



**LISBOA
SCHOOL OF
ECONOMICS &
MANAGEMENT**

MESTRADO

MÉTODOS QUANTITATIVOS PARA A DECISÃO ECONÓMICA EMPRESARIAL

TRABALHO FINAL DE MESTRADO

RELATÓRIO DE ESTÁGIO

ANÁLISE DOS PRESTADORES DE SAÚDE

CARLA JOÃO RODRIGUES CARNEIRO

OUTUBRO - 2018



**LISBOA
SCHOOL OF
ECONOMICS &
MANAGEMENT**

**MESTRADO EM
MÉTODOS QUANTITATIVOS PARA A DECISÃO
ECONÓMICA EMPRESARIAL**

**TRABALHO FINAL DE MESTRADO
RELATÓRIO DE ESTÁGIO**

ANÁLISE DOS PRESTADORES DE SAÚDE

CARLA JOÃO RODRIGUES CARNEIRO

ORIENTAÇÃO:

PROF. DR. JOSÉ PEDRO GAIVÃO

DR. DANIEL SILVA

OUTUBRO - 2018

Agradecimentos

A realização deste trabalho não seria possível sem o apoio de algumas pessoas.

Primeiramente, agradeço aos meus pais e ao meu irmão, que sempre me apoiaram incondicionalmente em todas as minhas escolhas académicas e pessoais e além disso estiveram sempre presentes e disponíveis para que as conseguisse conquistar.

Quero agradecer às colegas de casa e amigas que se tornaram em família. Durante todo este percurso foram fundamentais e permitiram que tudo fosse mais fácil.

Agradeço ao meu colega Daniel Tomás que me acompanhou no estágio e foi um pilar durante a realização deste projeto.

Quero também agradecer à coordenação do mestrado por toda a disponibilidade e acompanhamento.

Estou grata ao Professor Doutor José Pedro Gaivão por todo o incentivo e disponibilidade para esclarecer todas as dúvidas e objetivos ao longo da realização do estágio e do relatório e foi incansável comigo. Agradeço ainda ao Doutor Daniel Silva que me orientou durante o estágio na empresa e ajudou a traçar objetivos.

Por fim, agradeço à Quidgest, S. A. pela oportunidade de estágio e, conseqüentemente, todo o conhecimento facultado.

Resumo

O presente relatório final de mestrado tem por base o estágio decorrido na Quidgest, S. A, teve como finalidade a Análise de Prestadores de Saúde, através da integração do *R* com o *SQL Server*.

Esta análise surgiu da necessidade de auxiliar todos os membros do Departamento da Saúde, que necessitavam de perceber qual a disposição geográfica e os dados disponíveis sobre os prestadores de saúde.

Os relatórios desenvolvidos através de indicadores apresentados em gráficos e tabelas possibilitam a célere utilização e interpretação por parte de todos os utilizadores do GENIO, uma vez que todos os relatórios foram integrados nesta ferramenta que a Quidgest detém.

Neste trabalho foi caracterizado o comportamento do rácio, isto é, o número de atos médicos sobre o número de prestadores de saúde com o auxílio de diversas ferramentas e metodologias. Verificou-se ainda que existem evidências na tendência da prestação de serviços de saúde consoante os meses do ano.

Palavras-chave: Prestadores de saúde, Atos Médicos, *SQL Server*, ambiente *R*, *stored procedures*, Relatórios, GENIO

Abstract

This final master's report is based on the internship at Quidgest, S.A, whose purpose was to Analyze Health Providers through the integration of R with SQL Server.

This analysis emerged in the need to assist all members of the Department of Health, who needed to understand the geographic disposition and the available data on health care providers.

The reports developed through indicators presented in graphs and tables enable the fast use and interpretation by all users of GENIO, since all the reports have been integrated in this tool that Quidgest owns.

In this work, it was characterized the behaviour of the ratio, that is, the number of medical acts on the number of health care providers with help of various tools and methodologies. It was also verified there is evidence in the trend of the provision of health services according to the months of the year.

Key-words: Health providers, Medical acts, *SQL Server*, *R*, *stored procedures*, *Reports*, *GENIO*

Índice

Agradecimentos	II
Resumo	III
Abstract	IV
Índice	V
Índice de Tabelas	VI
Índice de Figuras	VII
Acrónimos	IX
Introdução	1
Parte I: Apresentação da empresa	3
Parte II: Enquadramento teórico	6
1.1 R	9
1.2 SQL	10
1.4 Integração no GENIO	12
1.5 Composição de uma série temporal	12
Parte III: Análise empírica	14
1. Objetivos	14
2. Dados e descrição das variáveis	14
3. Limitações à análise	21
Integridade dos dados:	21
Tratamento dos dados:	21
Inserção de dados:	21
A integração do R com o SQL Server	22
Análise de Resultados	24
Caracterização de uma série temporal sobre o rácio	25
Alisamento Exponencial	29
Holt-Winters	30
Conclusão	32
Referências bibliográficas	34
ANEXOS	36
Anexo I - Stored Procedures	36
Anexo II - Relatórios	52

Índice de Tabelas

Tabela I - N ^o prestadores de saúde	14
Tabela II - Decomposição da base de dados	15
Tabela III - Erro absoluto médio	31

Índice de Figuras

Figura 1 – Médias mensais do relatório “Meses”	19
Figura 2 - Processo de integração do SQL Server com o R.....	22
Figura 3 - Decomposição de uma série temporal.....	25
Figura 4 - Outliers	27
Figura 5 - Previsão com o modelo Alisamento Exponencial.....	29
Figura 6 - Previsão com o modelo Holt-Winters	30
Figura 7 - Código do stored procedure 'prestador'.....	36
Figura 8 - Código do stored procedure "oftalmologia"	37
Figura 9 - Código do stored procedure "concelho".....	38
Figura 10 - Código do stored procedure "registo"	39
Figura 11 - Código do stored procedure "teste1"	40
Figura 12 - Código do stored procedure "graficomes"	41
Figura 13 - Código do stored procedure "st"	42
Figura 14 - Código do stored procedure "decompose"	43
Figura 15 - Código do stored procedure "outlier"	44
Figura 16 - Código do stored procedure "alisamentoex"	45
Figura 17 - Código do stored procedure "previsao"	46
Figura 18 - Código do stored procedure "racio"	47
Figura 19 - Código do stored procedure "provincia"	48
Figura 20 - Código do stored procedure "tabela"	48
Figura 21 - Código do stored procedure "tabela1"	49
Figura 22 - Código do stored procedure "tabela2"	49
Figura 23 - Código do stored procedure "tabela3"	49
Figura 24 - Código do stored procedure "tabeladescr"	50
Figura 25 - Código do stored procedure "tabelaout"	50
Figura 26 - Código do stored procedure "conclusao"	51
Figura 27 - Output do relatório "NrPrestadoresDist"	52

Figura 28 - Output do relatório "Nr Atos Concelho"	53
Figura 29 - Output do relatório "Concelhos"	54
Figura 30 - Output do relatório "Registo Mensal"	55
Figura 31 - Output do relatório "Descrição"	56
Figura 32 - Output do relatório "Rácio"	57
Figura 33 - Output do relatório "Série Temporal"	58
Figura 34 - Output do relatório "Decomposição"	59
Figura 35 - Output do relatório "Outlier"	60
Figura 36 - Output do relatório "Previsão AE"	61
Figura 37 - Output do relatório "Previsao HW"	62

Acrónimos

BSC - *Balanced Scorecard*

SGBD – *Sistema de gestão de base dados*

SMSS - *SQL Server Management Studio*

SSRS - *SQL Server Reporting Services*

MVS - *Microsoft Visual Studio*

SQL - *Structured Query Language*

SSAS - *SQL Server Analysis Services*

Introdução

A Quidgest privilegia da experiência e de uma perspetiva tecnológica que lhe permite estar presente em diversos projetos e continentes.

A forma inovadora de desenvolver *software* confirma-se no sucesso em Portugal e em mercados e clientes um pouco por todo o mundo. A enumerar os sucessos está a informatização da Administração Pública e o seu sistema de gestão SINGAP.

A razão pela qual foi considerado ingressar neste estágio deve-se sobretudo às características que a Quidgest apresenta no seu dia-a-dia e pelo desafio que seria explorar uma área com pouco contacto anteriormente - a programação em *R* e *Structured Query Language (SQL)*.

No início do estágio foram estabelecidas algumas metas em concordância com os orientadores do estágio para os seguintes meses de aprendizagem. Inicialmente, foi necessário adquirir conhecimento nos diferentes sistemas de gestão e de programação, através de formações e *workshops*. Posteriormente, foi feito o tratamento e análise dos dados disponibilizados pela empresa referentes à área da saúde de modo a estruturar os dados, seguido do desenvolvimento de relatórios através da integração do *SQL Server* com o *R*. O objetivo final era a implementação dos relatórios no GENIO de modo a permitir uma análise fácil dos dados por parte dos colaboradores da Quidgest.

Com isto, o presente relatório é composto por três partes que explicam todo o desenvolvimento e os resultados obtidos durante o estágio.

A primeira parte do relatório explana a história, a estrutura organizacional e os sistemas de gestão utilizados pela Quidgest. A segunda parte envolve o enquadramento teórico das metodologias e ferramentas utilizadas de modo a compreender a dimensão dos conceitos aplicados ao longo do relatório. A análise

empírica faz parte da terceira parte, isto é, a análise, a descrição dos dados e a identificação de problemas e todas as conclusões associadas.

Os dados deste relatório referem-se a prestadores de serviços de saúde convencionados a uma entidade privada. Este relatório, inclui também um *case study* sobre a integração do R com o *SQL Server* tal como um capítulo de previsão dos dois modelos usados na análise dos prestadores de saúde.

Parte I: Apresentação da empresa

A Quidgest é uma empresa tecnológica multinacional especializada em *software*, criada em 1988 e com origem em Portugal, desenvolve sistemas de informação adequados a cada cliente. Entre outros prémios e distinções é PME Líder, PME Excelência, integra a rede de PME inovadoras da COTEC e é reconhecida internacionalmente com o Nível 3 do CMMI.

A maior vantagem da Quidgest é a geração automática de *software* através da plataforma GENIO.

O GENIO modela e gera automaticamente sistemas de informação integrados em diferentes arquiteturas e linguagens. Produz sistemas diferenciadores e únicos que permitem otimizar recursos e aumentar a produtividade, atrair competências e investimento, com capacidade de adaptação a contextos diferentes, de lidar com o inesperado ou de evoluir rapidamente. O GENIO obtém soluções para sistemas obsoletos e diferencia-se pela produção de milhares de linhas de código por segundo (cerca de 98% de código é gerado automaticamente). Esta fonte de vantagens competitivas permite uma diminuição dos prazos de entrega, geração rápida de protótipos funcionais, redução nos custos de manutenção, partilha de uma linguagem comum de “programação” dentro da empresa e ainda flexibilização na criação de sistemas.

As organizações ou instituições clientes da Quidgest são acompanhadas e o desenvolvimento de *software* é complementado pelos de serviços de consultoria de negócios, definição de procedimentos, formação e pela manutenção técnica evolutiva de sistemas de informação.

Está presente em diversas áreas de negócio, entre as quais: Administração Pública e Regulação, Apoio ao desenvolvimento, Aprovisionamento, Logística, Distribuição e Gestão de Ativos, Banca, Seguros e Gestão Financeira, Gestão Documental, Cidadania e BPM, Engenharia, Construção, Obras Públicas, Energia e

Ambiente, I&D, Tecnologia e *Outsourcing*, Recursos Humanos e Formação, Privacidade Digital e Cibersegurança, Qualidade e Gestão Estratégica, *Sensing* e *Internet of people*, Saúde, Desporto e Lazer.

A Quidgest criou três plataformas de modo a controlar o funcionamento da empresa, dos projetos com os clientes e das funções dos seus colaboradores, sendo eles, o "Platina", que presta o apoio ao cliente, o "BSC Quidgest", que permite avaliar o nível de sucesso dos objetivos impostos pela empresa e o "Quigenio" que é o sistema integrado de gestão utilizado para recolher o máximo de informação sobre os colaboradores.

O Platina foi concebido para perceber a satisfação dos clientes de modo a aumentar a eficácia do desempenho inovador que a Quidgest cria. Através de uma plataforma *online*, os clientes reportam os erros e incidentes que ocorrem e esses são reencaminhados para o respetivo departamento para garantir um nível de serviço mais rápido e eficaz e para que exista transparência de processos e comunicação entre ambas as partes. Assim, potencializa-se a evolução e a qualidade dos sistemas e permite a criação de valor acrescentado para os clientes.

