



THETA LAB

comexTL



comexTL

Dados do Comércio Exterior Brasileiro no R
Theta Lab

Alan Leal*

<https://alanleal-econ.com/index.php/codes/comextl/>

Resumo

O pacote `comexTL` representa uma solução completa e moderna para acesso e análise de dados de comércio exterior brasileiro diretamente no ambiente R. Desenvolvido para superar as limitações tradicionais de acesso aos dados oficiais do ComexStat (MDIC), o pacote oferece uma interface unificada que elimina a necessidade de downloads manuais, processamento complexo de arquivos e conhecimento detalhado das estruturas de códigos de classificação.

O pacote implementa quatro módulos especializados: (1) gerenciamento inteligente de tabelas de referência com sistema de cache, incluindo códigos NCM, países, UFs e municípios; (2) análise detalhada de exportações por NCM com agregações flexíveis; (3) análise especializada de importações com métricas específicas (FOB, frete, seguro) e cálculo automático de valores CIF; e (4) análise de comércio exterior municipal baseada em dados por SH4 e município da empresa.

A arquitetura do `comexTL` utiliza tecnologia Apache Arrow para processamento eficiente de arquivos Parquet volumosos, hospedados em infraestrutura Cloudflare R2, garantindo alta performance e disponibilidade. O sistema de cache multicamadas otimiza consultas repetidas e permite trabalho offline, enquanto funções de validação automática e tratamento de erros asseguram robustez na análise de dados.

As funcionalidades incluem filtragem hierárquica por códigos SH (SH2, SH4, SH6), análise temporal multi-anual, agregação por variáveis geográficas (país, UF, município), logísticas (via de transporte, URF) e econômicas (valor FOB, peso, quantidade). O pacote calcula automaticamente métricas derivadas relevantes, como participação percentual de frete e seguro nas importações, contagem de municípios únicos em análises regionais, e classificações setoriais baseadas em códigos SH.

Este trabalho documenta integralmente a utilização do pacote, apresentando desde conceitos fundamentais até análises avançadas, incluindo exemplos práticos de dashboards de monitoramento, análises regionais comparativas e integração com outras ferramentas do ecossistema R. A documentação visa atender tanto usuários iniciantes quanto pesquisadores experientes em análise de comércio exterior, oferecendo uma base sólida para estudos acadêmicos e aplicações profissionais no setor.

Palavras-chave: Comércio Exterior, R, ComexStat, Dados Abertos, Análise Econômica, NCM, Sistema Harmonizado, Exportação, Importação.

*Economista. E-mail: prof@alanleal-econ.com.

Conteúdo

1	Introdução	5
1.1	Motivação	5
1.2	Arquitetura do Pacote	5
2	Instalação	5
2.1	Instalação via GitHub	5
2.2	Dependências	6
3	Fonte de Dados e Estrutura	6
3.1	ComexStat - MDIC	6
3.2	Estrutura dos Dados	6
3.2.1	Dados Gerais (NCM detalhado)	6
3.2.2	Dados Municipais (SH4)	7
4	Tabelas de Referência	7
4.1	show_available_tables()	7
4.2	load_reference_tables()	7
4.3	search_in_table()	8
5	Análise de Exportações	8
5.1	comex_stat_geral_exps()	8
6	Análise de Importações	9
6.1	comex_stat_geralimps()	9
7	Análise Municipal	10
7.1	analise_comex_municipal()	10
8	Sistema de Cache	11
8.1	Características do Cache	11
8.2	Funções de Gerenciamento (Internas)	12
9	Exemplos Práticos Avançados	12
9.1	Análise Completa da Balança Comercial	12
9.2	Análise Setorial com Tabelas de Referência	13
9.3	Análise Regional e Municipal Integrada	13
10	Boas Práticas e Otimização	14
10.1	Performance	14
10.2	Tratamento de Erros	14
11	Limitações e Considerações	15
11.1	Limitações Técnicas	15
11.2	Estrutura dos Dados	15
11.3	Qualidade dos Dados	15

12 Desenvolvimento e Contribuições	15
12.1 Estrutura do Código	15
12.2 Como Contribuir	16
12.3 Roadmap Futuro	16
13 Referências	16
13.1 Fontes Oficiais	16
13.2 Recursos Técnicos	16
14 Apêndices	17
14.1 Códigos de Exemplo Completos	17
14.2 Códigos de Referência NCM/SH	21
15 Glossário	22
16 Licença e Agradecimentos	22
16.1 Licença	22
16.2 Agradecimentos	22
16.3 Citação Acadêmica	22
17 Contato e Suporte	23
17.1 Suporte Técnico	23
17.2 Contato do Desenvolvedor	23
17.3 Atualizações e Novidades	23

1 Introdução

O pacote `comexTL` é uma ferramenta avançada desenvolvida em R que facilita o acesso aos dados oficiais de comércio exterior brasileiro diretamente do ambiente R. Este pacote foi criado para simplificar a obtenção, manipulação e análise dos dados do ComexStat (MDIC), eliminando a necessidade de downloads manuais de arquivos volumosos e processamento complexo de bases de dados.

