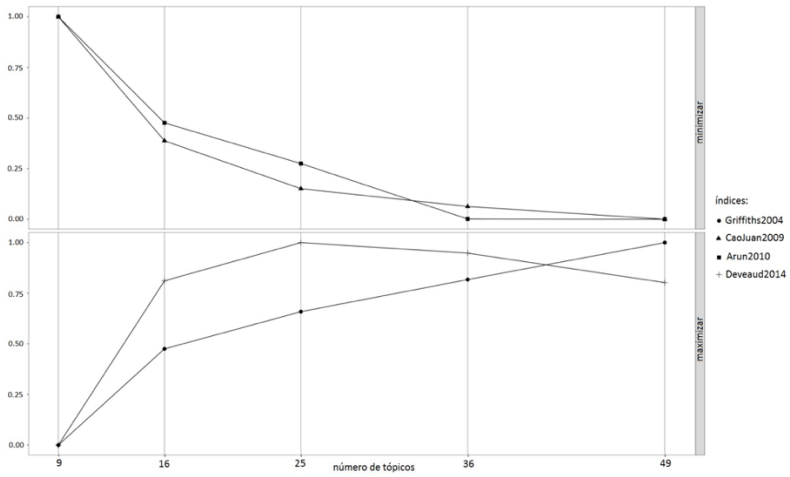


Figura 16: Número ótimo de tópicos, de acordo com diferentes medidas de qualidade de tópico, para ADIs julgadas tradicionalmente.

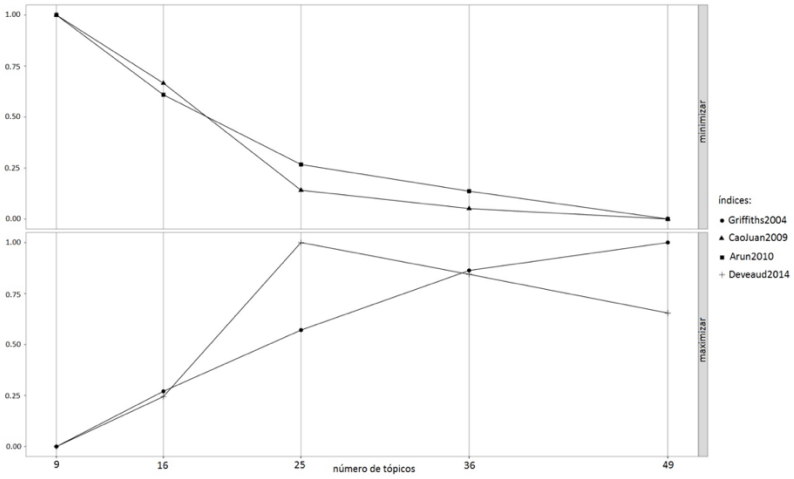


Figura 17: Número ótimo de tópicos, de acordo com diferentes medidas de qualidade de tópico, para ADIs julgadas em lista.

Como se pode se observar nas figuras, os melhores resultados foram encontrados para cerca de 25 tópicos. O aumento do número de tópicos acarreta perdas no índice de Deveaud e ganhos pequenos nos demais índices.

Figura 18: Resultado da modelagem com 25 tópicos com a técnica LDA das ADIs julgadas tradicionalmente. Na figura, temos as 15 palavras mais frequentes por tópico.

Figura 19: Resultado da modelagem com 25 tópicos com a técnica LDA das ADIs julgadas em lista. Na figura, temos as 15 palavras mais frequentes por tópico.

As figuras 18 e 19 mostram o resultado do modelo LDA para 25 tópicos para casos julgados tradicionalmente e em lista, respectivamente. Observa-se que não há predominância de um único ramo do direito, como seria de se esperar, dada a amplitude da amostra. Previsivelmente, a maior parte dos tópicos versa sobre questões de direito público. Da mesma forma, não é claro que há diferença significativa nos tópicos dos casos julgados tradicionalmente.

3.3 Taxa de unanimidade

Um resultado interessante da pesquisa é a alta taxa de unanimidade dos casos julgados em lista, cerca de 90%, contra 63% dos casos julgados tradicionalmente em rito presencial. Veja-se a figura 20:

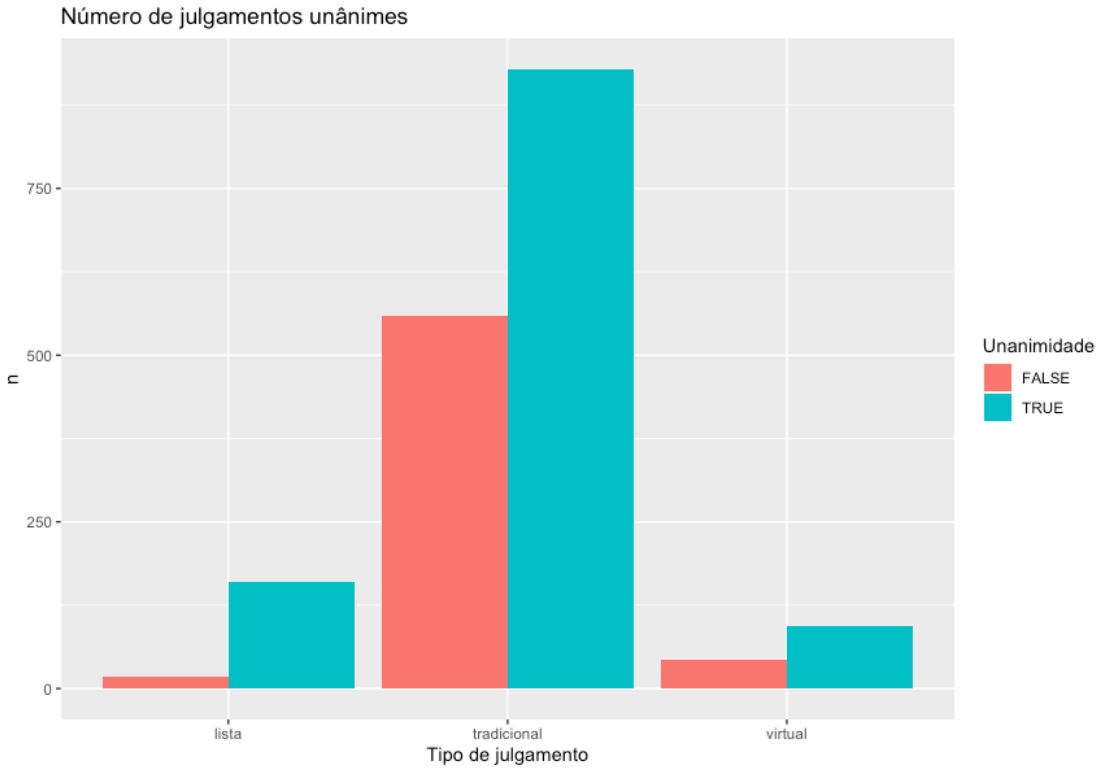


Figura 20: Taxa de unanimidade de ADIs, por modo de julgamento. Em azul, temos as ADIs decididas à unanimidade, em vermelho as decididas por maioria.

A considerável diferença na taxa de unanimidade sugere uma correlação entre a forma de julgamento e o resultado. Felizmente, é possível verificar se há de fato essa

associação por meio de um teste de hipótese denominado teste qui-quadrado de Pearson. Ele objetiva verificar se duas variáveis são associadas ou independentes.

O teste faz uso da estatística de teste de Pearson, cujo valor pode ser comparado com uma distribuição qui-quadrado, gerando assim um p-valor associado que pode ser usado para rejeitar ou não a hipótese de independência. A figura 21 mostra o resultado do teste:

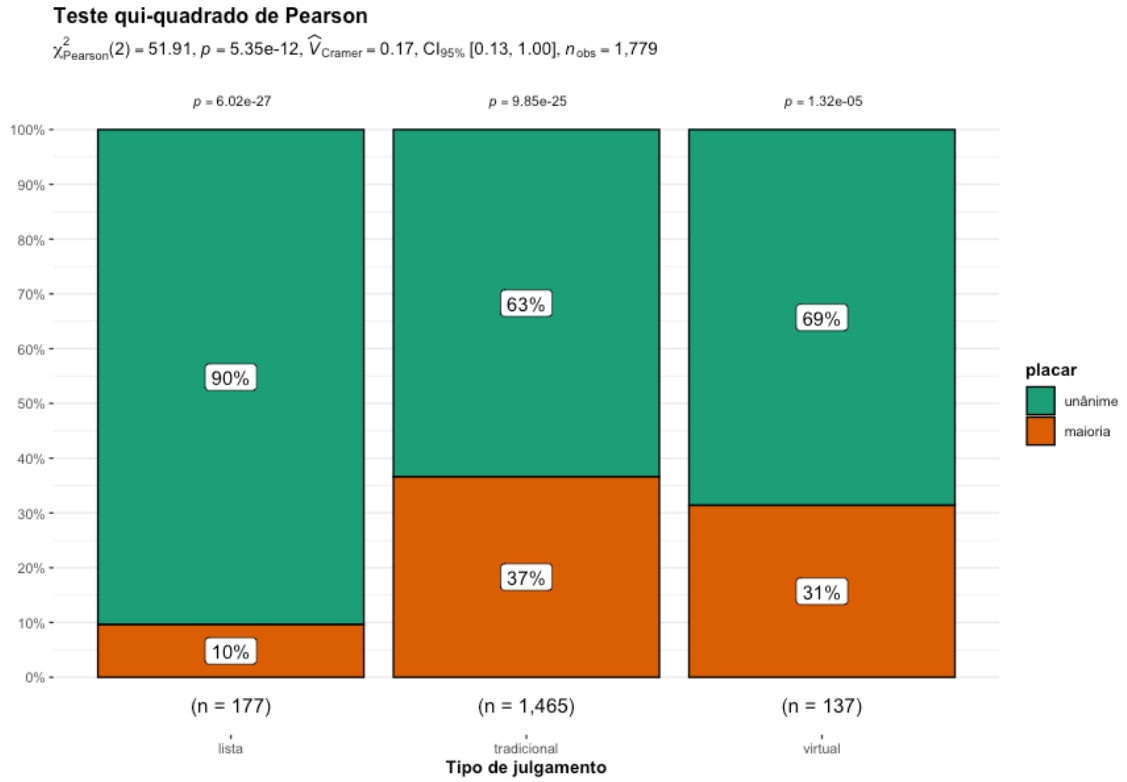


Figura 21: Resultado do teste qui-quadrado de Pearson entre as variáveis “tipo de julgamento” (lista, tradicional e virtual) e “placar” (unânime ou maioria). Como o p-valor é bem menor que a taxa de significância, podemos rejeitar a hipótese de independência entre as duas variáveis.