O *Balanced Scorecard* (BSC) é uma ferramenta que mede o desempenho e visa identificar e melhorar várias funções internas de uma empresa e melhorá-las. A Quidgest criou o "BSC Quidgest" que foi instituído com o propósito de operacionalizar a gestão estratégica e focar-se no que é fundamental para a empresa, como visualizar rapidamente os resultados e trabalhar colaborativamente com toda a equipa alinhada e motivada em prol dos objetivos comuns, sem omitir a necessidade de implementar as atividades necessárias à correção de desvios. Esta plataforma permite mobilizar os recursos mais adequados e evitar desperdícios e custos na circulação de informação.

O BSC Quidgest apresenta quatro focos: a visão, a gestão, as pessoas e a tecnologia. É necessário a comunicação da estratégia da empresa; a nível da gestão, o objetivo passa por ter as responsabilidades definidas e todos os níveis envolvidos com uma sequência organizacional definida por um mapa estratégico; a nível de pessoas,

deve haver controlo em tempo real dos resultados obtidos pelos colaboradores e através dos resultados individuais perceber a produtividade que traduz a evolução da estratégia implementada; a nível da tecnologia, o BSC é de fácil implementação e utilização, escalável e elevada capacidade de integração.

O Quigenio é um sistema integrado de gestão que foi desenvolvido pela Quidgest e é utilizado desde a criação da Quidgest e por todos os membros da empresa. Tem como objetivo o preenchimento de pequenos formulários de forma a que os dados, as ações, os projetos e os incidentes fiquem todos registados e sejam utilizados para fins estatísticos da empresa.

Assim, percorrendo o Quigenio, todos os colaboradores têm que registar as suas ações no separador "Ações", talvez uma das funcionalidades mais importantes do sistema na medida que todo o horário de trabalho do colaborador tem que ficar registado e essas ações têm um ou vários projetos em que o colaborador está a inserido.

No separador "Incidentes" é possível registar todos os incidentes de um colaborador como, por exemplo, o esquecimento do cartão de acesso à empresa. O separador dos "Projetos" permite criar o projeto e ver quais os colaboradores alocados.

Relativamente ao separador "Gestão", é possível consultar os contactos e os dados dos colaboradores, isto é, se for necessário o contacto entre colaboradores, existe sempre o acesso ao *email* pessoal da Quidgest. Ainda assim, o Quigenio oferece muitas outras funcionalidades.

As três plataformas de gestão acima descritas requerem formação por parte de todos os colaboradores para uma eficiente administração.

Parte II: Enquadramento teórico

A história da *big data* está intrinsecamente ligada à *data science*. O conceito de *big data* foi utilizado pela primeira vez por Michael Cox and David Ellsworth, em 1997.

Visualization provides an interesting challenge for computer systems: data sets are generally quite large, taxing the capacities of main memory, local disk, and even remote disk. We call this the problem of big data. When data sets do not fit in main memory (in core), or when they do not fit even on local disk, the most common solution is to acquire more resources. Cox, M & Ellsworth, D.

Assim, o conceito de *big data* refere-se ao crescente volume de dados estruturados e não estruturados que são criados e recolhidos através das tecnologias e dos sistemas de informação. O armazenamento dos dados, segundo Inmon (2000), suporta o processo informacional e estabelece a facilidade para a integração num mundo desintegrado de sistemas de aplicação. Também organiza e armazena os dados necessários para o processo informacional e analítico ao longo do tempo.

O processo histórico das tecnologias e dos sistemas de informação lembram o surgimento dos conceitos de *business analytics* e *business intelligence* que, segundo Negash (2004), fazem parte da solução que ajuda a organização a tomarem decisões baseadas nos dados que recolhem, fornecendo análises simplificadas e visuais que, de outra forma, seriam extremamente difíceis de compreender.

O aumento de dados despertou a oportunidade de expandir o *business intelligence*. O *business analytics* tem acompanhado o processo que segundo Duan e Cao (2015), a sua emergência deve-se aos desafios e às oportunidades do conceito de *big data*. A ligação entre *business intelligence* e *business analytics* centra-se no tratamento dos dados em informação com a utilização da tecnologia e a estatística.

O *business intelligence* pode ser abordado a nível da estrutura, conteúdo e estado de uma base dados quando essa é manipulada e alterada através de processos de código como, por exemplo, *stored procedures* e *SQL*. A construção e manipulação

dos dados que foram fornecidos são considerados uma interpelação de *business intelligence* na medida que pode ser compreendido em quatro pontos: o *reporting*, a análise, a monitorização e a análise preditiva.

O *business analytics* refere-se a técnicas e processos qualitativos e quantitativos usados para ajudar as organizações a planear e a perceber a *performance* dos negócios baseado nos dados. O *business analytics* apresenta algumas vantagens. Segundo Duan e Cao (2015), um dos principais objetivos é tornar a informação transparente, os serviços mais precisos e também perceber qual nível de serviços que se seguem através da identificação de tendências e determinação de padrões em grande volume de dados. Isto pode ser a chave para deduzir, por exemplo, eventuais *outliers*.

O *business intelligence* e o *business analytics* partilham uma relação mútua. O *business analytics* concentra-se nas reflexões que possam ser benéficas para o planeamento estratégico, resultando em vantagens competitivas que, no caso deste trabalho, permite perceber qual a distribuição geográfica dos prestadores de saúde, quais as especialidades mais relevantes e quais as lacunas geográficas na prestação de serviços de saúde,

Em várias organizações, a tecnologia de informação tem sido o estímulo para o processo de inovação, em que é possível transformar grande quantidade de valores brutos em informações valiosas.

Assim sendo, uma vez que os dados são processados em informação o conceito de sistemas de informação emerge. Um sistema de informação é uma combinação organizada de pessoas, *hardware*, *software*, redes de comunicação e dados que são colhidos, transformados e divulgados como informação com um propósito específico.

Logo, há necessidade de integrar as ferramentas de *business intelligence* e *business analytics* com os sistemas de gestão de base dados (SGBD).

O sistema de gestão de base dados é um *software* que gere o armazenamento, a atualização e recuperação de uma base dados. Segundo Silberschatz, Korth e

Sudarshan (2009), é caracterizado por um conjunto de dados interrelacionados, por um conjunto de programas que acedem aos dados e por ambiente que seja eficiente e conveniente de usar.

Os utilizadores dos sistemas têm recursos para executar vários tipos de operações. Geralmente, um SGBD manipula os dados em si, isto é, o formato dos dados, nome dos campos e a estrutura dos registos. Um SGBD pode ser representado em três níveis de arquitetura, o nível físico em que os ficheiros são guardados em suportes de armazenamento informático e, a partir daí são manipulados pelo SGBD. O nível conceptual em que a informação é organizada em tabelas e relacionamentos e por último, o nível de visualização que corresponde à forma como os dados são representados aos utilizadores finais, através de gráficos e tabelas. Um exemplo de sistema de gestão de base de dados é o *SQL Server*.

As ferramentas utilizadas neste trabalho permitem uma fácil consulta dos dados, através de *queries*, onde os mesmos são tratados e transformados em informação através de diferentes metodologias, tal como o *R* - que permite a manipulação e visualização do conteúdo da base de dados. E, assim, surgem os conceitos de *data manipulation* e *data visualization* ligados ao *business intelligence* e *business analytics*.

Se por um lado existe um grande volume de dados, esses podem ser alterados para facilitar o processo de leitura e para ficarem melhor organizados. Nesse sentido, uma das ferramentas utilizadas para o desenvolvimento deste trabalho, o *SQL Server*, segundo Batra (2018), permite através do seu código próprio a manipulação dos dados como, por exemplo, a inserção (*INSERT*), a atualização (*UPDATE*), a eliminação (*DELETE*), a junção (*JOIN*) e toda a outra linguagem que restringe os dados de várias tabelas de uma base de dados em análise. O *SQL Server* admite a integração de outras linguagens na sua habilidade de transformação dos dados, como o *R*. Quando os dados estão processados, podem ser apresentados em diversos formatos visuais, como texto, gráficos ou tabelas. Segundo Reiner e Potter (2007), o conceito de *data*

visualization faz com que a leitura dos dados seja mais atrativa e perceptível para os usuários e que facilita a tomada de decisão e, neste trabalho, está inerentemente ligado aos *packages* que o *R* oferece.

Atualmente, é quase impossível imaginar um serviço de cuidados de saúde sem que este utilize um sistema computadorizado de informação pelas inúmeras vantagens que podem trazer, tendo como primeiro objetivo apoiar os funcionários nas suas funções e, para além disso, oferecer um melhor serviço ao utente (Ammenwerth 2003).

Segundo Ammenwerth (2003), neste caso, os Sistemas de Informação na saúde facilitam o processo que reúne, guarda, processa e faculta a informação de uma organização de saúde. É uma combinação de procedimentos, pessoas, tecnologias e informação com o objetivo de desenvolver e proteger a saúde da população. É por tal motivo que, quando nos referimos a um sistema de informação, não nos referimos apenas às tecnologias da informação, já que deveremos considerar também outros fatores que são igualmente importantes, nomeadamente os utilizadores.

A análise das variáveis deste trabalho contribuem para tentar perceber a localização da prestação dos serviços de saúde e se os atos médicos são geridos de forma eficiente.

A resolução deste trabalho foi composta por diversas metodologias e ferramentas que se complementaram e permitiram a análise dos dados, tais como o *R*, o *SQL Server Management Studio (SMSS)*, o *SQL Server Reporting Services (SSRS)*, o *stored procedure*, o *Microsoft Visual Studio (MVS)* e também a integração no GENIO.

1.1 R

O *R* foi criado por *Ross Ihaka* e *Robert Gentleman* há mais de 20 anos na Universidade de *Auckland*, na Nova Zelândia, segundo Ihaka e Gentleman (1996). O *R* surgiu da combinação com o *S* – é o principal veículo para a pesquisa em metodologia estatística.

O *R* é uma linguagem e um ambiente para dados estatísticos e gráficos (R-project, 2018). O *R* proporciona uma larga variedade estatística desde modelos lineares e não-lineares, testes estatísticos, análise de séries temporais e é altamente extensível. O ambiente *R* é um conjunto integrado de recursos de *software* para manipulação de dados, cálculo e exibição gráfica que inclui uma eficaz manipulação e armazenamento de dados; uma vasta e coerente coleção de ferramentas integradas para análise de dados; uma linguagem de programação bem desenvolvida, simples e eficiente que inclui condições, *loops* e funções definidas pelo utilizador. O *R* pode ser estendido através de *packages* e também complementado por outros programas.

1.2 SQL

Em 1992, a *Microsoft* assumiu a responsabilidade maior sobre o futuro do SQL Server para o NT. No ano seguinte, a *Microsoft* torna-se o segundo maior vendedor de software ligado ao sistema de gestão de base de dados. No entanto, em 1994 a parceria entre a *Microsoft* e *Sybase* é dada como terminada. Nos anos seguintes a *Microsoft* lançou diversas versões do SQL Server até à atual, o SQL Server 2019 e tem vindo a facilitar a execução de aplicações mais exigentes.

O *Structured Query Language (SQL)* é uma linguagem de programação que permite a manipulação, controlo, transação e consultas de dados. Deste modo tem como função ser a interface entre o utilizador e a gestão da base de dados.

1.2.1 SQL Server Management Studio

O *SQL Server Management Studio (SSMS)* é um módulo que é usado para configurar, gerir e executar todas as componentes do *Microsoft SQL Server*. O *SSMS* inclui editores de script e ferramentas gráficas que trabalha com objetos e características do servidor. A principal característica do *SSMS* é o *object explorer* que permite ao utilizador pesquisar, selecionar e agir sobre qualquer um dos objetos dentro do servidor. O *SQL Server Management Studio* permite através das suas

funcionalidades criar objetos como os *stored procedures* que permitem incorporar a linguagem *R* e fazer a criação de relatórios no *Microsoft Visual Studio* para ser possível a escolha dos parâmetros criados nos *stored procedures* o *reporting* dos dados através do *SQL Server Reporting Services* para um posterior reporte gráfico.

1.2.2 SQL Server Reporting Services

O *SQL Server Reporting Services* é um módulo do *SQL* responsável pela construção de relatórios. É permitido fazer ações desde a construção de tabelas, a relatórios que são expansíveis e permitem exibir categorias e subcategorias da informação analisada até apresentar dados de uma forma gráfica. Neste trabalho, foi utilizado de forma a que fosse possível a integração de todos os relatórios no GENIO.