1.1 Motivação

Os dados de comércio exterior brasileiro são fundamentais para análises econômicas, mas tradicionalmente apresentam desafios significativos:

- Arquivos Parquet volumosos hospedados em diferentes locais
- Estruturas complexas com múltiplas tabelas de referência
- Necessidade de conhecimento detalhado das classificações (NCM, SH, países, etc.)
- Processamento manual de dados a nível geral e municipal
- Tratamento específico para dados de exportação e importação

O `comexTL` resolve esses problemas oferecendo uma interface unificada e intuitiva, com sistema de cache inteligente e funções especializadas para diferentes níveis de análise.

1.2 Arquitetura do Pacote

O pacote está estruturado em quatro módulos principais:

- **Tabelas de Referência:** Gerenciamento de códigos NCM, países, UFs e municípios
- **Dados Gerais de Exportação:** Análise detalhada por NCM/SH de exportações
- **Dados Gerais de Importação:** Análise detalhada por NCM/SH de importações
- **Dados Municipais:** Análise por município da empresa (SH4)

2 Instalação

2.1 Instalação via GitHub

O pacote pode ser instalado diretamente do GitHub usando o pacote `devtools`:

```
# Instalar devtools se necessário
if (!require(devtools)) {
  install.packages("devtools")
}

# Instalar comexTL
devtools::install_github("alanleal-econ/comexTL")

# Carregar o pacote
library(comexTL)
```

2.2 Dependências

O pacote comexTL requer as seguintes dependências:

- arrow - Leitura de arquivos Parquet
- dplyr - Manipulação de dados
- readr - Leitura de arquivos CSV
- purrr - Programação funcional
- rlang - Metaprogramação e expressões

3 Fonte de Dados e Estrutura

3.1 ComexStat - MDIC

O pacote acessa dados hospedados em Cloudflare R2, derivados do ComexStat:

- **URL Base:** <https://pub-113e4cc603524d8783e99dec0b56b682.r2.dev>
- **Formato:** Arquivos Parquet para dados principais, CSV para tabelas de referência
- **Cobertura Temporal:** Dados disponíveis por ano
- **Atualizações:** Seguem o cronograma oficial do MDIC

3.2 Estrutura dos Dados

3.2.1 Dados Gerais (NCM detalhado)

- **Exportação:** exportacao/EXP_YYYY.parquet
- **Importação:** importacao/IMP_YYYY.parquet
- **Variáveis:** CO_ANO, CO_MES, CO_NCM, CO_UNID, CO_PAIS, SG_UF_NCM, CO_VIA, CO_URF, QT_ESTAT, KG_LIQUIDO, VL_FOB
- **Importação adicional:** VL_FRETE, VL_SEGURO

3.2.2 Dados Municipais (SH4)

- **Exportação:** municipio/exportacao/EXP_YYYY_MUN.parquet
- **Importação:** municipio/importacao/IMP_YYYY_MUN.parquet
- **Variáveis:** CO_ANO, CO_MES, SH4, CO_PAIS, SG_UF_MUN, CO_MUN, KG_LIQUIDO, VL_FOB

4 Tabelas de Referência

4.1 show_available_tables()

Exibe todas as tabelas de referência disponíveis no pacote.

```
show_available_tables()
```

Tabelas Disponíveis:

- **Código "1":** NCM.csv - Nomenclatura Comum do Mercosul
- **Código "2":** NCM_SH.csv - NCM com códigos SH
- **Código "3":** PAIS.csv - Países
- **Código "4":** UF.csv - Unidades Federativas
- **Código "5":** UF_MUN.csv - UF e Municípios

4.2 load_reference_tables()

Carrega uma ou múltiplas tabelas de referência com sistema de cache.

```
load_reference_tables(  
  codigos_tabelas,  
  force_download = FALSE  
)
```

Parâmetros:

- **codigos_tabelas:** Vetor de códigos ("1", "2", "3", "4", "5")
- **force_download:** Forçar novo download ignorando cache

Exemplos:

```
# Carregar tabela de NCM  
ncm_data <- load_reference_tables("1")  
head(ncm_data)  
  
# Carregar múltiplas tabelas  
ref_tables <- load_reference_tables(c("3", "4"))  
names(ref_tables)  
head(ref_tables$tabela_3_pais)  
  
# Forçar novo download  
pais_fresh <- load_reference_tables("3", force_download = TRUE)
```

4.3 search_in_table()