O valor do p-valor é consideravelmente menor que o limite padrão de significância estatística de 0.05, o que sugere que os resultados são significativos e podemos rejeitar a hipótese nula de que as variáveis são independentes. O resultado é intuitivo, visto que casos de menor complexidade tendem a causar menor controvérsia. Ademais, a sistemática do julgamento em lista, em que a unanimidade é suposta na ausência de divergência, incentiva a formação de julgamentos unânimes.

3.3 Classificador bayesiano ingênuo

A aplicação de um classificador bayesiano pode ser um mecanismo interessante para sugerir se há elementos textuais que preveem se um caso será julgado em lista. Ao se avaliar a performance de um classificador, é comum o uso do método *hold-out*, em que parte da amostra é excluída do treinamento. Assim, o algoritmo é treinado com parte dos casos, associando seu conteúdo textual ao rótulo desejado (no caso, o tipo de julgamento) e então tem sua performance avaliada tentando classificar um conjunto de documentos com rótulos ocultos, mas conhecidos ao pesquisador. Se o classificador obtém boa performance classificando esse conjunto de documentos, isso é evidência de que há elementos no texto que podem ajudar a prever como um caso será julgado (ie. Qual será a classificação do caso).

No presente caso, treinou-se um classificador bayesiano ingênuo com 80% da amostra total de casos posteriores a 2015. Excluiu-se processos anteriores ao advento do julgamento em lista porque, do contrário, o classificador se valeria das datas contidas no corpo do processo para realizar a tarefa de classificação. Isso seria contraproducente e não apresentaria evidência de elementos textuais substantivos no texto que levam um caso a ser julgado de uma forma ou de outra.

A performance do classificador foi então avaliada tentando prever a classificação dos 20% processos sobressalentes. As tabelas 2 e 3 mostram os resultados:

Tabela 2: Matriz de confusão do classificador bayesiano ingênuo.

	Previsão (Lista)	Previsão (Tradicional)	Total
Real (Lista)	35	1	36
Real (Tradicional)	15	12	27
Total	50	13	63

Tabela 3: Estatísticas de performance do classificador. Observa-se que a acurácia é menor que a taxa de prevalência, o que indica uma performance pior do que o classificador teria se sempre chutasse “lista”.

Sensibilidade	Prevalência	Precisão	Especificidade	F1	Kappa	Acurácia
0.7	0.794	0.972	0.923	0.814	0.45	0.75

Apesar da taxa de acurácia relativamente alta (75%), a taxa de prevalência sugere que a performance do classificador é pior que a sorte. Isto é, se o classificador unicamente chutasse ‘lista’, a categoria prevalente na amostra, ele acertaria 79.4% das vezes. O conteúdo do texto, assim, em nada contribuiu para a previsão acertada da forma de julgamento.

***

4- Conclusão

O presente trabalho analisou a figura do julgamento em lista de Ações Diretas de Inconstitucionalidade por meio de uma metodologia primariamente quantitativa, com elementos qualitativos. Para tanto, se valeu do uso de técnicas de coleta e raspagem de dados em Python para construir uma base de dados original com todos os acórdãos de ADIs publicados pela Corte até o final de 2019. Os dados coletados foram então utilizados para elaborar modelos computacionais que foram usados conjuntamente com dados qualitativos.

No primeiro capítulo, traçou-se um panorama metodológico de conceitos fundamentais do processamento de linguagem natural e da análise automatizada de conteúdo nas ciências sociais. No capítulo seguinte, descreveu-se a base de dados construída e a análise qualitativa feita com as sessões televisionadas do Supremo. Finalmente, no terceiro capítulo, apresentou-se os resultados obtidos pelos modelos aplicados.

A principal contribuição feita por essa pesquisa é descritiva, medindo e quantificando o uso da técnica do julgamento em lista pelo Supremo Tribunal Federal. Até a elaboração deste trabalho, não havia estimativa da proporção de ADIs julgadas em lista. Os resultados encontrados mostram que o Tribunal fez amplo e crescente uso desse modo de julgamento até o ano de 2019, quando passou a julgar ADIs também em plenário virtual. Espera-se que trabalhos futuros façam uma análise qualitativa minuciosa do conteúdo das ADIs identificadas como julgadas em lista.

A queda brusca ocorrida nesse ano sugere uma possível substituição do julgamento em lista pelo plenário virtual, que permite ainda maior celeridade ao dispensar a presença física na sessão. A aparente substituição do julgamento em lista pelo plenário virtual, por sua vez, pode ser compreendida como um mecanismo de gestão do tempo da sessão do plenário. Outros trabalhos, como Gomes, já documentaram comportamentos estratégicos da Corte objetivando economizar tempo nas sessões[187].

A análise da dimensão temporal das sessões, por sua vez. revelou uma duração que não é compatível com a deliberação complexa própria de debates constitucionais. Só foi possível se ter dimensão da celeridade dessa modalidade de julgamento após o

[187] GOMES, Kelton de Oliveira, **A monocratização das liminares em controle concentrado de constitucionalidade no âmbito do Supremo Tribunal Federal (1988-2018)**, Master, Universidade de Brasília, 2019.

trabalho qualitativo quase artesanal de cronometragem das sessões. Com uma duração mediana de 39 segundos, a estrutura das sessões de julgamento em lista não comporta debates entre ministros. O debate, quando ocorre, é excepcional. Na ausência de manifestação contrária, a maior parte dos casos foi decidida sem nem mesmo leitura de voto. Nesse sentido, o plenário virtual, apesar de possuir fragilidades, pode representar um avanço, sendo, ao menos, mais transparente.