1.2.2.1 Stored procedure

O *stored procedure* é uma sub-rotina do *SQL Server* que permite ser salva para que o código seja utilizado repetidamente. Para fazer uma consulta que é necessária várias vezes de um procedimento extenso e que requer a execução de várias instruções *SQL*, para economizar tempo e memória, pode ser salvo em procedimentos armazenados e sempre que é chamada, o código é executado (Stored procedures, 2018).

Todos os *scripts* dos *stored procedures* presentes neste trabalho declaram um *input*, definem um código que seja maior que zero, sendo o resultado positivo, o *output* é dado em forma de gráfico ou tabela. Caso contrário, se o resultado for menor que um, o gráfico apresenta uma mensagem "Não existem dados para os parâmetros selecionados".

1.2.3 Microsoft Visual Studio

O *Microsoft Visual Studio (MVS)* é um ambiente de desenvolvimento integrado da *Microsoft* que, neste caso, permite a criação de relatórios para consulta de dados. O *MVS* não suporta nenhuma linguagem de programação, a ligação entre o *SSMS* e o *MVS* é feita através dos *stored procedures* criados *à priori*.

1.4 Integração no GENIO

O projeto SVD teve como objetivo a integração dos *reportings* feitos no SSRS. A integração foi feita da seguinte forma, no *Menu* existe um separador denominado por *Reports* onde são colocados os diversos *reports* em submenus e dentro desses submenus irão aparecer os *reports* produzidos numa página. É alterado o código de *html* de cada menu para abrir uma janela para consulta dos dados. O código a ser alterado é: `<iframe src = "link" width = "" height = ""></iframe>`.

1.5 Composição de uma série temporal

Uma série temporal é uma sequência de observações ao longo do tempo. Neste trabalho final de mestrado, foi obtida uma série temporal que rastreia os dados reais dos atos médicos registados sobre os prestadores de saúde, isto é, o rácio das diferentes especialidades. O principal objetivo é tornar os dados da especialidade de medicina geral e familiar do distrito do Porto numa série temporal, decompô-la e analisar a previsão dos 24 meses seguintes à última data.

Segundo o modelo clássico as séries temporais são compostas por quatro padrões:

- a tendência, que é o comportamento de longo prazo da série e que caracteriza o andamento mais notório da série;
- os movimentos oscilatórios, flutuações nos valores da variável que se repetem com certa periodicidade e podem ser resultado de variações da economia, como períodos de crescimento ou recessão;
- a sazonalidade, oscilações periódicas nos valores da variável com duração inferior a um ano, e que se repetem todos os anos, geralmente em função das estações do ano;
- os movimentos irregulares ou aleatórios, que são as flutuações inexplicáveis resultantes de fatos inesperados como catástrofes naturais, atentados terroristas, decisões intempestivas de governos, etc.

A previsão dos valores futuros de uma sucessão cronológica representa um dos objetivos mais importantes da análise de séries temporais e pode realizar-se através de diferentes metodologias, tendo em conta o tipo de utilização, a extensão (curto, médio e longo termo) e, principalmente, a disponibilidade dos dados.

Dada uma série temporal observada até ao momento t , y_1, y_2, \dots, y_n , $i=1, \dots, n$, pretende-se prever o valor a observar no momento $t+m$, Y_{t+m} (variável aleatória), com base na série temporal, sendo $m \geq 1$.

Designa-se por $Y_{t+m} = X_t(m)$ o preditor de Y_{t+m} com origem em t (t é a origem da previsão e m o horizonte da mesma). Assim $Y_t(m)$ diz-se função dos valores observados da série temporal: $Y_t(m) = f(y_1, y_2, \dots)$.

Para se escolher um preditor de modo que a sua distância à origem t seja mínima, utiliza-se o método dos mínimos quadrados. O critério empregue para medir a distância entre Y_{t+m} e $Y_t(m)$ é o erro quadrático médio, $E[(Y_{t+m} - Y_t(m))^2]$, "e o problema da procura do "melhor" preditor reduz-se, deste modo, a determinar a função de previsão que minimiza esse erro", segundo *Turkman (1993)*. O melhor preditor é dado por $Y_t(m) = E[Y_{t+m} | y_1, y_2, \dots, y_n]$ em termos das observações y_1, y_2, \dots, y_n .

Para a especialidade a prever foram escolhidos dois modelos, o modelo de Holt-Winters e o modelo Alisamento Exponencial, cada modelo é calculado automaticamente conforme a fórmula R sendo que a parametrização é escolhida à priori. O período base em análise é entre 2010 e 2016 e a previsão é feita a 24 meses.

Quaisquer ferramentas e metodologias utilizadas para observar e compreender a informação disponível possibilitam que sejam tomadas decisões pelas organizações de forma a melhorar os resultados obtidos pela análise e no caso deste estudo perceber quais são e que comportamentos os prestadores de saúde poderão ter a longo prazo.

Parte III: Análise empírica

A parte da análise empírica engloba o conteúdo descritivo do relatório de estágio. Explica os objetivos deste trabalho, descreve os dados e as variáveis e elucida as limitações da análise.

Para a facilitar a interpretação dos resultados, está presente nesta parte a explicação sobre a integração do *R* com o *SQL Server* e por fim a caracterização de uma série temporal, onde são aplicados os *reports* desenvolvidos.

1. Objetivos

O estudo baseia-se na exploração de uma base de dados que é composta por registos de atos médicos que são classificados por especialidade e registados pelas entidades prestadoras de serviços de saúde distribuídas pelas respetivas freguesias, concelhos e distritos. O objetivo principal é identificar as especialidades com maior peso nas entidades prestadoras de serviços de saúde, caracterizar o seu comportamento ao longo do tempo e perceber a sua relevância em diferentes distritos. Toda esta informação será tratada e reportada através da integração do *R* com o *SQL Server Reporting Services*.

2. Dados e descrição das variáveis

A base de dados é composta por 2.771.863 atos médicos registados pelos prestadores de serviços de saúde. Os prestadores de saúde são no total 715 entidades que estão divididas por distritos, conforme explicitado na tabela 1.

Aveiro	Beja	Braga	Bragança	Castelo Branco	Coimbra	Évora	Faro	Guarda
41	2	57	11	6	97	8	10	11
Leiria	Lisboa	Portalegre	Porto	Santarém	Setúbal	Viana do Castelo	Vila Real	Viseu
22	147	7	188	20	37	15	18	22

Tabela 1 - Nº prestadores de saúde

Sabe-se que os prestadores se localizam em maior número no Porto, que regista 188, seguido pelo distrito de Lisboa, que conta com 147 e por Coimbra com 97 prestadores. Os prestadores têm nos seus serviços diferentes tipos de especialidade de modo a prestar o maior número de serviços e observar o maior número de doentes. Os serviços estão agrupados por diversas especialidades.

Como existem várias limitações na base de dados, procedeu-se à seleção das especialidades com a informação necessária para poder realizar uma análise mais aprofundada e perceber o comportamento que os atos médicos apresentam ao longo do tempo. Dos dados iniciais apenas os dados das especialidades de medicina geral e familiar, estomatologia e oftalmologia farão parte da análise nos distritos mencionados acima.

Como é possível reparar na tabela 2, sobre a decomposição dos dados em que a análise se baseia, nota-se que medicina geral e familiar juntamente com estomatologia representam mais de 80% dos atos médicos registados pelos prestadores de saúde enquanto a especialidade de oftalmologia representa 3%.

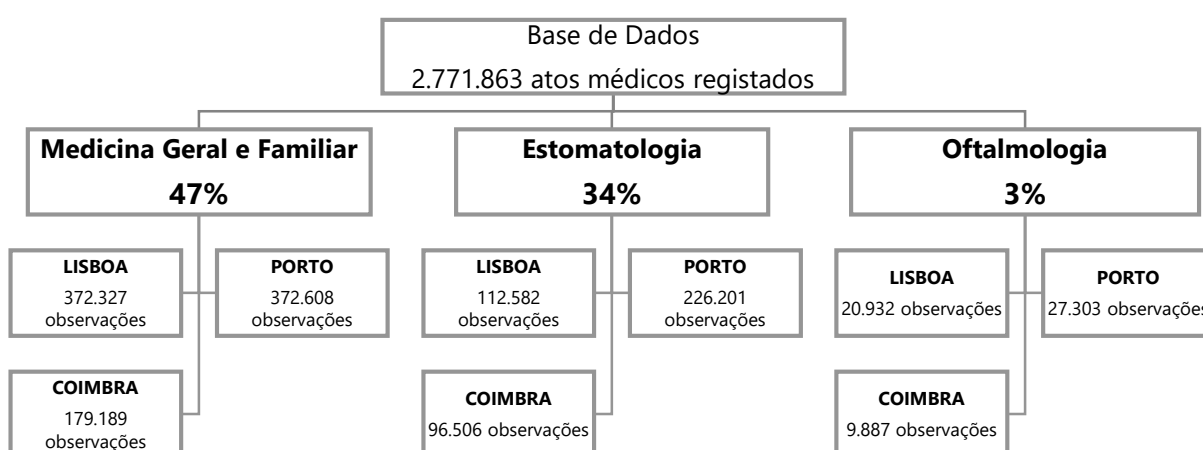


Tabela II - Decomposição da base de dados

A especialidade de medicina geral e familiar será analisada por um período de seis anos, em dois modelos distintos de previsão, sendo eles, o modelo de *Holt-Winters* e o Alisamento Exponencial.

A informação da base de dados pode ser dividida por variáveis que dão origem ao estudo do seu comportamento. Assim, o objetivo inicial foi tratar toda essa informação e reportá-la. A variável mais importante é o ato médico que é registado nos sistemas dos prestadores de saúde. Cada ato médico é composto pela data do registo, pelo código de prestador que fez esse mesmo registo, pela respetiva descrição e pela especialidade ao qual pertence. O código do prestador dá a informação do distrito, do concelho e também da freguesia onde estão localizados.

A análise começa por compreender a distribuição geográfica dos atos médicos registados através da representação por concelhos de Portugal continental, ou seja, a parametrização do *report* "Concelhos" permite escolher uma data inicial e uma data final a analisar e também o distrito e a especialidade. O objetivo é perceber a dispersão entre Lisboa, Porto e Coimbra e em cada uma das especialidades referidas.

Dos concelhos pertencentes a Lisboa, em estomatologia, Odivelas é o que apresenta maior número de atos em todo o ano de 2016, seguido de Sintra e Vila Franca de Xira com valores superiores a mil atos de saúde prestados. Já no Porto, e sobre a mesma especialidade e no mesmo período, o concelho com maior expressão é o de Penafiel com registos superiores a quatro mil atos. No entanto, Gondomar, Vila Nova de Gaia e Porto superam os dois mil atos. Em Coimbra, dentro dos mesmos parâmetros, Lousã e Coimbra são os concelhos mais expressivos, assemelhando-se ao registo dos concelhos do distrito do Porto com mais de quatro mil atos registados.

Sobre a especialidade de oftalmologia, apenas Lisboa apresenta relevância relativamente aos outros concelhos, com mais de três mil atos registados. No distrito do Porto, o concelho de Penafiel registou novamente mais de dois mil atos registados pelos prestadores de saúde, seguido do concelho do Porto com cerca de 1418 atos registados nesta especialidade. Passando para o distrito de Coimbra, apenas o próprio

concelho de Coimbra tem importância e o número de atos registados ficam abaixo dos anteriormente analisados, com apenas 923 atos médicos registados.

Relativamente a medicina geral e familiar, esta é uma especialidade bastante importante na análise dos registos médicos, apresenta valores muito superiores aos anteriores e tem um peso idêntico nos vários distritos. Em Lisboa, o concelho de Oeiras representa quase vinte mil atos enquanto o próprio concelho do Porto ultrapassa os trinta e um mil atos registados. Em Coimbra, apenas o concelho de Coimbra apresentou registos de atos médicos na especialidade de medicina geral e familiar ultrapassando os vinte e dois mil atos.

A consulta do número de prestadores de saúde por distrito é possível ser analisada dado uma data inicial e uma data final do período pretendido. Cada especialidade pode ser consultada por tipo de atos médicos e o número total de vezes que esses atos foram registados.