Busca termos específicos dentro das tabelas de referência.

```
search_in_table(codigo_tabela, termo_busca, coluna = NULL)
```

Exemplos:

```
# Buscar "café" na tabela NCM
cafe_ncm <- search_in_table("1", "café")
print(head(cafe_ncm))

# Buscar países com "Unidos"
países_unidos <- search_in_table("3", "Unidos", coluna = "NO_PAIS")
print(países_unidos)

# Buscar municípios de São Paulo
sp_municipios <- search_in_table("5", "São Paulo")
```

5 Análise de Exportações

5.1 comex_stat_geral_exps()

Função principal para análise de dados de exportação a nível geral (detalhado por NCM).

```
comex_stat_geral_exps(
  tipo = "exportacao",
  anos = 2024,
  sh2 = NULL,
  sh4 = NULL,
  sh6 = NULL,
  co_pais = NULL,
  sg_uf_ncm = NULL,
  co_urf = NULL,
  co_via = NULL,
  co_unid = NULL,
  agregar_por = NULL,
  metricas = c("VL_FOB", "KG_LIQUIDO", "QT_ESTAT"),
  force_download = FALSE
)
```

Parâmetros:

- tipo: "exportacao"(padrão para esta função)
- anos: Vetor de anos (ex: 2024 ou c(2023, 2024))
- sh2/sh4/sh6: Códigos de produto (hierárquicos)
- co_pais: Códigos de países de destino
- sg_uf_ncm: Siglas das UFs de origem
- co_urf: Códigos das URFs
- co_via: Códigos das vias de transporte

- `co_unid`: Códigos das unidades de medida
- `agregar_por`: Variáveis adicionais para agrupar
- `metricas`: Métricas a calcular

Exemplos:

```
# Exemplo 1: Exportações de soja (SH2 = 12) em 2024 por UF
exp_soja_2024 <- comex_stat_geral_exps(
  anos = 2024,
  sh2 = 12,
  agregar_por = c("CO_ANO", "SG_UF_NCM")
)
print(head(exp_soja_2024))

# Exemplo 2: Análise multi-anual de exportações específicas
exp_multianos <- comex_stat_geral_exps(
  anos = c(2022, 2023, 2024),
  sh6 = c(120100, 120190), # Soja específica
  co_pais = c(132, 189),   # China e Estados Unidos
  agregar_por = c("CO_ANO", "CO_PAIS")
)
print(head(exp_multianos))

# Exemplo 3: Exportações por via de transporte
exp_por_via <- comex_stat_geral_exps(
  anos = 2024,
  sh2 = 26, # Minérios
  agregar_por = c("CO_VIA"),
  metricas = c("VL_FOB", "KG_LIQUIDO")
)
print(head(exp_por_via))
```

6 Análise de Importações

6.1 `comex_stat_geral_imps()`

Função especializada para análise de dados de importação com métricas específicas.

```
comex_stat_geral_imps(
  anos = 2024,
  sh2 = NULL,
  sh4 = NULL,
  sh6 = NULL,
  co_pais = NULL,
  sg_uf_ncm = NULL,
  co_urf = NULL,
  co_via = NULL,
  co_unid = NULL,
  agregar_por = NULL,
  metricas = c("VL_FOB", "KG_LIQUIDO", "QT_ESTAT", "VL_FRETE", "VL_SEGURO"),
  force_download = FALSE
)
```

Métricas Específicas de Importação:

- VL_FOB: Valor FOB em US\$
- VL_FRETE: Valor do frete em US\$
- VL_SEGURO: Valor do seguro em US\$
- **Métricas Derivadas:** VL_TOTAL_CIF, PERC_FRETE, PERC_SEGURO

Exemplos:

```
# Exemplo 1: Importações básicas por SH6
imp_equipamentos <- comex_stat_geral_imps(
  anos = 2024,
  sh6 = c(901110, 901120), # Equipamentos médicos
  agregar_por = c("CO_PAIS")
)
print(head(imp_equipamentos))

# Exemplo 2: Análise de custos de importação por país
imp_custos <- comex_stat_geral_imps(
  anos = c(2023, 2024),
  sh2 = 85, # Máquinas elétricas
  co_pais = c(132, 249), # China e EUA
  agregar_por = c("CO_ANO", "CO_PAIS"),
  metricas = c("VL_FOB", "VL_FRETE", "VL_SEGURO")
)
print(head(imp_custos))

# Verificar métricas derivadas
print(names(imp_custos))
# Deve incluir: VL_TOTAL_CIF, PERC_FRETE, PERC_SEGURO
```

7 Análise Municipal

7.1 analise_comex_municipal()