Neste trabalho, a repetitividade do conteúdo das ADIs foi medida por meio da taxa de singularidade associada a cada modo de julgamento. Intuitivamente, esperava-se que ADIs julgadas em lista, se versassem sempre sobre os mesmos conteúdos e citassem os mesmos precedentes, seriam mais semelhantes entre si e teriam um índice de singularidade maior, mais distante de 1, do que ADIs julgadas tradicionalmente. De forma contraintuitiva, o resultado encontrado sugere que as ADIs julgadas tradicionalmente a partir de 2015 têm mais em comum entre si do que ações julgadas em lista. Assim, a hipótese de que ADIs julgadas em lista seriam mais semelhantes entre si não foi confirmada.

Finalmente, o resultado dos modelos automatizados de análise de conteúdo não indicou a presença de elementos textuais para definir quais casos seriam julgados em lista. A alta singularidade dos casos julgados em sistema e lista sugere um uso generalizado do instituto. Em nossa amostra, cerca de 74% das ADIs publicadas julgadas em 2018 foram julgadas em lista, contra 30% em 2015.

Em termos substantivos, a maior conclusão deste trabalho é que julgamentos em lista são largamente unânimes. Isso não é particularmente surpreendente, é intuitivo que julgamentos abreviados sejam largamente consensuais, mas nem tudo que é intuitivo é confirmado empiricamente. Nesse sentido, a presente dissertação contribui com o estudo do comportamento do Supremo Tribunal Federal.

Devido à pandemia do COVID-19, o Supremo Tribunal Federal passou a fazer uso majoritário de sessões remotas. Assim, a figura do julgamento em lista perdeu importância frente ao julgamento em plenário virtual. Dessa forma, a contribuição do trabalho é, em parte, mais histórica que substantiva, versando sobre um período único da história do Supremo Tribunal Federal em que o julgamento em lista serviu, antes da previsão regimental, como um híbrido entre o julgamento presencial e o plenário virtual, trazendo a celeridade do julgamento remoto ao ambiente deliberativo tradicional de um plenário.

***

5- Bibliografia

AGGARWAL, Charu C.; ZHAI, ChengXiang (Orgs.). **Mining Text Data**. Boston, MA: Springer US, 2012. Disponível em: <http://link.springer.com/10.1007/978-1-4614-3223-4>. Acesso em: 19 set. 2021.

ARUN, R.; SURESH, V.; VENI MADHAVAN, C. E.; *et al*. On Finding the Natural Number of Topics with Latent Dirichlet Allocation: Some Observations. *In*: ZAKI, Mohammed J.; YU, Jeffrey Xu; RAVINDRAN, B.; *et al* (Orgs.). **Advances in Knowledge Discovery and Data Mining**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, v. 6118, p. 391–402. (Lecture Notes in Computer Science). Disponível em: <http://link.springer.com/10.1007/978-3-642-13657-3_43>. Acesso em: 18 ago. 2022.

ASH, Elliott; CHEN, Daniel L.; GALLETTA, Sergio. Measuring Judicial Sentiment: Methods and Application to US Circuit Courts. **Economica**, v. 89, n. 354, p. 362–376, 2022.

ASKITAS, Nikolaos; ZIMMERMANN, Klaus F. Google Econometrics and Unemployment Forecasting. **Applied Economics Quarterly**, v. 55, n. 2, p. 107–120, 2009.

BARROSO, Luís Roberto. **O controle de constitucionalidade no direito brasileiro: exposição sistemática da doutrina e análise crítica da jurisprudência**. 7a edição revista e atualizada. São Paulo, SP: Editora Saraiva, 2016.

BARROSO, Luís Roberto; MONTEDONIO REGO, Frederico. Como Salvar o Sistema de Repercussão Geral: Transparência, Eficiência e Realismo na Escolha do que o Supremo Tribunal Federal Vai Julgar. **Revista Brasileira de Políticas Públicas**, v. 7, n. 3, 2018. Disponível em: <https://www.publicacoes.uniceub.br/RBPP/article/view/4824>. Acesso em: 27 mar. 2021.

BENDER, Emily M.; GEBRU, Timnit; MCMILLAN-MAJOR, Angelina; *et al*. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? *In*: **Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency**. Virtual Event Canada: ACM, 2021, p. 610–623. Disponível em: <https://dl.acm.org/doi/10.1145/3442188.3445922>. Acesso em: 20 set. 2021.

BENGFORT, Benjamin; BILBRO, Rebecca; OJEDA, Tony. Applied Text Analysis with Python. p. 332, 2018.

BENOIT, Ken. Text as Data: An Overview. *In*: CURINI, Luigi; FRANZESE, Robert (Eds.). **The SAGE Handbook of Research Methods in Political Science and International Relations**. 1 Oliver's Yard, 55 City Road London EC1Y 1SP: SAGE Publications Ltd, 2020, p. 461–497. Disponível em: <https://sk.sagepub.com/reference/the-sage-handbook-of-research-methods-in-political-science-and-ir/i4365.xml>. Acesso em: 19 set. 2021.

BENVINDO, Juliano Zaiden; COSTA, Alexandre Araújo. A Quem Interessa o Controle Concentrado De Constitucionalidade? - O Descompasso entre Teoria e Prática na Defesa dos Direitos Fundamentais. p. 84, 2014.