A análise pode ser complementada com o relatório do "Registo Mensal" que apresenta o número de atos que são registados mensalmente durante um ano. Neste caso é fornecida a média anual da especialidade e do distrito pretendido e tem por base o ano de 2016. Os distritos apresentam registos mensais diferentes tal como acontece com as especialidades supramencionadas. Através do relatório "Registo Mensal" é possível analisar todas as flutuações dos atos médicos. Desta forma, medicina geral e familiar apresenta uma média anual acima dos três mil atos médicos registados. No Porto, a média dos registos ultrapassa os cinco mil registos. No entanto, em Coimbra, a média encontra-se abaixo dos dois mil atos médicos registados. Relativamente a estomatologia, a média anual está perto dos 750 registos. No Porto, a média é representada com quase mil e quinhentos registos e em Coimbra, ao revelar uma menor expressão ultrapassa os 900 registos anuais. A especialidade de oftalmologia apresenta oscilações muito idênticas à de estomatologia, apenas o número de registos é menor e apresenta uma média um pouco acima dos 250 registos.

No distrito de Lisboa, os registos mensais de atos médicos, apresentaram uma semelhança nas diferentes especialidades mencionadas, dado que nos meses de verão, principalmente no mês de agosto, nota-se uma quebra do número de registos com valores abaixo da média anual enquanto os meses de verão apresentaram diferentes flutuações entre as especialidades. Em estomatologia, existe um grande crescimento do número de atos registados entre agosto e outubro sendo que nos meses seguintes, novembro e dezembro, o número de atos médicos fica mesmo abaixo dos 500 registos. No entanto, os meses de fevereiro até junho apresentaram valores constantes. Relativamente à medicina geral e familiar, os meses de janeiro a maio e também os meses de setembro, outubro e novembro apresentaram registos acima da média anual, que está fixada nos entre 3.500 e os 4.000 registos. Em relação à especialidade de oftalmologia, esta apresenta oscilações muito idênticas à de medicina geral e familiar.

No distrito do Porto, a especialidade de estomatologia revela algumas quebras ao longo do ano, sendo o mês menos expressivo o de dezembro, mas, no entanto, agosto apresentou também um baixo registo de atos médicos. Em relação a medicina geral e familiar, os meses iniciais de 2016 registaram um número de atos médicos superiores à média anual e o mês de agosto regista novamente uma grande quebra nos atos médicos registados, tal como no mês de dezembro. A especialidade de oftalmologia é que apresenta, em 2016 e no Porto, a maior discrepância entre meses ao longo do ano dado que os seus valores se apresentam constantemente em picos, com resalto para o decréscimo no mês de agosto.

Coimbra é o distrito com menos atos médicos registados, facto que se explica pela quantidade de prestadores de saúde que lá estão convencionados. Contudo, a semelhança aos outros distritos mantém-se. Estomatologia apresenta algumas oscilações de crescimento e quebras e o mês de agosto não foge à regra.

A medicina geral e familiar apresenta fevereiro como o mês com menos registos de atos médicos seguido de meses de algum crescimento, aparecendo agosto com

um decréscimo e posteriormente, um aumento dos registos dos atos médicos até ao mês de dezembro, em que existe uma grande diminuição. A especialidade de oftalmologia, ao contrário de estomatologia e medicina geral e familiar, apresenta um menor número de registos no total, ao invés dos primeiros meses de 2016 que apresentaram uma descida constante até março, tal como nos meses de junho, julho e agosto. O mês de novembro apresentou o maior número de registos de atos médicos pelos prestadores de saúde.

Como alguns meses apresentam um comportamento muito idêntico nas diferentes especialidades e distritos, fazer uma análise mais centrada nesses meses é possível através do relatório "Meses" que reporta o comportamento de todos os meses do ano por um período em escolha e perceber a evolução ao longo dos anos. Por exemplo, agosto é um dos meses que apresenta sempre um baixo registo em número de atos médicos. Analisando entre 2010 e 2016, medicina geral e familiar é, a especialidade mais representativa da base de dados, no Porto existe uma crescente evolução do mês de agosto, no entanto, mostra-se com alguns picos e a média destes seis anos apenas fica um pouco acima dos 600 registos. Todavia, janeiro apresenta valores entre 800 e os 1900 registos, sendo a sua média mais do dobro do que foi registado em agosto.

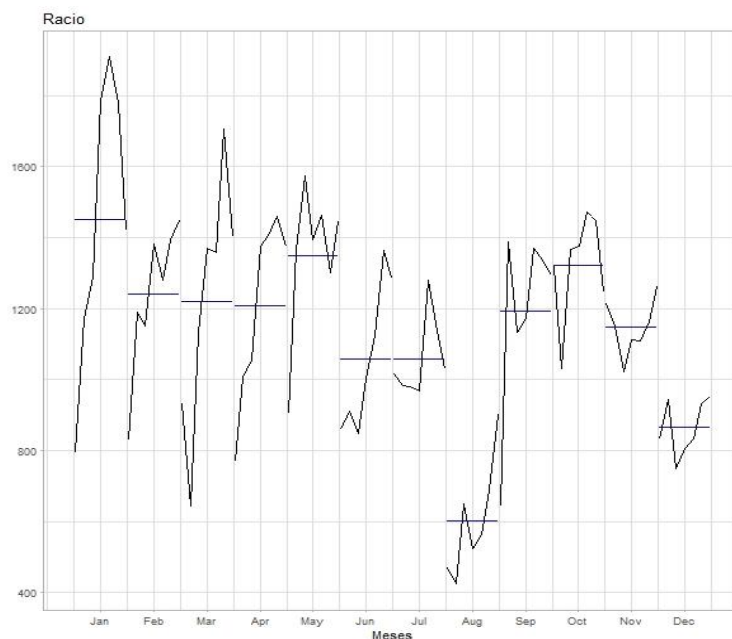


Figura 1 – Médias mensais do relatório "Meses"

A diversidade de cada especialidade é de tal forma extensa que foi criado um relatório "Descrição" para ser possível a consulta e análise descritiva do ato médico.

A análise do comportamento dos atos médicos é feita pelo rácio que explica a relação entre o número de atos médicos que são registados ao longo de anos com o número de prestadores que registam esses mesmos atos. O resultado do rácio é dado pelo "Total" de prestadores de cada mês e do respetivo ano e pela "Quantidade" de atos médicos registados. O rácio é então o número de atos médicos sobre o número de prestadores.

$$\text{rácio} = \frac{\text{número de atos médicos}}{\text{número de prestadores de saúde}}$$

Toda a análise é baseada no cálculo do rácio, ou seja, o rácio de cada especialidade relativo a cada distrito analisado com base na comparação com diferentes anos. Este relatório serve para uma análise suplementar onde é possível perceber as oscilações entre dois anos diferentes.

Os relatórios que foram feitos posteriormente fazem a decomposição desse mesmo rácio numa série temporal de modo a analisar a sua sazonalidade, tendência e a componente errática e, por último, a análise da previsão em 24 meses das especialidades e distritos referidos acima, através do modelo mais adequado de previsão.

O relatório denominado por "Série Temporal" apresenta o registo temporal do rácio entre o número de atos médicos registados e número de prestadores que prestam esses mesmos serviços. É possível observar o rácio de cada ano ou de vários anos.

Os relatórios que dizem respeito à apresentação dos gráficos da previsão está discriminada e sumariada num só capítulo.

3. Limitações à análise

Integridade dos dados:

Os dados para a realização da análise são inseridos pelos colaboradores dos prestadores de saúde e por isso apresentam algumas falhas. Existem registos de atos médicos incompletos, datas incorretas e nulas e a especialidade é repetida muitas vezes dado que os prestadores de serviços de saúde são compostos por várias especialidades o que torna difícil a triagem dos dados. Contudo, a remoção desses dados para a análise afetam a amostra para análise.

Tratamento dos dados:

Ao analisar e consultar várias colunas de diferentes tabelas a junção dos dados elimina ou não lê muitos dos dados incompletos e valores nulos. As variáveis iniciais, ao serem aplicadas no ambiente *R* e na linguagem de programação *SQL Server*, transformam-se noutra tipo de objetos como, por exemplo, em *data frame* ou vetores.

Inserção de dados:

Para um tratamento e uma melhor exploração dos dados foi necessário inserir dados em tabelas novas para facilitar a consulta e a visualização dos mesmos. As tabelas criadas estão relacionadas com dados geográficos, onde inseri as coordenadas geográficas de cada distrito e respetivos concelhos existentes.

A integração do R com o SQL Server

Segundo Kastrun e Koesmarno (2018), a linguagem R está disponível desde os anos 90 e ao longo dos anos foi ganhando popularidade devido à sua forma de livre acesso e à sua enorme extensão de *packages* que foram desenvolvidos pela comunidade que utiliza o R tornando-o numa linguagem versátil, que permite análises estatísticas robustas e com capacidades preditivas de modelação. Enquanto isso, o SQL Server introduziu através do SQL Server Analysis Services (SSAS), a análise e o recurso ao *data mining* que, com o passar do tempo, os utilizadores do SQL Server mostravam-se mais interessados em perceber as demais abordagens e métodos estatísticos.

A análise dos prestadores de saúde é elaborada através da integração do R com as funcionalidades do SQL Server, isto é, o R permite a análise avançada de grandes quantidades de dados armazenados em base de dados. Neste caso, as análises preditivas e o *machine learning*, que consiste num campo da inteligência artificial e que tem a capacidade dos programas computacionais se adaptarem a novos dados sem a intervenção humana, pode ser complementado entre várias linguagens, como por exemplo, o SQL Server.

Na prática, a ideia de *machine learning* passa pela construção de um código de programação ou um algoritmo que identifica os dados e usa parâmetros construídos no código para formar padrões para a tomada de decisões, caso os dados sejam atualizados, o código adapta-se aos parâmetros (*Machine Learning*, 2018). Na figura 2 é possível perceber o processo de integração entre o R e o SQL Server.

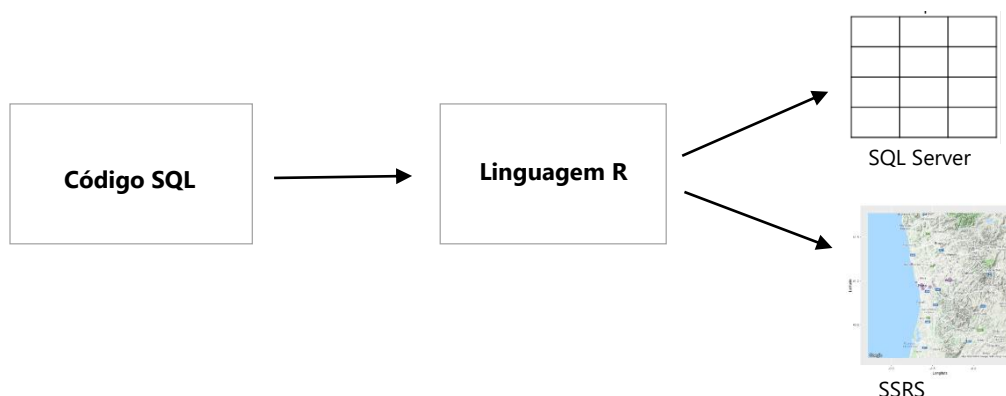


Figura 2 - Processo de integração do SQL Server com o R

A integração do *R* com o *SQL Server* através do *package* do *R* – o *RODBC* e permite ter flexibilidade de análise estatística.

O *package* *RODBC* possibilita fazer a conexão entre a base de dados do *SQL Server* e o *R* introduzindo o *uid* – identificação e *pwd* - *password*, quando requerida, para fazer a autenticação e a respetiva *query* para que o *SQL Server* envie os dados para o *R*. No entanto, para que os *scripts* do *R* sejam utilizados no *SQL Server* é necessário configurá-lo e, portanto, ativar os *scripts* externos pela instrução "sp_configure ' external scripts enabled', 1;".

Depois dos *scripts* ativos procede-se à criação, que engloba:

- *@language* = N ' *language* ' – indica a linguagem pretendida;
- *@script* = N ' *script* ' – é representado por todo o *script* *R* e deve ser *nvarchar* (max), o valor padrão para introdução dos dados é *InputDataSet*, e o resultado é dado pelo *OutputDataSet*;
- *@input_data_1* = N ' *input_data_1* ' – especifica os dados usados pelo *script* externo para a consulta dos dados, neste caso o *R*;
- *@parallel* = 0 | 1, assume-se que *@parallel* = 1 para ativar a execução de *scripts* do *R*;
- *@params* = N ' *@parameter_name data_type* ' – é a declaração da lista de parâmetros de entrada que são usados no *script* *R*. Como *@parallel* = 1 é necessário especificar uma cláusula e um esquema de saída como é possível ver em anexo nos distintos *stored procedures*.