Função para análise de comércio exterior por município da empresa (nível SH4).

```
analise_comex_municipal(
  tipo = "exportacao",
  anos = 2024,
  sh2 = NULL,
  sh4 = NULL,
  co_pais = NULL,
  sg_uf_mun = NULL,
  co_mun = NULL,
  agregar_por = NULL,
  metricas = c("VL_FOB", "KG_LIQUIDO"),
  force_download = FALSE
)
```

Características Especiais:

- Dados limitados a SH4 (não há NCM ou SH6)

- SH2 derivado automaticamente de SH4
- Métricas limitadas: VL_FOB e KG_LIQUIDO
- Adiciona N_MUNICIPIOS quando não agregando por CO_MUN

Exemplos:

```
# Exemplo 1: Exportações municipais por UF
exp_mun_uf <- analise_comex_municipal(
  tipo = "exportacao",
  anos = 2024,
  sh4 = c(9011, 9012),
  agregar_por = "SG_UF_MUN"
)
print(head(exp_mun_uf))
# Inclui coluna N_MUNICIPIOS

# Exemplo 2: Importações de São Paulo por produto
imp_sp <- analise_comex_municipal(
  tipo = "importacao",
  anos = c(2023, 2024),
  co_mun = 3550308, # São Paulo
  agregar_por = c("CO_ANO", "SH4")
)
print(head(imp_sp))

# Exemplo 3: Análise por SH2 derivado
exp_sh2_municipal <- analise_comex_municipal(
  tipo = "exportacao",
  anos = 2024,
  sh2 = 85, # Máquinas elétricas
  sg_uf_mun = c("SP", "RJ", "MG"),
  agregar_por = c("SG_UF_MUN", "CO_MES")
)
print(head(exp_sh2_municipal))
```

8 Sistema de Cache

8.1 Características do Cache

O pacote implementa três sistemas de cache independentes:

- `.comex_tabelas_cache`: Tabelas de referência
- `.comexstat_cache`: Dados gerais de exportação
- `.import_cache`: Dados gerais de importação
- `.municipal_cache`: Dados municipais

8.2 Funções de Gerenciamento (Internas)

Embora não exportadas, o pacote inclui funções internas para gerenciar cache:

```
# Listar arquivos em cache (função interna)
# comexTL:::list_cached_files()
# comexTL:::list_cached_import_files()
# comexTL:::list_cached_municipal_files()

# Limpar cache específico (função interna)
# comexTL:::clear_cache()
# comexTL:::clear_import_cache()
# comexTL:::clear_municipal_cache()
```

9 Exemplos Práticos Avançados

9.1 Análise Completa da Balança Comercial

```
library(comexTL)
library(dplyr)
library(ggplot2)

# Carregar dados de exportação e importação para 2024
exp_2024 <- comex_stat_geral_exps(
  anos = 2024,
  agregar_por = c("CO_ANO", "CO_MES")
)

imp_2024 <- comex_stat_geralimps(
  anos = 2024,
  agregar_por = c("CO_ANO", "CO_MES"),
  metricas = c("VL_FOB")
)

# Calcular balança comercial mensal
balanca_2024 <- exp_2024 %>%
  select(CO_ANO, CO_MES, VL_EXPORTACAO = VL_FOB) %>%
  left_join(
    imp_2024 %>% select(CO_ANO, CO_MES, VL_IMPORTACAO = VL_FOB),
    by = c("CO_ANO", "CO_MES")
  ) %>%
  mutate(
    SALDO = VL_EXPORTACAO - VL_IMPORTACAO,
    DATA = as.Date(paste(CO_ANO, CO_MES, "01", sep = "-"))
  )

# Visualização
ggplot(balanca_2024, aes(x = DATA)) +
  geom_line(aes(y = VL_EXPORTACAO / 1e6, color = "Exportações")) +
  geom_line(aes(y = VL_IMPORTACAO / 1e6, color = "Importações")) +
  geom_line(aes(y = SALDO / 1e6, color = "Saldo")) +
  labs(title = "Balança Comercial Brasileira - 2024",
       x = "Mês", y = "Valor (US$ milhões)",
       color = "Legenda") +
  theme_minimal()
```

9.2 Análise Setorial com Tabelas de Referência

```
# Carregar tabela NCM para classificação setorial
ncm_table <- load_reference_tables("1")

# Exportações por grandes setores
exp_setores <- comex_stat_geral_exps(
  anos = 2024,
  agregar_por = c("sh2")
) %>%
  mutate(
    SETOR = case_when(
      sh2 >= 1 & sh2 <= 15 ~ "Agropecuária",
      sh2 >= 25 & sh2 <= 27 ~ "Mineração",
      sh2 >= 28 & sh2 <= 38 ~ "Química",
      sh2 >= 84 & sh2 <= 85 ~ "Máquinas",
      TRUE ~ "Outros"
    )
  ) %>%
  group_by(SETOR) %>%
  summarise(
    VL_FOB_TOTAL = sum(VL_FOB, na.rm = TRUE),
    KG_TOTAL = sum(KG_LIQUIDO, na.rm = TRUE),
    .groups = "drop"
  ) %>%
  arrange(desc(VL_FOB_TOTAL))

print(exp_setores)
```