BIRD, Steven. Natural Language Processing with Python. p. 504, 2009.

BLACK, Ryan C.; TREUL, Sarah A.; JOHNSON, Timothy R.; *et al*. Emotions, Oral Arguments, and Supreme Court Decision Making. **The Journal of Politics**, v. 73, n. 2, p. 572–581, 2011.

BLEI, David M. Probabilistic topic models. **Communications of the ACM**, v. 55, n. 4, p. 77–84, 2012.

BOX, George E. P. Science and Statistics. **Journal of the American Statistical Association**, v. 71, n. 356, p. 791–799, 1976.

CAO, Juan; XIA, Tian; LI, Jintao; *et al*. A density-based method for adaptive LDA model selection. **Neurocomputing**, v. 72, n. 7–9, p. 1775–1781, 2009.

CASTRO, Marcus Faro de. O Supremo Tribunal Federal e a Judicialização da Política. p. 19, 1997.

COELHO, Gustavo Rodrigues. **Utilizando Text Mining na Taxonomia Processual**. 2018.

COSTA, Alexandre Araújo; CARVALHO, Alexandre Douglas Zaidan de; FARIAS, Felipe Justino de. Controle de constitucionalidade no Brasil: eficácia das políticas de concentração e seletividade. **Revista Direito GV**, v. 12, n. 1, p. 155–187, 2016.

COSTA, Alexandre Araújo; COSTA, Henrique Araújo. Evolução do perfil dos demandantes no controle concentrado de constitucionalidade realizado pelo STF por meio de ADIs e ADPFs. v. 49, n. 2, p. 47, 2018.

DE AZEVEDO SOARES, Fabio. **Mineração de Textos na Coleta Inteligente de Dados na Web**. MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA, PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO, Rio de Janeiro, Brazil, 2008. Disponível em: <http://www.maxwell.vrac.puc-rio.br/Busca_etds.php?strSecao=resultado&nrSeq=13212@1>. Acesso em: 13 set. 2021.

DESPOSATO, Scott W.; INGRAM, Matthew C.; LANNES, Osmar P. Power, Composition, and Decision Making: The Behavioral Consequences of Institutional Reform on Brazil's *Supremo Tribunal Federal*. **Journal of Law, Economics, and Organization**, v. 31, n. 3, p. 534–567, 2015.

DEVEAUD, Romain; SANJUAN, Eric; BELLOT, Patrice. Accurate and effective latent concept modeling for ad hoc information retrieval. **Document numérique**, v. 17, n. 1, p. 61–84, 2014.

DWORKIN, Ronald. **A matter of principle**. 9th print. Cambridge, Mass: Harvard Univ. Press, 2000.

EVANS, Michael; MCINTOSH, Wayne; LIN, Jimmy; *et al*. Recounting the Courts? Applying Automated Content Analysis to Enhance Empirical Legal Research: Automated Content Analysis to Enhance Empirical Legal Research. **Journal of Empirical Legal Studies**, v. 4, n. 4, p. 1007–1039, 2007.

FALCÃO, Joaquim; CERDEIRA, Pablo; ARGUELHES, Diego. I Relatório do Supremo em Números - O múltiplo Supremo. **Revista de Direito Administrativo**, v. 262, p. 399, 2013.

FEDERAL RESERVE BANK OF SAN FRANCISCO; SHAPIRO, Adam H.; SUDHOF, Moritz; *et al*. Measuring News Sentiment. **Federal Reserve Bank of San Francisco, Working Paper Series**, p. 01–49, 2020.

GENTZKOW, Matthew; KELLY, Bryan; TADDY, Matt. Text as Data. **Journal of Economic Literature**, v. 57, n. 3, p. 535–574, 2019.

GOMES, Kelton de Oliveira. **A monocratização das liminares em controle concentrado de constitucionalidade no âmbito do Supremo Tribunal Federal (1988-2018)**. Master, Universidade de Brasília, 2019.

GOMES NETO, Jose Mario Wanderley; FEITOSA, Raymundo Juliano Do Rego; DOS SANTOS FILHO, Moacir Ferreira; *et al*. Litígios Esquecidos: Análise empírica dos processos de controle concentrado de constitucionalidade aguardando julgamento. **Revista de Estudos Empíricos em Direito**, v. 4, n. 2, 2017. Disponível em: <https://revistareed.emnuvens.com.br/reed/article/view/146>. Acesso em: 26 mar. 2021.

GRIFFITHS, Thomas L.; STEYVERS, Mark. Finding scientific topics. **Proceedings of the National Academy of Sciences**, v. 101, n. suppl_1, p. 5228–5235, 2004.

GRIMMER, Justin; STEWART, Brandon M. Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts. **Political Analysis**, v. 21, n. 3, p. 267–297, 2013.

HARTMANN, Ivar Alberto Martins; FERREIRA, Lívia Da Silva. Ao relator, tudo: o impacto do aumento do poder do ministro relator no Supremo. **Revista Opinião Jurídica (Fortaleza)**, v. 13, n. 17, p. 268, 2016.

HAUSLADEN, Carina I.; SCHUBERT, Marcel H.; ASH, Elliott. Text classification of ideological direction in judicial opinions. **International Review of Law and Economics**, v. 62, p. 105903, 2020.

HVITFELDT, Emil; SILGE, Julia. **Supervised machine learning for text analysis in R**. First edition. Boca Raton: CRC Press, 2022. (Data science series).