No *Microsoft Visual Studio*, a base de dados é inserida e são criados os *reports* com base nos *stored procedures* do *SQL Server Management Studio*, de seguida são formulados os *datasets* com os gráficos e tabelas necessárias à análise dos dados. Os parâmetros são definidos e, posteriormente, é feito a construção do *template* do relatório.

Em anexo, encontram-se os *stored procedures* criados e a descrição dos *scripts*.

Análise de Resultados

Relativamente aos resultados, a figura 27, “NrPrestadoresDist”, presente nos anexos, representa o número de prestadores por distrito, no ano de 2016. A maioria dos prestadores de saúde convencionados encontram-se no litoral, sobretudo no Porto e Lisboa, contrastando com os números de Évora, por exemplo.

Na figura 28, o relatório apresenta o número de atos médicos por cada concelho. Os concelhos que compõem os diferentes distritos apresentam densidade populacional discrepante, no entanto, o número de atos médicos registados depende do número de prestadores de saúde convencionados nessas zonas. A diferença entre Abrantes e Cascais apenas está relacionada com o número de atos médicos que foram registados, uma vez que o número de prestadores de saúde é igual (5).

O relatório “Concelhos”, figura 29, permite parametrizar o distrito a analisar, de modo, a serem só apresentados os seus concelhos. O próprio concelho de Braga apresenta um número de registos muito superior aos restantes concelhos, isso pode ser explicado pelos prestadores convencionados (19).

A figura 30, do relatório “Registo Mensal” permite ter uma visão anual da especialidade de Medicina Geral e Familiar no Porto e o comportamento representado vai de encontro às conclusões já anteriormente tiradas. Os meses de inverno comprovam um maior número de registos ao contrário dos meses de verão.

O relatório “Descrição”, figura 31, foi pensado para listar todos os atos médicos que tenham sido registados em cada especialidade e, apesar deste relatório apresentar diversas páginas, em Medicina Geral e Familiar, o ato médico registado o maior número de vezes é “Clínica Geral – Consulta Normal” (177.265 atos).

Na figura 32, o “Rácio”, possibilita rapidamente entender que, no geral, o número de atos médicos entre 2010 e 2016 subiu substancialmente, contudo, o comportamento ao longo do ano é bastante homogéneo em ambos os anos.

Caracterização de uma série temporal sobre o rácio

A figura 3 apresenta a decomposição da série temporal de medicina geral e familiar, do distrito do Porto, sobre o rácio entre o número de atos médicos e número de prestadores de saúde sendo que o período em análise é de 2010 a 2016. A série temporal é transformada através do código *R*, pela função *ts* composta pelos argumentos – os dados, a frequência, o ano inicial e o ano final. A série temporal representada na figura 3 pertence ao relatório “Série Temporal”, que se encontra explicado em anexo na figura 33.

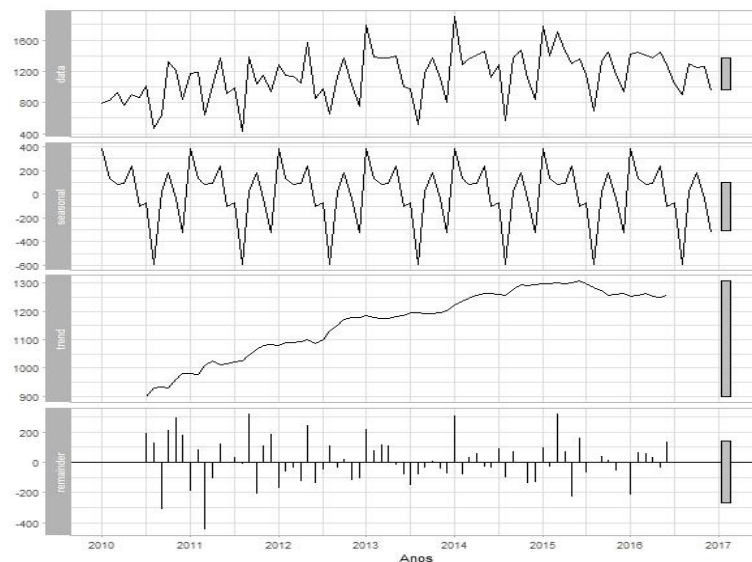


Figura 3 - Decomposição de uma série temporal

Este relatório ‘Série Temporal’ tem frequência mensal para o período temporal escolhido, neste caso, 2010 e 2016, o código transforma os dados numa série temporal através da função do *R*, a função *ts* tendo como argumentos, os dados, a frequência, o ano inicial e o ano final, tal como, `dados = ts (dados$racio, frequency = 12, start = c(c, 1), end= c(d,12))`. Inicialmente, a série é decomposta em quatro componentes pela função ‘*decompose*’. Observando o primeiro gráfico verifica-se que o comportamento da série se repete todos os anos, sendo que nos meses de verão (julho a setembro) o rácio apresenta valores muito inferiores aos meses de inverno (dezembro e janeiro), o que evidencia uma forte componente de sazonalidade, o que é confirmado através do

segundo gráfico. É perceptível que as oscilações se repetem todos os anos seguindo sempre o mesmo padrão. Assim, a sazonalidade existente na série é explicada pela ocorrência de mais consultas na área de saúde de medicina geral e familiar durante os meses de inverno e, contrariamente, nos meses de verão esta especialidade não é tão procurada e muitas das clínicas que prestam serviços de saúde encontram-se encerradas para férias. Segundo a Direção Geral de Saúde (2017), *"No outono/inverno ocorrem com frequência temperaturas baixas e há um aumento da incidência das infeções respiratórias na população, maioritariamente devidas à epidemia sazonal da gripe. No entanto, outros agentes virais e bacterianos ocorrem em simultâneo com a gripe."*, o que pode explicar os valores anteriormente analisados.

Analisando a série à primeira vista, tudo indica que esta poderá ter uma tendência pouco acentuada, como se pode confirmar no terceiro gráfico, o da tendência, dado que se apresenta tendencialmente crescente até ao final do ano de 2015. Em 2016 o seu comportamento é mais regular e pode ser explicado pela informatização dos métodos de trabalho e, conseqüentemente, pelo aumento do número de registos dos atos médicos ao longo dos anos.

Por último, o quarto gráfico apresenta a componente errática que representa os movimentos aleatórios que não são explicados por nenhuma das outras componentes e representa, portanto, todos os fenómenos de carácter aleatório.

Pela análise ao gráfico é possível perceber que não tem nenhum padrão definido para o ruído, isto é, movimentos aleatórios e pode considerar-se um sinal positivo.

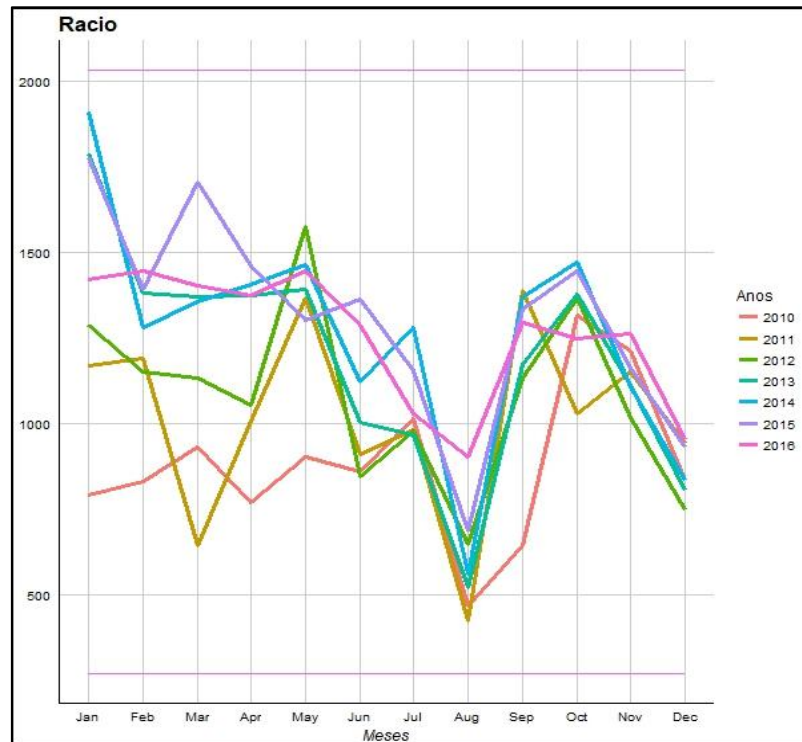


Figura 4 - Outliers

Relativamente aos *outliers*, sabe-se que um ponto é considerado *outlier* quando o valor observado é superior ou é inferior ao intervalo interquartil, "*interquartile range*", que abreviado é "*iqr*". Então, tem-se $IQR = Q_3 - Q_1$, em que, Q_1 tem probabilidade de 25% e o Q_3 tem probabilidade de 75%, o que leva a que se obtenham os seguintes limites:

$$\text{Limite Superior} = Q_3 + 1.5 \times IQR$$

$$\text{Limite Inferior} = Q_1 - 1.5 \times IQR$$

Assim, facilmente se encontram os valores dos limites:

$$\text{Limite Superior} = 2032.31$$

$$\text{Limite Inferior} = 272.31$$

Dada a figura 4 é possível perceber que existe um ponto que se destaca dos demais, e refere-se ao ano de 2011 no mês de março. Este comportamento explica-se porque foram registados cerca de 2580 atos médicos em quatro prestadores de serviços de saúde. O ponto revela-se abaixo do normal porque o número de

prestadores é o dobro que a maioria dos meses do ano de 2011. Em contrapartida, no mês de março e no ano de 2015 o número de atos médicos registados é quase o triplo de 2011 (6821) e o número de prestadores de serviços mantém-se. Não se registam *outliers* entre o ano de 2010 e 2016. As restantes observações apresentam valores da componente errática entre os 500 e os 1500 atos registados.

Alisamento Exponencial

Considere-se uma série de dados observados até ao instante $t-1$ e que se pretende obter uma previsão pontual para o instante t . De modo a obter as estimativas dos valores futuros da série, o método do alisamento exponencial simples usa a previsão do último instante. Portanto, a figura 6, apresenta a previsão desde o instante $t-1$, neste caso, 2016, e pela observação este modelo prevê que os meses de verão registem valores inferiores aos restantes meses do ano como continuação do que aconteceu até ao ano de 2016. O cálculo dos valores da previsão é feito pela função $etsforecast = forecast(ets, h=24)$, como é possível verificar na figura 16 em anexo. É também de referir que a azul está representado o nível de confiança que indica o grau de fiabilidade que a previsão deste parâmetro tem.

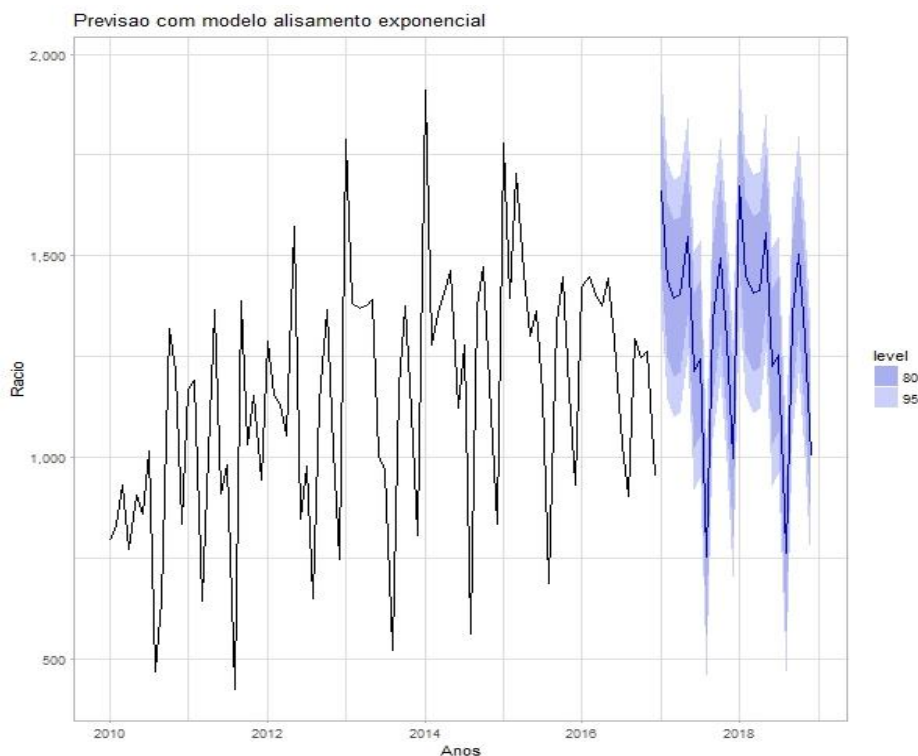


Figura 5 - Previsão com o modelo Alisamento Exponencial

Holt-Winters

O método de *Holt-Winters* multiplicativo é adequado a séries com tendência linear e com movimentos de caráter sazonal. Apesar dos valores da previsão serem automaticamente calculados pela função $HWforecast = forecast(HW, h=24)$, como é possível confirmar na figura 17, em anexo, e não descreverem cada uma das equações que compõem este modelo, as equações de atualização são as seguintes:

$$a(t) = \frac{Y_t}{S_{t-s}} + (1 - \alpha)[a(t - 1) + b(t - 1)], \quad 0 < \alpha < 1$$

$$b(t) = \beta[a(t) + a(t - 1)] + (1 - \beta)b(t - 1), \quad 0 < \beta < 1$$

$$S_t = \frac{\gamma Y_t}{a(t)} + (1 - \gamma)S_{t-s}, \quad 0 < \gamma < 1$$

Onde $a(t)$, $b(t)$ e $S(t)$ representam as expressões do nível, do declive e do índice sazonal, respetivamente, da série. S é o comprimento de sazonalidade, isto é, o número de meses, trimestres ou semestres do ano, e por fim α e β são constantes de alisamento. As previsões h passos à frente do método de Holt-Winters multiplicativo são obtidas através da função:

$$P_{t+h} = [a(t) + b(t) * h] * S_{t+h-s}, h = 1, 2, \dots$$

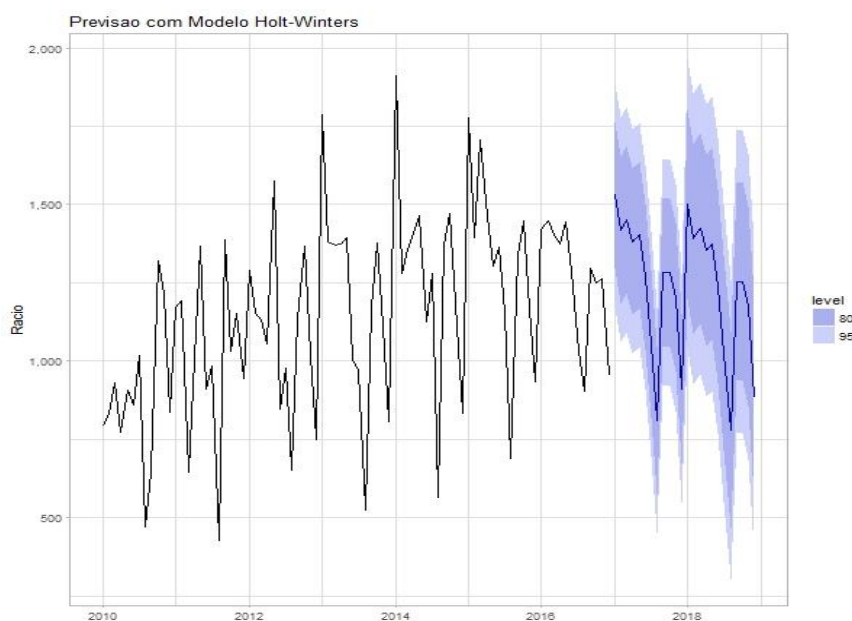


Figura 6 - Previsão com o modelo Holt-Winters

Pela observação da figura 6, existe uma quebra entre os meses de verão assemelhando-se ao que se verifica na previsão com o método de alisamento exponencial. Os registos nos meses de inverno são também superiores como anteriormente descrito.

Neste modelo, está também representado a azul o nível de confiança que indica o grau de fiabilidade que a previsão deste parâmetro tem.

O modelo mais adequado de previsão é dado através da informação da conclusão do relatório de cada um dos modelos de previsão, como podemos ver em anexo nas figuras 36 e 37. A informação é dada com base no cálculo do erro absoluto médio em percentagem, como se pode confirmar na tabela III.

Modelo	Erro absoluto médio percentual
Alisamento Exponencial	0,220924
Holt-Winters	0,242877

Tabela III - Erro absoluto médio

A diferença entre os dois modelos está presente no seu objetivo. Apesar do método de *Holt-Winters* ser um alisamento exponencial este procura minimizar o erro quadrático médio enquanto o método do Alisamento exponencial procura minimizar a função verosimilhança.

Facilmente se encontra o modelo mais adequado para análise da previsão que neste caso é o método de Alisamento Exponencial.

Conclusão

O projeto de estágio teve como objetivo analisar os dados sobre os prestadores de saúde que faziam os registos dos atos médicos e perceber como os dados podiam ser estruturados, de modo a criar relatórios e a facilitar a consulta dos mesmos.

A integração do ambiente *R* e a linguagem *SQL Server* simplificou o processo de criação dos relatórios necessários, tornando-o mais eficiente e automático.

Como o departamento de saúde apresentava alguns problemas relacionados com a qualidade e integridade dos dados, uma vez que se encontravam apenas numa base de dados e não tinham sido submetidos a tratamento prévio, neste projeto fez-se o tratamento e validação dos dados por forma de transformar esses dados em informação, na forma de relatórios, tornando-os numa mais-valia para a Quidgest.

A crescente e gradual inserção de dados que os prestadores foram introduzindo mostra a importância da informação, que é um processo que permite a construção de conhecimento. A informação é o elemento fundamental de que dependem os processos de decisão. É neste sentido que os sistemas de informação na saúde procuram ajudar a obter a melhor informação possível.

A posterior integração no GENIO permitiu aos colaboradores realizar consultas, reduzindo significativamente o tempo de análise, tornando mais céleres a tomada de decisões.

Relativamente à análise dos dados, é perceptível que os distritos Porto, Coimbra e Lisboa apresentam valores significativamente distantes de outros distritos tanto no registo de atos médicos como no número de prestadores de serviços de saúde em diversas patologias. Existe uma especialidade que se evidenciou, a medicina geral e familiar e, por isso, a caracterização e a previsão dos dados centrou-se nesta especialidade.

Este trabalho conseguiu ter um impacto positivo e ficou perceptível o que falta fazer para que os prestadores serviços de saúde convencionados sejam mais eficazes, reconhecendo que tem que existir algumas ações a ser executadas para que possam organizar de melhor forma a prestação de cuidados de saúde. Assim mesmo com o estágio, foi conseguida uma visão mais ampla sobre o que é a prestação de serviços de saúde, em particular as convenções.

Todos os resultados gerados podem ainda ser melhorados caso a inserção dos dados nas bases de dados seja cada vez mais completa de modo a evitar erros de leitura e de transformação de dados.

No geral, todos os objetivos propostos foram concretizados e embora este trabalho tenha sido feito numa área diferente ao pretendido inicialmente, ainda assim, as expectativas foram correspondidas e foi estimulante desde o primeiro momento realizar este Trabalho Final de Mestrado.

Referências bibliográficas

1. Cox, M. & Ellsworth, D. (1997), *Application-controlled demand paging for out-of-core visualization*, Proceeding VIS '97 Proceedings of the 8th conference on Visualization '97 Pages 235-ff.
Disponível em: <https://www.forbes.com/sites/gilpress/2013/05/09/a-very-short-history-of-big-data/#402568c765a1> [acesso a 05/07/2018]
2. Negash, S. (2004), *Business Intelligence*, Communications of the Association for Information Systems (Volume13, 2004) 177-195,
Disponível em:
<https://aisel.aisnet.org/cgi/viewcontent.cgi?article=3234&context=cais>
3. Batra, R. (2018), *SQL Primer: An accelerated Introduction to SQL Basics*, Apress.
Disponível em: https://www.safaribooksonline.com/library/view/sql-primer-an/9781484235768/A457348_1_En_6_Chapter.html [acesso a 06/07/2018]
4. Inmon, W. H. (2000), *What is a data warehouse?* Disponível em:
<http://repository.binus.ac.id/2009-2/content/M0584/M058459913.pdf>
[acesso a 05/07/2018]
5. História da Quidgest (2018)
www.quidgest.pt [acesso a 09/07/2018]
6. *Machine Learning* (2018)
www.investopedia.com/terms/m/machine-learning.asp
[acesso a 25/07/2018]
7. Kastrun, T. e Koesmarno, J. (2018), *SQL Server 2017 Machine Learning Services with R*, Packt.
8. *Stored Procedures* (2018)
www.w3schools.com/sql/sql_stored_procedures.asp [acesso a 06/09/2018]
9. Sobre o R (2018)
<https://www.r-project.org/about.html> [acesso a 26/03/2018]

10. Direção Geral de Saúde (2017), PLANO DE CONTINGÊNCIA SAÚDE SAZONAL – MÓDULO INVERNO [acesso a 13/09/2018]
11. Duan, Y. e Cao, G. (2015), *Understanding the Impact of Business Analytics on Innovation*, ECIS 2015 Completed Research Papers. Paper 40. [acesso a 26/01/2019]
12. Reiner, R.K., Turban, E., Potter, R.E. (2007). *Introduction to Information Systems—Supporting and Transforming Business*, John Wiley.
13. Silberschatz, A., Korth, H., Sudarshan, S. (2009), *Database System Concepts*, Sixth Edition, McGraw-Hill.
14. Ihaka, R., Gentleman, R. (1996), *R: A Language for Data Analysis and Graphics*, *Journal of Computational and Graphical Statistics* 5, no. 3 (1996): 299–314.
15. Ammenwerth, E; Gräber, S; Herrmann, G; Bürkle, T & König, J, (2003), *Evaluation of health information systems—problems and challenges*. *International Journal of Medical Informatics*.

ANEXOS

Anexo I - Stored Procedures

O *script* do *stored procedure* "prestador" declara o *tprestador* como o *input*, define um centro com coordenadas geográficas e se o número de linhas do *tprestador* for maior que 0 então o resultado será um gráfico com pontos com diferentes tamanhos dependendo do número de prestadores por distrito. Caso contrário, se o número de prestadores for menor que um, o gráfico apresenta uma mensagem "Não existem dados para os parâmetros selecionados", neste caso, um período de tempo.

```

CREATE proc [dbo].[prestador]
@DataInicial date,
@DataFinal date
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(RColorBrewer)
library(ggmap)
tprestador=InputDataSet
centro = c(-8.118896484375,39.53793974517628)
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
if(nrow(tprestador)>0){
plot=ggmap(get_googlemap(center=centro, scale=2, zoom=7), extent="normal")+
geom_point(aes(x=tprestador$Lon, y=tprestador$Lat, size = tprestador$Qt),data = tprestador, alpha= 0.4,
col= "darkorange4")+
scale_size(range=c(0,20))+
labs(x='Longitude',y='Latitude', size = 'Nr Prestadores')
print(plot) } else if (nrow(tprestador)<1) {
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black" )
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what=raw(), n=1e6));'

,@input_data_1 = N'select DISTRITO, count(DISTRITO) as [Qt], Lon, Lat
from(select distinct codprest, [GPSDADOS].[dbo].[Dados].DISTRITO, Lon, Lat
FROM [GPSDADOS].[dbo].[Dados]
inner join [GPSDADOS].[dbo].[Distritos] on [GPSDADOS].[dbo].[Dados].DISTRITO=[GPSDADOS].[dbo].[Distritos].Distrito
where data between @a and @b) as subquery
group by DISTRITO, Lon, Lat
order by DISTRITO'

,@parallel = 1
,@params= N'@a date, @b date'
,@a=@DataInicial
,@b=@DataFinal
with result sets ((plot varbinary (max)));
end

```

Figura 7 - Código do stored procedure 'prestador'

O *stored procedure* "oftalmologia" dá resultado do número de atos por concelho através de uma imagem de satélite conseguida através da fórmula do *ggmap*, de modo a perceber a densidade do número de atos e a localização dos registos.

```

create proc [dbo].[oftalmologia]
@Especialidade nvarchar (max)
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(RColorBrewer)
library(ggmap)
tespecialConce=InputDataSet
centro = c(-8.118896484375,39.53793974517628)
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
if(nrow(tespecialConce)>0){
plot=ggmap(get_googlemap(center=centro, scale=2, zoom=7), extent="normal")+
geom_point(aes(x=tespecialConce$lon, y=tespecialConce$lat, size = (tespecialConce$Total)),
data = tespecialConce, alpha= 0.4, col= "orangered4")+
scale_size(range=c(0,20))+
labs(x="Longitude", y="Latitude", size = 'Nr de atos')
print(plot)} else if (nrow(tespecialConce)<1) {
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black") }
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what=raw(), n=1e6));'
, @input_data_1 = N'select id, lon, lat, concelhos.concelho,count([GPSDADOS].[dbo].[Concelhos].CONCELHO) as Total
from [GPSDADOS].[dbo].[Concelhos]
inner join [GPSDADOS].[dbo].[Dados] on [GPSDADOS].[dbo].[Dados].CONCELHO=[GPSDADOS].[dbo].[Concelhos].CONCELHO
where data > '1900-01-01' and ESPECIAL like @a
group by lon,lat,[GPSDADOS].[dbo].[Concelhos].CONCELHO, id
order by [GPSDADOS].[dbo].[Concelhos].CONCELHO asc'
,@parallel = 1
,@params= N'@a nvarchar(max)'
,@a=@Especialidade
with result sets ((plot varbinary (max)));
end