9.3 Análise Regional e Municipal Integrada

```
# Carregar tabelas de referência
uf_table <- load_reference_tables("4")
municipios_table <- load_reference_tables("5")

# Análise de exportações por região (dados gerais)
exp_regional <- comex_stat_geral_exps(
  anos = 2024,
  sh2 = 12, # Soja
  agregar_por = c("SG_UF_NCM")
) %>%
  left_join(uf_table, by = c("SG_UF_NCM" = "SG_UF")) %>%
  mutate(
    REGIAO = case_when(
      SG_UF_NCM %in% c("AC", "AM", "AP", "PA", "RO", "RR", "TO") ~ "Norte",
      SG_UF_NCM %in% c("AL", "BA", "CE", "MA", "PB", "PE", "PI", "RN", "SE") ~ "Nordeste",
      SG_UF_NCM %in% c("DF", "GO", "MT", "MS") ~ "Centro-Oeste",
      SG_UF_NCM %in% c("ES", "MG", "RJ", "SP") ~ "Sudeste",
      SG_UF_NCM %in% c("PR", "RS", "SC") ~ "Sul",
      TRUE ~ "Outros"
    )
  ) %>%
  group_by(REGIAO) %>%
  summarise(VL_FOB_TOTAL = sum(VL_FOB, na.rm = TRUE), .groups = "drop")
```

```
# Análise municipal detalhada para estados selecionados
exp_municipal_detalhada <- analise_comex_municipal(
  tipo = "exportacao",
  anos = 2024,
  sh2 = 12,
  sg_uf_mun = c("MT", "RS", "GO"),
  agregar_por = c("SG_UF_MUN", "CO_MUN")
) %>%
  left_join(
    municipios_table %>% select(CO_MUN, NO_MUN_MIN),
    by = "CO_MUN"
  ) %>%
  arrange(desc(VL_FOB))

print("Top 10 municípios exportadores de soja:")
print(head(exp_municipal_detalhada, 10))
```

10 Boas Práticas e Otimização

10.1 Performance

Para melhor performance:

- Use filtros específicos sempre que possível
- Aproveite o sistema de cache - evite `force_download = TRUE`
- Para análises exploratórias, limite os anos inicialmente
- Use hierarquia SH6 > SH4 > SH2 conforme necessidade

Exemplo otimizado:

```
# Melhor: análise específica
resultado_otimizado <- comex_stat_geral_exps(
  anos = 2024,
  sh4 = c(1201, 1202), # Soja específica
  co_pais = c(132, 189), # China e EUA
  agregar_por = c("CO_PAIS")
)

# Evitar: análise muito ampla sem filtros
# resultado_lento <- comex_stat_geral_exps(anos = c(2020:2024))
```

10.2 Tratamento de Erros

```
# Sempre verificar se dados foram encontrados
resultado <- comex_stat_geral_exps(
  anos = 2024,
  sh6 = 999999 # Código inexistente
)
```

```
if (nrow(resultado) == 0) {  
  cat("Nenhum dado encontrado com os filtros aplicados\n")  
  # Verificar códigos válidos  
  ncm_validos <- search_in_table("1", "produto_interesse")  
} else {  
  print(head(resultado))  
}
```

11 Limitações e Considerações

11.1 Limitações Técnicas

- **Dependência de conectividade:** Requer internet para downloads iniciais
- **Tamanho dos arquivos:** Arquivos anuais podem ser grandes (>100MB)
- **Memória:** Análises multi-anuais podem consumir memória significativa
- **Dados municipais:** Limitados a SH4, sem detalhamento NCM

11.2 Estrutura dos Dados

- **Dados gerais:** NCM completo, mas sem informação municipal da empresa
- **Dados municipais:** Município da empresa, mas apenas SH4
- **Importação vs Exportação:** Importação tem métricas adicionais (frete, seguro)
- **Vias e URFs:** Códigos numéricos que requerem tabelas de referência para interpretação

11.3 Qualidade dos Dados

- Dados derivados do ComexStat oficial (MDIC)
- Sujeitos a revisões e correções posteriores
- Possíveis inconsistências em códigos de classificação
- Defasagem temporal conforme cronograma oficial

12 Desenvolvimento e Contribuições

12.1 Estrutura do Código

O pacote está organizado em módulos especializados:

- `comex_stat_tabelas.R`: Gerenciamento de tabelas de referência
- `comex_stat_geral_exps.R`: Dados gerais de exportação

- `comex_stat_geralimps.R`: Dados gerais de importação
- `analise_comex_municipal.R`: Dados municipais