JALORETTO, M.F.; MUELLER, B.P.M. O Procedimento de Escolha dos Ministros do Supremo Tribunal Federal – Uma Análise Empírica. **Economic Analysis of Law Review**, v. 2, n. 1, p. 170–187, 2011.

JO, Taeho. **Text Mining**. Cham: Springer International Publishing, 2019. (Studies in Big Data). Disponível em: <http://link.springer.com/10.1007/978-3-319-91815-0>. Acesso em: 19 set. 2021.

JURAFSKY, Dan; MARTIN, James H. **Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition**. Upper Saddle River, N.J: Prentice Hall, 2000. (Prentice Hall series in artificial intelligence).

KAPISZEWSKI, Diana. Power Broker, Policy Maker, or Rights Protector? *In*: HELMKE, Gretchen; RIOS-FIGUEROA, Julio (Orgs.). **Courts in Latin America**. Cambridge: Cambridge University Press, 2011, p. 154–186. Disponível em: <https://www.cambridge.org/core/product/identifier/CBO9780511976520A014/type/book_part>. Acesso em: 26 mar. 2021.

KAUFMAN, Aaron Russell; KRAFT, Peter; SEN, Maya. Improving Supreme Court Forecasting Using Boosted Decision Trees. **Political Analysis**, v. 27, n. 3, p. 381–387, 2019.

LAVER, Michael; BENOIT, Kenneth; GARRY, John. Extracting Policy Positions from Political Texts Using Words as Data. **American Political Science Review**, v. 97, n. 02,

2003. Disponível em: <http://www.journals.cambridge.org/abstract_S0003055403000698>. Acesso em: 14 set. 2021.

LIEBMAN, Benjamin L.; ROBERTS, Margaret; STERN, Rachel E.; *et al*. Mass Digitization of Chinese Court Decisions: How to Use Text as Data in the Field of Chinese Law. **SSRN Electronic Journal**, 2017. Disponível em: <http://www.ssrn.com/abstract=2985861>. Acesso em: 25 dez. 2020.

LIVERMORE, Michael A.; ROCKMORE, Daniel N. (Orgs.). **Law as data: computation, text, & the future of legal analysis**. Santa Fe: SFI Press, 2019. (Seminar, book 3).

MACHADO, Maíra Rocha. **Pesquisar empiricamente o direito**. [s.l.]: Rede de Estudos Empíricos em Direito, 2017.

MANNING, Christopher; RAGHAVAN, Prabhakar; SCHUETZE, Hinrich. Introduction to Information Retrieval. p. 581, 2009.

MARTIN, Andrew D.; QUINN, Kevin M. Dynamic Ideal Point Estimation via Markov Chain Monte Carlo for the U.S. Supreme Court, 1953–1999. **Political Analysis**, v. 10, n. 2, p. 134–153, 2002.

MITCHELL, Ryan. Web Scraping with Python. p. 306, 2018.

MOREIRA, Davi; IZUMI, Maurício. O texto como dado: desafios e oportunidades para as ciências sociais. **Revista Brasileira de Informação Bibliográfica em Ciências Sociais - BIB**, v. 86, p. 138–174, 2018.

NERY FERREIRA, Pedro Fernando Almeida; MUELLER, Bernardo. How judges think in the Brazilian Supreme Court: Estimating ideal points and identifying dimensions. **EconomiA**, v. 15, n. 3, p. 275–293, 2014.

O'CONNOR, Brendan; BAMMAN, David; SMITH, Noah A. Computational Text Analysis for Social Science: Model Assumptions and Complexity. p. 8, 2011.

PIANTADOSI, Steven T. Zipf's word frequency law in natural language: A critical review and future directions. **Psychonomic Bulletin & Review**, v. 21, n. 5, p. 1112–1130, 2014.

QUINN, Kevin M.; MONROE, Burt L.; COLARESI, Michael; *et al*. How to Analyze Political Attention with Minimal Assumptions and Costs. **American Journal of Political Science**, v. 54, n. 1, p. 209–228, 2010.

SHAH, Dev; ISAH, Haruna; ZULKERNINE, Farhana. Predicting the Effects of News Sentiments on the Stock Market. *In*: **2018 IEEE International Conference on Big Data (Big Data)**. Seattle, WA, USA: IEEE, 2018, p. 4705–4708. Disponível em: <https://ieeexplore.ieee.org/document/8621884/>. Acesso em: 19 set. 2021.

SILVA, Jeferson Mariano. **Jurisdição constitucional em Espanha (1981-1992) e Brasil (1988-1997)**. 2016.

SLAPIN, Jonathan B.; PROKSCH, Sven-Oliver. A Scaling Model for Estimating Time-Series Party Positions from Texts. **American Journal of Political Science**, v. 52, n. 3, p. 705–722, 2008.

STEMLER, Igor Tadeu Silva Viana. **Identificação de Precedentes Judiciais por Agrupamento Utilizando Processamento de Linguagem Natural**. Mestrado, Universidade de Brasília, 2019.

SULEA, Octavia-Maria; ZAMPIERI, Marcos; MALMASI, Shervin; *et al*. Exploring the Use of Text Classification in the Legal Domain. **arXiv:1710.09306 [cs]**, 2017. Disponível em: <http://arxiv.org/abs/1710.09306>. Acesso em: 22 maio 2021.

TATE, C. Neal; VALLINDER, Torbjörn (Orgs.). **The global expansion of judicial power**. New York: New York University Press, 1995.