```

Figura 8 - Código do stored procedure "oftalmologia"

O *stored procedure* "concelho" representa graficamente a densidade de cada especialidade por concelho, através do gráfico do *ggmap*.

```

CREATE PROC [dbo].[concelho]
@Distrito nvarchar (max),
@Especialidade nvarchar (max),
@DataInicial date,
@DataFinal date
as
begin
exec sp_execute_external_script
@language = N'R'
,@script = N'library(ggplot2)
library(RColorBrewer)
library(ggmap)
tespecialConce=InputDataSet
centro = c(tespecialConce[1,5],tespecialConce[1,6])
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
if(nrow(tespecialConce)>0){
plot=ggmap(get_googlemap(center=centro, scale=2, zoom=9), extent="normal")+
geom_point(aes(x=tespecialConce$Lon, y=tespecialConce$Lat, size = (tespecialConce$Total)),
data = tespecialConce, alpha= 0.4,
col= "darkorchid4")+
labs(x="Longitude", y="Latitude", size = 'Nr. de Atos')
print(plot)} else if (nrow(tespecialConce)<1) {
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black") }
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what=raw(), n=1e6));'

,@input_data_1 = N'select [GPSDADOS].[dbo].[Concelhos].Lon,[GPSDADOS].[dbo].[Concelhos].Lat,[GPSDADOS].[dbo].[Dados].CONCELHO,
count([GPSDADOS].[dbo].[Dados].CONCELHO)as [Total],[GPSDADOS].[dbo].[Centros].lon,
[GPSDADOS].[dbo].[Centros].lat
from [GPSDADOS].[dbo].[Dados]
inner join [GPSDADOS].[dbo].[Concelhos] on [GPSDADOS].[dbo].[Concelhos].[CONCELHO]=[GPSDADOS].[dbo].[Dados].[CONCELHO]
inner join [GPSDADOS].[dbo].[Distritos] on [GPSDADOS].[dbo].[Distritos].[Distrito]=[GPSDADOS].[dbo].[Dados].[DISTRITO]
inner join [GPSDADOS].[dbo].[Centros] on [GPSDADOS].[dbo].[Centros].[Distrito]=[GPSDADOS].[dbo].[Dados].[DISTRITO]
where [GPSDADOS].[dbo].[Dados].DISTRITO like @a and [GPSDADOS].[dbo].[Dados].[ESPECIAL] like @b and data between @c and @d
group by [GPSDADOS].[dbo].[Dados].CONCELHO, [GPSDADOS].[dbo].[Dados].DISTRITO, [GPSDADOS].[dbo].[Concelhos].Lon,
[GPSDADOS].[dbo].[Concelhos].Lat, [GPSDADOS].[dbo].[Distritos].Lon, [GPSDADOS].[dbo].[Distritos].Lat,
[GPSDADOS].[dbo].[Centros].Lon, [GPSDADOS].[dbo].[Centros].Lat
order by [GPSDADOS].[dbo].[Dados].CONCELHO'

,@parallel = 1
,@params= N'@a nvarchar(max),@b nvarchar(max),@c date, @d date'
,@a=@Distrito
,@b=@Especialidade
,@c=@DataInicial
,@d=@DataFinal
with result sets ((plot varbinary (max)));
end

```

Figura 9 - Código do *stored procedure* "concelho"

O *stored procedure* "registo" resulta num gráfico com os registos mensais por especialidade, distrito e ano, em número de atos médicos. Para cada linha que apresentasse valor 'NA', foi-lhe atribuído valor 0.

```
ALTER proc [dbo].[registo]
@Especialidade nvarchar (max),
@Distrito nvarchar (max),
@Ano int
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(scales)
library(dplyr)
library(ggthemes)
registo=InputDataSet
df=data.frame(1:12)
colnames(df)="Mes"
df=left_join(df,registo, by = "Mes")
for(i in 1:nrow(df)){
if(is.na(df$Qt[i])==T){
df$Qt[i]=0
}}
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
if(nrow(registo)>0){
plot=ggplot()+
geom_line(aes(x=df$Mes,y=df$Qt), size=0.5)+
geom_line(aes(x=df$Mes,y=mean(df$Qt)), color= 'blue')+
scale_x_continuous(breaks = 1:12)+
geom_point(aes(x=df$Mes,y=df$Qt))+
scale_y_continuous(limits = c(0,max(df$Qt)))+
labs(x="Meses", y="Nr. de Atos Medicos")+
geom_text(aes(x = df$Mes[12], y = mean(df$Qt), label = "Media"), color ="blue")+
theme_gdocs()_---\,
print(plot)} else if (nrow(registo)<1) {
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black") }
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what=raw(), n=1e6));'
, @input_data_1 = N'SELECT MONTH(DATA) Mes,COUNT(DISTRITO) Qt
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b AND YEAR(DATA) = @c and Especial like @a
GROUP BY DISTRITO, YEAR (DATA), MONTH(DATA)
ORDER BY YEAR (DATA), MONTH(DATA), DISTRITO'
,@parallel = 1
,@params= N'@a nvarchar(max), @b nvarchar(max), @c int'
,@a=@Especialidade
,@b=@Distrito
,@c=@Ano
with result sets ((plot varbinary (max)));
end
```

Figura 10 - Código do *stored procedure* "registo"

O *stored procedure* "teste1" faz o cálculo do rácio entre o número de atos médicos sobre número de prestadores.

```
ALTER proc [dbo].[teste1]
@Especialidade nvarchar (max),
@Distrito nvarchar (max),
@Ano1 int,
@Ano2 int
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(scales)
library(dplyr)
library(ggthemes)
racio=InputDataSet
ano=filter(racio,racio$ano==c | racio$ano==d )
ano$racio = ano$Qt/ano$Total
df=data.frame(1:12)
colnames(df)="Mes"
df=left_join(df,ano, by = "Mes")
for(i in 1:nrow(df)){
  if(is.na(df$Qt[i])==T){
    df$racio[i]=0
  }
}
a=filter(df,df$ano==c)
b=filter(df,df$ano==d)
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
if(nrow(a)>0 & nrow(b)>0){
plot=ggplot()+
  geom_line(aes(x=a$Mes,y=a$racio), size=0.5, color= "red")+
  geom_line(aes(x=a$Mes, y=mean(a$racio)), color= "red")+
  geom_line(aes(x=b$Mes,y=b$racio), size=0.5, color= "blue")+
  geom_line(aes(x=b$Mes, y=mean(b$racio)), color= "blue")+
  scale_x_continuous(breaks = 1:12)+
  geom_point(aes(x=df$Mes,y=df$racio))+
  scale_y_continuous(limits = c(0,max(df$racio)))+
  labs(x="Meses", y="Racio")+
  geom_text(aes(x = a$Mes[12], y = mean(a$racio), label = "Media"), color = "red")+
  geom_text(aes(x = b$Mes[12], y = mean(b$racio), label = "Media"), color = "blue")+
  theme_gdocs()
print(plot) else if (nrow(a)<1 | nrow(b)<1) {
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black") }
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what="raw"), n=1e6));'
, @input_data_1 = N'SELECT sub1.ano,sub1.Mes, Total,Qt
FROM(SELECT ano,Mes, COUNT(codprest) Total
FROM (SELECT distinct codprest,MONTH(data) as Mes,year(data) as ano
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a) SUB
GROUP BY ano,Mes) SUB1
INNER JOIN (SELECT MONTH(DATA) Mes,year(data) as ano,COUNT(DISTRITO) Qt
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a
GROUP BY DISTRITO, yEAR (DATA), MONTH(DATA)) SUB2 ON SUB2.ano=SUB1.ano and SUB2.mes=SUB1.mes
Order by sub1.ano,sub1.Mes'

@parallel = 1
@params= N'@a nvarchar(max), @b nvarchar(max), @c int, @d int'
@a=@Especialidade
@b=@Distrito
@c=@Ano1
@d=@Ano2
with result sets ((plot varbinary (max)));
end
```

Figura 11 - Código do *stored procedure* "teste1"

O *stored procedure* "graficommes" representa os registos mensais do rácio dos meses dos anos pretendidos, por cada especialidade e distrito.

```
ALTER proc [dbo].[graficommes]
@Especialidade nvarchar (max),
@Distrito nvarchar (max),
@AnoInicial int,
@AnoFinal int
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(scales)
library(dplyr)
library(ggthemes)
library(forecast)
racio=InputDataSet
racio=filter(racio,racio$ano>(c-1) & racio$ano<(d+1))
racio$racio = racio$Qt/racio$Total

if (nrow(racio)>0){
for(j in 0:(max(racio$ano)-min(racio$ano))){
for(i in 1:nrow(racio)){
if(racio$ano[i] == min(racio$ano)+j){racio$meses[i]=racio$Mes[i]+12*j}
}
}
meses=1:((max(racio$ano)-min(racio$ano)+1)*12)
meses=data.frame(meses)
e=left_join(meses,racio,by="meses")
dados=data.frame(e$meses,e$racio)
colnames(dados)=c("meses","racio")

for(i in 1:nrow(dados)){
if(is.na(dados$racio[i])==T){
dados$racio[i]=0
}
}
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
if(nrow(dados)>0){
dados = ts(dados$racio, frequency = 12, start = c(c, 1), end= c(d,12))
plot=ggmonthplot(dados, xlab="Meses", ylab= " ",main = "Racio")+
theme_light()
print(plot)} else if (nrow(dados)<1) {
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black" )} else if (nrow(racio)<1) {
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black" )}
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what="raw()", n=1e6));'
, @input_data_1 = N'SELECT sub1.ano,sub1.Mes, Total,Qt
FROM(SELECT ano,Mes, COUNT(codprest) Total
FROM (SELECT distinct codprest,MONTH(data) as Mes,year(data) as ano
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a) SUB
GROUP BY ano,Mes) SUB1
INNER JOIN (SELECT MONTH(DATA) Mes,year(data) as ano,COUNT(DISTRITO) Qt
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a
GROUP BY DISTRITO, yEAR (DATA), MONTH(DATA)) SUB2 ON SUB2.ano=SUB1.ano and SUB2.mes=SUB1.mes
Order by sub1.ano,sub1.Mes'
,@parallel = 1
,@params= N'@a nvarchar(max), @b nvarchar(max), @c int, @d int'
,@a=@Especialidade
,@b=@Distrito
,@c=@AnoInicial
,@d=@AnoFinal
with result sets ((plot varbinary (max)));
end
```

Figura 12 - Código do *stored procedure* "graficommes"