12.2 Como Contribuir

O projeto é open-source e aceita contribuições:

- **Issues**: Reporte bugs ou sugira funcionalidades
- **Pull Requests**: Contribua com código
- **Documentação**: Melhore exemplos e explicações
- **Testes**: Adicione casos de teste

12.3 Roadmap Futuro

Funcionalidades planejadas:

- Interface gráfica Shiny para usuários não-programadores
- Integração com dados históricos (pré-2020)
- Funções especializadas para análises setoriais
- Export para diferentes formatos (Excel, PowerBI)
- Dashboards automáticos de monitoramento

13 Referências

13.1 Fontes Oficiais

- ComexStat - MDIC: <https://comexstat.mdic.gov.br/>
- Dados Abertos: <https://www.gov.br/produtividade-e-comercio-exterior/pt-br/assuntos/comercio-exterior/estatisticas>
- Manual Metodológico: Secretaria de Comércio Exterior (SECEX)

13.2 Recursos Técnicos

- Repositório GitHub: <https://github.com/alanleal-econ/comexTL>
- Cloudflare R2: <https://pub-113e4cc603524d8783e99dec0b56b682.r2.dev>
- Apache Arrow: <https://arrow.apache.org/docs/r/>

14 Apêndices

14.1 Códigos de Exemplo Completos

A.1 - Dashboard Básico de Comércio Exterior

```
library(comexTL)
library(dplyr)
library(ggplot2)
library(plotly)

# Função para criar dashboard básico
criar_dashboard_comex <- function(ano = 2024) {

  # 1. Carregar dados básicos
  cat("Carregando dados de exportação...\n")
  exp_data <- comex_stat_geral_exps(
    anos = ano,
    agregar_por = c("CO_MES", "sh2")
  )

  cat("Carregando dados de importação...\n")
  imp_data <- comex_stat_geralimps(
    anos = ano,
    agregar_por = c("CO_MES", "sh2"),
    metricas = c("VL_FOB")
  )

  # 2. Preparar dados mensais
  dados_mensais <- exp_data %>%
    group_by(CO_MES) %>%
    summarise(EXPORTACOES = sum(VL_FOB, na.rm = TRUE), .groups = "drop"
    ) %>%
    left_join(
      imp_data %>%
        group_by(CO_MES) %>%
        summarise(IMPORTACOES = sum(VL_FOB, na.rm = TRUE), .groups
          = "drop"),
      by = "CO_MES"
    ) %>%
    mutate(
      SALDO = EXPORTACOES - IMPORTACOES,
      MES_NOME = month.name[CO_MES]
    )

  # 3. Top setores de exportação
  top_setores_exp <- exp_data %>%
    mutate(
      SETOR = case_when(
        sh2 >= 1 & sh2 <= 15 ~ "Agropecuária",
        sh2 >= 25 & sh2 <= 27 ~ "Mineração",
        sh2 >= 28 & sh2 <= 38 ~ "Química",
        sh2 >= 84 & sh2 <= 85 ~ "Máquinas",
        TRUE ~ "Outros"
      )
    ) %>%
    group_by(SETOR) %>%
    summarise(VLOR = sum(VL_FOB, na.rm = TRUE), .groups = "drop") %>%
```

```

    arrange(desc(VALOR)) %>%
    slice_head(n = 5)

# 4. Criar gráficos
g1 <- ggplot(dados_mensais, aes(x = CO_MES)) +
  geom_line(aes(y = EXPORTACOES/1e9, color = "Exportações"), size =
    1.2) +
  geom_line(aes(y = IMPORTACOES/1e9, color = "Importações"), size =
    1.2) +
  geom_line(aes(y = SALDO/1e9, color = "Saldo"), size = 1.2) +
  labs(title = paste("Balança Comercial Brasileira -", ano),
    x = "Mês", y = "Valor (US$ bilhões)",
    color = "Legenda") +
  theme_minimal() +
  theme(legend.position = "bottom")

g2 <- ggplot(top_setores_exp, aes(x = reorder(SETOR, VALOR), y = VALOR/
  1e9)) +
  geom_col(fill = "steelblue", alpha = 0.8) +
  coord_flip() +
  labs(title = "Top 5 Setores - Exportações",
    x = "Setor", y = "Valor (US$ bilhões)") +
  theme_minimal()

# 5. Retornar resultados
return(list(
  dados_mensais = dados_mensais,
  top_setores = top_setores_exp,
  grafico_mensal = g1,
  grafico_setores = g2
))
}

# Usar a função
dashboard <- criar_dashboard_comex(2024)
print(dashboard$grafico_mensal)
print(dashboard$grafico_setores)