TETLOCK, Paul C. Giving Content to Investor Sentiment: The Role of Media in the Stock Market. **The Journal of Finance**, v. 62, n. 3, p. 1139–1168, 2007.

VIANNA, Luiz Werneck (Org.). **A judicialização da política e das relações sociais no Brasil**. Rio de Janeiro: Editora Revan, 1999.

WILKERSON, John; CASAS, Andreu. Large-Scale Computerized Text Analysis in Political Science: Opportunities and Challenges. **Annual Review of Political Science**, v. 20, n. 1, p. 529–544, 2017.

YU, Shuiyuan; XU, Chunshan; LIU, Haitao. Zipf's law in 50 languages: its structural pattern, linguistic interpretation, and cognitive motivation. **CoRR**, v. abs/1807.01855, 2018. Disponível em: <http://arxiv.org/abs/1807.01855>.

***

6- Anexo

Duração das ADIs julgadas em lista em 2018

Número	Início do julgamento	Voto relator inicio	Voto relator fim	Voto relator	Fim do julgamento	Total	Obs
ADI 5585	01:16:00	01:16:18	01:17:13	55s	01:17:23	83s	
ADI 5300	02:02:15	02:02:23	02:04:02	99s	02:06:22	247s	
ADI 5723	01:46:01	01:46:01	01:46:01	0s	01:46:18	17s	
ADI 4544	02:32:15	02:32:50	02:33:14	24s	02:33:20	65s	
ADI 3785	01:06:48	01:06:59	01:07:01	2s	01:07:12	24s	
ADI 3666	01:17:12	01:17:12	01:17:12	0s	01:17:32	20s	
ADI 5024	01:12:08	01:12:08	01:12:08	0s	01:12:29	21s	
ADI 4019	01:18:10	01:18:15	01:19:19	64s	01:19:23	73s	Presidido pelo Fux, porque Toffoli estava impedido.
ADI 5111	01:07:31	01:07:42	01:07:44	2s	01:08:07	36s	Divergência do Marco Aurélio quanto à modulação.
ADI 4962	00:40:15	00:40:40	00:41:12	32s	00:41:16	61s	
ADI 4601	00:02:27	00:02:27	00:02:27	0s	00:02:48	21s	
ADI 5432	00:10:09	00:10:47	00:10:58	11s	00:11:25	76s	Consta que há, mas não foi feita.
ADI 4058	01:50:40	01:51:15	01:51:33	18s	01:51:52	72s	Toffoli impedido, Fux Presidiu.

ADI 5725	01:06:37	01:06:37	01:06:37	0s	01:07:15	38s	Fachin acompanhou pela colegialidade, registrando divergências.
ADI 4647	02:24:30	02:24:45	02:24:50	5s	02:24:54	24s	
ADI 1606	01:43:18	01:43:18	01:43:18	0s	01:45:55	157s	
ADI 5158	01:08:10	01:09:17	01:10:37	80s	01:16:42	512s	Registrou-se que ocorreu a sustentação. Advogado presente. Voto colhido individualmente. Ampla discussão.
ADI 3144	01:17:27	01:17:39	01:19:05	86s	01:20:00	153s	
ADI 3418	01:02:43	01:02:53	01:03:07	14s	01:03:12	29s	
ADI 4421	02:02:38	02:02:38	02:02:38	0s	02:04:58	140s	Toffoli retificou seu voto. Constava como "devolução de vista".
ADI 4332	00:34:03	00:47:17	00:52:51	334s	00:54:11	1208 s	Sustentação oral. Gilmar se manifestou apontando que a demanda é reiterada e que a OAB deveria fazer uma avaliação,

							considerando o interesse público, para que esse tipo de pleito não ocupe a Corte. Acompanhou o relator.
ADI 5004	00:44:27	00:44:48	00:45:14	26s	00:45:17	50s	
ADI 1975	02:05:00	02:05:00	02:05:00	0s	02:05:50	50s	Toffoli impedido, Fux Presidiu.
ADI 4243	01:43:57	01:43:57	01:43:57	0s	01:44:15	18s	
ADI 5213	01:46:13	01:46:33	01:47:21	48s	01:47:36	83s	
ADI 5107	01:59:17	01:59:58	02:01:27	89s	02:02:07	170s	
ADI 5018	01:38:11	01:38:30	01:39:02	32s	01:42:23	252s	
ADI 5535	01:45:43	01:45:43	01:45:43	0s	01:46:00	17s	
ADI 2323	01:27:51	01:27:51	01:27:51	0s	01:28:35	44s	
ADI 1757	01:14:45	01:27:51	01:27:51	0s	01:15:29	44s	
ADI 4133	01:16:18	01:16:27	01:16:40	13s	01:18:06	108s	Presidido pelo Fux, porque Toffoli estava impedido.
ADI 5109	00:00:34	00:01:42	00:04:22	160s	00:36:28	2154 s	Relator continua votando em 0:25:20 até 0:28:52. Amplo debate.
ADI 4807	01:49:24	01:49:34	01:50:23	49s	01:50:28	64s	