```

A.2 - Análise Comparativa Multi-Regional

```

# Função para análise regional comparativa
analise_regional_completa <- function(anos = c(2023, 2024), produto_sh2 =
  12) {

  # Carregar tabela de UFs
  uf_table <- load_reference_tables("4")

  # Definir regiões
  regioes <- list(
    Norte = c("AC", "AM", "AP", "PA", "RO", "RR", "TO"),
    Nordeste = c("AL", "BA", "CE", "MA", "PB", "PE", "PI", "RN", "SE"),
    CentroOeste = c("DF", "GO", "MT", "MS"),
    Sudeste = c("ES", "MG", "RJ", "SP"),
    Sul = c("PR", "RS", "SC")
  )

  # Dados gerais por UF
  dados_uf <- comex_stat_geral_exps(
    anos = anos,

```

```

    sh2 = produto_sh2,
    agregar_por = c("CO_ANO", "SG_UF_NCM")
  )

  # Dados municipais para análise detalhada
  dados_municipais <- analise_comex_municipal(
    tipo = "exportacao",
    anos = anos,
    sh2 = produto_sh2,
    agregar_por = c("CO_ANO", "SG_UF_MUN", "CO_MUN")
  )

  # Agrupar por região
  dados_regionais <- dados_uf %>%
    mutate(
      REGIAO = case_when(
        SG_UF_NCM %in% regioes$Norte ~ "Norte",
        SG_UF_NCM %in% regioes$Nordeste ~ "Nordeste",
        SG_UF_NCM %in% regioes$CentroOeste ~ "Centro-Oeste",
        SG_UF_NCM %in% regioes$Sudeste ~ "Sudeste",
        SG_UF_NCM %in% regioes$Sul ~ "Sul",
        TRUE ~ "Outros"
      )
    ) %>%
    filter(REGIAO != "Outros") %>%
    group_by(CO_ANO, REGIAO) %>%
    summarise(
      VL_FOB_TOTAL = sum(VL_FOB, na.rm = TRUE),
      KG_TOTAL = sum(KG_LIQUIDO, na.rm = TRUE),
      N_UFS = n_distinct(SG_UF_NCM),
      .groups = "drop"
    )

  # Calcular crescimento anual
  crescimento <- dados_regionais %>%
    arrange(REGIAO, CO_ANO) %>%
    group_by(REGIAO) %>%
    mutate(
      CRESCIMENTO_VALOR = (VL_FOB_TOTAL / lag(VL_FOB_TOTAL) - 1) *
        100,
      CRESCIMENTO_VOLUME = (KG_TOTAL / lag(KG_TOTAL) - 1) * 100
    ) %>%
    filter(!is.na(CRESCIMENTO_VALOR))

  return(list(
    dados_uf = dados_uf,
    dados_municipais = dados_municipais,
    dados_regionais = dados_regionais,
    crescimento = crescimento
  ))
}

# Executar análise para soja
analise_soja <- analise_regional_completa(c(2023, 2024), 12)
print("Crescimento regional - Soja:")
print(analise_soja$crescimento)

```

A.3 - Sistema de Monitoramento de Performance

```
# Função para monitorar performance do sistema
monitorar_sistema <- function() {

  # Verificar status do cache
  cat("=== STATUS DO SISTEMA comexTL ===\n\n")

  # Cache de tabelas
  cat("\n1. CACHE DE TABELAS DE REFERÊNCIA:\n")
  tabelas_cache <- tryCatch({
    comexTL:::list_cached_tables()
  }, error = function(e) {
    cat("Erro ao acessar cache de tabelas:", e$message, "\n")
    return(NULL)
  })

  # Cache de exportações
  cat("\n2. CACHE DE EXPORTAÇÕES:\n")
  exp_cache <- tryCatch({
    comexTL:::list_cached_files()
  }, error = function(e) {
    cat("Erro ao acessar cache de exportações:", e$message, "\n")
    return(NULL)
  })

  # Cache de importações
  cat("\n3. CACHE DE IMPORTAÇÕES:\n")
  imp_cache <- tryCatch({
    comexTL:::list_cached_import_files()
  }, error = function(e) {
    cat("Erro ao acessar cache de importações:", e$message, "\n")
    return(NULL)
  })

  # Cache municipal
  cat("\n4. CACHE MUNICIPAL:\n")
  mun_cache <- tryCatch({
    comexTL:::list_cached_municipal_files()
  }, error = function(e) {
    cat("Erro ao acessar cache municipal:", e$message, "\n")
    return(NULL)
  })

  # Teste de conectividade
  cat("\n5. TESTE DE CONECTIVIDADE:\n")
  conectividade <- tryCatch({
    teste_url <- "https://pub-113e4cc603524d8783e99dec0b56b682.r2.dev/
    tabelas/PAIS.csv"
    teste <- readLines(teste_url, n = 1)
    cat(" Conexão com Cloudflare R2: OK\n")
    TRUE
  }, error = function(e) {
    cat(" Erro de conectividade:", e$message, "\n")
    FALSE
  })

  # Resumo do sistema
  cat("\n6. RESUMO:\n")
}
```

```

cat("- Tabelas em cache:", ifelse(is.null(tabelas_cache), 0, length(
  tabelas_cache)), "\n")
cat("- Arquivos exportação:", ifelse(is.null(exp_cache), 0, length(exp_
  cache)), "\n")
cat("- Arquivos importação:", ifelse(is.null(imp_cache), 0, length(imp_
  cache)), "\n")
cat("- Arquivos municipais:", ifelse(is.null(mun_cache), 0, length(mun_
  cache)), "\n")
cat("- Conectividade:", ifelse(conectividade, "OK", "FALHA"), "\n")

return(invisible(list(
  tabelas = tabelas_cache,
  exportacoes = exp_cache,
  importacoes = imp_cache,
  municipais = mun_cache,
  conectividade = conectividade
)))
}

# Executar monitoramento
status_sistema <- monitorar_sistema()