ADI 2087	00:14:40	00:15:02	00:18:04	182s	00:19:23	283s	
ADI 3863	01:13:34	01:13:34	01:13:34	0s	01:14:02	28s	Presidido pelo Fux, porque Toffoli estava impedido.
ADI 854	01:04:14	01:04:48	01:04:53	5s	01:05:00	46s	
ADI 5832	01:11:39	01:11:39	01:11:39	0s	01:12:12	33s	Toffoli presidiu, percebeu que estava impedido, registrou que o Min. Fux presidiu.
ADI 4977	01:44:10	01:44:27	01:44:43	16s	01:44:47	37s	
ADI 3995	01:20:48	01:21:32	01:21:37	5s	01:26:48	360s	Presidido pelo Fux, porque Toffoli estava impedido.
ADI 5776	01:51:56	01:51:56	01:51:56	0s	01:54:49	173s	Min. Marco Aurélio se manifestou.
ADI 3500	01:11:39	01:11:39	01:11:39	0s	01:12:12	33s	Barroso impedido.
ADI 5275	00:19:33	00:19:33	00:19:33	0s	00:20:05	32s	
ADI 4562	01:03:56	01:03:56	01:03:56	0s	01:04:13	17s	
ADI 5260	00:20:45	00:20:45	00:20:45	0s	00:21:14	29s	
ADI 5352	00:15:14	00:15:14	00:15:14	0s	00:18:24	190s	Min. Alexandre de Moraes votou pela

							inconstitucionalidade formal e material. Após divergência do Min. Toffoli, retificou sua posição e votou pela inconstitucionalidade meramente formal. Fachin e Marco Aurélio se manifestaram. Fachin registrou divergência pelo não conhecimento, mas acompanhou.
ADI 5077	00:14:45	00:14:45	00:14:45	0s	00:15:13	28s	
ADI 4913	01:48:31	01:48:38	01:49:12	34s	01:49:23	52s	
ADI 5103	00:47:37	00:47:37	00:47:37	0s	00:49:06	89s	
ADI 5140	00:01:04	00:01:04	00:01:04	0s	00:01:29	25s	
ADI 5462	00:01:50	00:01:50	00:01:50	0s	00:02:10	20s	
ADI 3894	01:13:32	01:13:32	01:13:32	0s	01:13:49	17s	
ADI 3915	02:02:08	02:02:23	02:04:02	99s	02:04:14	126s	
ADI 5961	01:58:50	01:58:50	01:58:50	0s	02:02:38	228s	Vencidos o Min. Relator e o Min. Toffoli. Toffoli

							registrou que isto prova que as listas são julgadas com atenção. Marco Aurélio aponta que "as críticas são grandes, mas é a única forma de viabilizar a jurisdição. Agora se presume que cada qual proceda ao exame do caso."
ADI 1374	01:03:56	01:03:56	01:03:56	0s	01:04:13	17s	
ADI 4314	01:11:39	01:11:39	01:11:39	0s	01:12:12	33s	Barroso impedido.
ADI 5307	00:03:10	00:03:10	00:03:10	0s	00:03:34	24s	
ADI 3659	01:26:57	01:26:57	01:26:57	0s	01:42:22	925s	Registrado como "devolução de vista". Min. Marco Aurélio e Toffoli vencidos na prejudicialidade.
ADI 158	01:13:38	01:13:54	01:18:57	303s	01:20:51	433s	
ADI 3185	01:14:21	01:14:21	01:14:21	0s	01:15:00	39s	
ADI 4984	00:41:17	00:41:42	00:43:49	127s	00:44:26	189s	

ADI 4759	01:13:32	01:13:32	01:13:32	0s	01:13:49	17s	
ADI 2605	01:28:36	01:28:36	01:28:36	0s	01:29:03	27s	Gilmar impedido.
ADI 4633	01:01:08	01:01:08	01:01:08	0s	01:06:33	325s	Marco Aurélio votou pela procedência por inconstitucionalidade formal. Toffoli registra que, apesar do julgamento ser em lista, todos estão atentos. Marco Aurélio ressalta que leu apenas parte do voto em nome da celeridade processual.
ADI 5336	01:05:17	01:05:17	01:05:17	0s	01:06:24	67s	01:06:24
ADI 2898	01:29:07	01:29:21	01:29:24	3s	01:29:29	22s	
ADI 5257	01:08:35	01:08:43	01:08:52	9s	01:09:00	25s	
ADI 1450	01:43:57	01:43:57	01:43:57	0s	01:44:15	18s	
ADI 2304	00:22:50	00:23:06	00:23:45	39s	00:23:53	63s	
ADI 4693	00:02:11	00:02:11	00:02:11	0s	00:02:35	24s	
ADI 4552	01:48:00	01:48:06	01:48:23	17s	01:48:30	30s	
ADI 4613	01:09:01	01:09:12	01:09:14	2s	01:09:36	35s	
ADI 3141	01:20:03	01:20:03	01:20:03	0s	01:20:35	32s	

ADI 1283	01:43:57	01:43:57	01:43:57	0s	01:44:15	18s	
ADI 5044	00:03:35	00:03:35	00:03:35	0s	00:16:50	795s	Fachin, Fux, Carmén Lúcia e Marco Aurélio se manifestaram. Alexandre de Moraes mudou o voto (originalmente votou procedente). Após o debate, Min. Dias Toffoli colheu os votos individualmente. Vencidos Min. Marco Aurélio e Dias Toffoli, que a julgaram improcedente.
ADI 2500	01:45:45	01:45:55	01:46:18	23s	01:46:22	37s	
ADI 5016	00:02:37	00:02:37	00:02:37	0s	00:03:09	32s	
ADI 4382	00:20:07	00:20:07	00:20:07	0s	00:20:44	37s	