```

14.2 Códigos de Referência NCM/SH

B.1 - Principais Códigos SH2

SH2	Descrição
01-05	Animais vivos e produtos do reino animal
06-15	Produtos do reino vegetal
16-24	Gêneros alimentícios; bebidas, líquidos alcoólicos
25-27	Produtos minerais
28-38	Produtos das indústrias químicas
39-40	Plásticos e borracha
41-43	Peles, couros, peleteria
44-49	Madeira, carvão vegetal, cortiça, palha
50-63	Matérias têxteis e suas obras
64-67	Calçados, chapéus, guarda-chuvas
68-71	Pedra, gesso, cimento, amianto, mica
72-83	Metais comuns e suas obras
84-85	Máquinas, aparelhos, material elétrico
86-89	Material de transporte
90-97	Instrumentos de precisão, ótica, relojoaria

B.2 - Principais Países (Códigos ComexStat)

Código	País	Código	País
132	China	249	Estados Unidos
160	Argentina	169	Alemanha
155	Países Baixos	158	Chile
185	Uruguai	178	Paraguai
174	Peru	136	Coreia do Sul

179	Bolívia	137	Espanha
147	Japão	115	Bélgica
151	México	142	Índia
170	França	145	Itália

15 Glossário

ComexStat Sistema oficial de estatísticas de comércio exterior do MDIC

NCM Nomenclatura Comum do Mercosul - classificação de 8 dígitos

SH Sistema Harmonizado - classificação internacional (2, 4 ou 6 dígitos)

FOB Free On Board - valor da mercadoria no porto de embarque

CIF Cost, Insurance and Freight - valor FOB + frete + seguro

UF_NCM Unidade Federativa onde foi produzida/beneficiada a mercadoria

UF_MUN Unidade Federativa do domicílio fiscal da empresa

URF Unidade da Receita Federal responsável pelo despacho

VIA Meio de transporte utilizado (aéreo, marítimo, terrestre, etc.)

UNID Unidade de medida estatística da mercadoria

16 Licença e Agradecimentos

16.1 Licença

Este pacote é distribuído sob licença MIT, permitindo uso livre para fins acadêmicos e comerciais. Para detalhes completos, consulte o arquivo LICENSE no repositório GitHub.

16.2 Agradecimentos

- **Secretaria de Comércio Exterior (SECEX/MDIC):** Pela disponibilização dos dados abertos
- **Apache Arrow:** Pela tecnologia que permite processamento eficiente de grandes volumes
- **Cloudflare:** Pela infraestrutura de distribuição de dados

16.3 Citação Acadêmica

Para citar este pacote em trabalhos acadêmicos:

Leal, A. (2024). *comexTL: Dados do Comércio Exterior Brasileiro no R por Theta Lab*. R package. Disponível em: <https://github.com/alanleal-econ/comexTL>

17 Contato e Suporte

17.1 Suporte Técnico

- **Issues GitHub:** <https://github.com/alanleal-econ/comexTL/issues>
- **Discussões:** <https://github.com/alanleal-econ/comexTL/discussions>
- **Pull Requests:** Contribuições são bem-vindas

17.2 Contato do Desenvolvedor

- **GitHub:** <https://github.com/alanleal-econ>
- **Perfil:** Economista especializado em comércio exterior, meio ambiente e clima. Esse código faz parte do esforço de ampliar o entendimento do comércio exterior, promovido pelo Theta Lab.

17.3 Atualizações e Novidades

Para acompanhar atualizações do pacote:

- Marque como "Watch" o repositório no GitHub
- Consulte regularmente a seção "Releases"
- Verifique o arquivo NEWS.md para mudanças detalhadas

—

Documentação gerada em 9 de agosto de 2025. Versão do pacote: em desenvolvimento.