O *stored procedure* "st" transforma os dados de um período de tempo pretendido numa série temporal. O objeto dados é dado pela função do R, *ts*, e tem como argumentos a seguinte fórmula, $\text{dados} = \text{ts}(\text{dados}\$\text{racao}, \text{frequency} = 12, \text{start} = \text{c}(\text{c}, 1), \text{end} = \text{c}(\text{d}, 12))$. Foi criado um *data frame* denominado por *dados*, que tem como campos os meses e o rácio, tendo as colunas o mesmo nome, a frequência é de 12 meses por cada ano escolhido, o vetor *start* permite escolher o ano inicial "c", neste caso, 2010, e começa no mês 1 e o vetor *end* permite escolher o ano final "d", neste caso, 2016.

```
ALTER proc [dbo].[st]
@Especialidade nvarchar (max),
@Distrito nvarchar (max),
@AnoInicial int,
@AnoFinal int
as
begin
exec sp_execute_external_script
@Language = N'R'
, @script = N'library(ggplot2)
library(scales)
library(dplyr)
library(ggthemes)
library(forecast)
racao=InputDataSet
racao=filter(racao,racao$ano>c-1) & racio$ano<(d+1)
racao$racao = racio$Qt/racio$Total

if (nrow(racao)>0){
for(j in 0:(max(racao$ano)-min(racao$ano))){
for(i in 1:nrow(racao)){
if(racao$ano[i] == min(racao$ano)+j){racao$meses[i]=racao$Mes[i]+12*j}
}
}
meses=1:(max(racao$ano)-min(racao$ano)+1)*12
meses=data.frame(meses)
e=left_join(meses,racao,by="meses")
dados=data.frame(e$meses,e$racao)
colnames(dados)=c("meses","racao")

for(i in 1:nrow(dados)){
if(is.na(dados$racao[i])==T){
dados$racao[i]=0
}
}

image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
if(nrow(dados)>0){
dados = ts(dados$racao, frequency = 12, start = c(c, 1), end= c(d,12))
plot=ggseasonplot(dados,main="Racio",xlab="Meses")+
labs(color="Anos") +
geom_line(size = 1.2) +
theme_economist() +
theme(plot.title = element_text(hjust=0.5))+
theme_gdocs()
print(plot)} else if (nrow(dados)<1) {
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black" )} else if (nrow(racao)<1) {
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black")}
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what=raw(), n=1e6));'
, @input_data_1 = N'SELECT sub1.ano,sub1.Mes, Total,Qt
FROM(SELECT ano,Mes, COUNT(codprest) Total
FROM (SELECT distinct codprest,MONTH(data) as Mes,year(data) as ano
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a) SUB
GROUP BY ano,Mes) SUB1
INNER JOIN (SELECT MONTH(DATA) Mes,year(data) as ano,COUNT(DISTRITO) Qt
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a
GROUP BY DISTRITO, YEAR (DATA), MONTH(DATA)) SUB2 ON SUB2.ano=SUB1.ano and SUB2.mes=SUB1.mes
Order by sub1.ano,sub1.Mes'

@parallel = 1
@params= N'@a nvarchar(max), @b nvarchar(max), @c int, @d int'
@a=@Especialidade
@b=@Distrito
@c=@AnoInicial
@d=@AnoFinal
with result sets ((plot varbinary (max)));
end
```

Figura 13 - Código do stored procedure "st"

O *stored procedure* "decompose" decompõe os dados do rácio em quatro componentes: os dados reais, a sazonalidade, a tendência e a componente errática. O *data frame* que é usado na construção da figura 3 é o mesmo que compõe o *stored procedure* "st". Através da função *ts* do R é posteriormente criado um gráfico automático com a informação presente no *data frame*.

```
ALTER proc [dbo].[decompose]
@Especialidade nvarchar (max),
@Distrito nvarchar (max),
@AnoInicial int,
@AnoFinal int
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(scales)
library(dplyr)
library(ggthemes)
library(forecast)
racio=InputDataSet
racio=filter(racio,racio$ano>(c-1) & racio$ano<(d+1))
racio$racio = racio$Qt/racio$Total

if (nrow(racio)>0){
for(j in 0:(max(racio$ano)-min(racio$ano))){
for(i in 1:nrow(racio)){
if(racio$ano[i] == min(racio$ano)+j){racio$meses[i]=racio$Mes[i]+12*j}
}
}
}
meses=1:(max(racio$ano)-min(racio$ano)+1)*12
meses=data.frame(meses)
e=left_join(meses,racio,by="meses")
dados=data.frame(e$meses,e$racio)
colnames(dados)=c("meses","racio")

for(i in 1:nrow(dados)){
if(is.na(dados$racio[i])==T){
dados$racio[i]=0
}
}

image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
if(nrow(dados)>0){
dados = ts(dados$racio, frequency = 12, start = c(c, 1), end= c(d,12))
plot= autoplot(decompose(dados), main = " ", xlab="Anos")+
theme_light()
print(plot)} else if (nrow(dados)<1) {
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black")} } else if (nrow(racio)<1) {
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black")}
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what=raw(), n=1e6));'
, @input_data_1 = N'SELECT sub1.ano,sub1.Mes, Total,Qt
FROM(SELECT ano,Mes, COUNT(codprest) Total
FROM (SELECT distinct codprest,MONTH(data) as Mes,year(data) as ano
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a) SUB
GROUP BY ano,Mes) SUB1
INNER JOIN (SELECT MONTH(DATA) Mes,year(data) as ano,COUNT(DISTRITO) Qt
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a
GROUP BY DISTRITO, yEAR (DATA), MONTH(DATA)) SUB2 ON SUB2.ano=SUB1.ano and SUB2.mes=SUB1.mes
Order by sub1.ano,sub1.Mes'
,@parallel = 1
,@params= N'@a nvarchar(max), @b nvarchar(max), @c int, @d int'
,@a=@Especialidade
,@b=@Distrito
,@c=@AnoInicial
,@d=@AnoFinal
with result sets ((plot varbinary (max)));
end
```

Figura 14 - Código do *stored procedure* "decompose"

O *stored procedure* "outlier" apresenta graficamente o cálculo dos limites inferior e superior dos *outliers*, através da fórmula supramencionada.

```
ALTER proc [dbo].[outlier]
@Especialidade nvarchar (max),
@Distrito nvarchar (max),
@AnoInicial int,
@AnoFinal int
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(scales)
library(dplyr)
library(ggthemes)
library(forecast)
racio=InputDataSet
racio=filter(racio,racio$ano>(c-1) & racio$ano<(d+1))
racio$racio = racio$Qt/racio$Total

if (nrow(racio)>0){
for(j in 0:(max(racio$ano)-min(racio$ano))){
for(i in 1:nrow(racio)){
if(racio$ano[i] == min(racio$ano)+j){racio$meses[i]=racio$Mes[i]+12*j}
}
}
meses=1:((max(racio$ano)-min(racio$ano)+1)*12)
meses=data.frame(meses)
e=left_join(meses,racio,by="meses")
dados=data.frame(e$meses,e$racio)
colnames(dados)=c("meses","racio")

for(i in 1:nrow(dados)){
if(is.na(dados$racio[i])==T){
dados$racio[i]=0
}
}
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
if(nrow(dados)>0){
dados = ts(dados$racio, frequency = 12, start = c(c, 1), end= c(d,12))
Q1=quantile(dados,probs = 0.25)
Q3=quantile(dados,probs = 0.75)
iqr=Q3-Q1
low=Q1-1.5*iqr
high=Q3+1.5*iqr
plot=ggseasonplot(dados,main="Racio",xlab="Meses")+
labs(color="Anos") +
geom_line(size = 1.2) +
geom_line(aes(y=low))+
geom_line(aes(y=high))+
theme(plot.title = element_text(hjust=0.5))+
theme_gdocs()
print(plot)} else if (nrow(dados)<1) {
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black")} else if (nrow(racio)<1) {
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black")}
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what=raw(), n=1e6));'
,@input_data_1 = N'SELECT sub1.ano,sub1.Mes, Total,Qt
FROM(SELECT ano,Mes, COUNT(codprest) Total
FROM (SELECT distinct codprest,MONTH(data) as Mes,year(data) as ano
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a) SUB
GROUP BY ano,Mes) SUB1
INNER JOIN (SELECT MONTH(DATA) Mes,year(data) as ano,COUNT(DISTRITO) Qt
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a
GROUP BY DISTRITO, YEAR (DATA), MONTH(DATA)) SUB2 ON SUB2.ano=SUB1.ano and SUB2.mes=SUB1.mes
Order by sub1.ano,sub1.Mes'
,@parallel = 1
,@params= N'@a nvarchar(max), @b nvarchar(max), @c int, @d int'
,@a=@Especialidade
,@b=@Distrito
,@c=@AnoInicial
,@d=@AnoFinal
with result sets ((plot varbinary (max)));
end
```

Figura 15 - Código do *stored procedure* "outlier"

O *stored procedure* "alisamentoex" permite o cálculo da previsão pelo método do Alisamento Exponencial.

```
ALTER proc [dbo].[alisamentoex]
@Especialidade nvarchar (max),
@Distrito nvarchar (max),
@AnoInicial int,
@AnoFinal int
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(scales)
library(dplyr)
library(ggthemes)
library(forecast)
racio=InputDataSet
racio=filter(racio,racio$ano>(c-1) & racio$ano<(d+1))
racio$racio = racio$Qt/racio$Total

if (nrow(racio)>0){
for(j in 0:(max(racio$ano)-min(racio$ano))){
for(i in 1:nrow(racio)){
if(racio$ano[i] == min(racio$ano)+j){racio$meses[i]=racio$Mes[i]+12*j}
}
}
meses=1:(max(racio$ano)-min(racio$ano)+1)*12
meses=data.frame(meses)
e=left_join(meses,racio,by="meses")
dados=data.frame(e$meses,e$racio)
colnames(dados)=c("meses","racio")

for(i in 1:nrow(dados)){
if(is.na(dados$racio[i])!=T){
dados$racio[i]=0
}
}
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
if(nrow(dados)>0){
dados = ts(dados$racio, frequency = 12, start = c(c, 1), end= c(d,12))
ets=ets(dados)
etsforecast=forecast(ets, h=24)
plot= autoplot(etsforecast, main="Previsao com modelo alisamento exponencial", xlab="Anos", ylab="Racio")+
scale_y_continuous(labels = comma)+
scale_x_continuous(breaks= pretty_breaks())+
theme_light()
print(plot)} else if (nrow(dados)<1) {
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black")} else if (nrow(racio)<1) {
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black")}
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what=raw(), n=1e6));'
, @input_data_1 = N'SELECT sub1.ano,sub1.Mes, Total,Qt
FROM(SELECT ano,Mes, COUNT(codprest) Total
FROM (SELECT distinct codprest,MONTH(data) as Mes,year(data) as ano
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a) SUB
GROUP BY ano,Mes) SUB1
INNER JOIN (SELECT MONTH(DATA) Mes,year(data) as ano,COUNT(DISTRITO) Qt
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a
GROUP BY DISTRITO, YEAR (DATA), MONTH(DATA)) SUB2 ON SUB2.ano=SUB1.ano and SUB2.mes=SUB1.mes
Order by sub1.ano,sub1.Mes'
,@parallel = 1
,@params= N'@a nvarchar(max), @b nvarchar(max), @c int, @d int'
,@a=@Especialidade
,@b=@Distrito
,@c=@AnoInicial
,@d=@AnoFinal
with result sets ((plot varbinary (max)));
end
```

Figura 16 - Código do *stored procedure* "alisamentoex"

O stored procedure “previsão” permite o cálculo da previsão pelo método do Holt-Winters.

```
ALTER proc [dbo].[previsao]
@Especialidade nvarchar (max),
@Distrito nvarchar (max),
@AnoInicial int,
@AnoFinal int
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(scales)
library(dplyr)
library(ggthemes)
library(forecast)
racio=InputDataSet
racio=filter(racio,racio$ano>(c-1) & racio$ano<(d+1))
racio$racio = racio$Qt/racio$Total

if (nrow(racio)>0){
for(j in 0:(max(racio$ano)-min(racio$ano))){
for(i in 1:nrow(racio)){
if(racio$ano[i] == min(racio$ano)+j){racio$meses[i]=racio$Mes[i]+12*j}
}
}
meses=1:((max(racio$ano)-min(racio$ano)+1)*12)
meses=data.frame(meses)
e=left_join(meses,racio,by="meses")
dados=data.frame(e$meses,e$racio)
colnames(dados)=c("meses","racio")

for(i in 1:nrow(dados)){
if(is.na(dados$racio[i])==T){
dados$racio[i]=0
}
}

image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
if(nrow(dados)>0){
dados = ts(dados$racio, frequency = 12, start = c(c, 1), end= c(d,12))
HW=HoltWinters(dados)
HWforecast=forecast(HW, h=24)
plot= autoplot(HWforecast, main="Previsao com Modelo Holt-Winters", xlab="Anos", ylab="Racio")+
scale_y_continuous(labels = comma)+
scale_x_continuous(breaks= pretty_breaks())+
theme_light()
print(plot)} else if (nrow(dados)<1) {
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black")} } else if (nrow(racio)<1) {
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
plot(c(0, 1), c(0, 1), ann = F, bty = 'n', type = 'n', xaxt = 'n', yaxt = 'n')
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black")}
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what=raw(), n=1e6));'
, @input_data_1 = N'SELECT sub1.ano,sub1.Mes, Total,Qt
FROM(SELECT ano,Mes, COUNT(codprest) Total
FROM (SELECT distinct codprest,MONTH(data) as Mes,year(data) as ano
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a) SUB
GROUP BY ano,Mes) SUB1
INNER JOIN (SELECT MONTH(DATA) Mes,year(data) as ano,COUNT(DISTRITO) Qt
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a
GROUP BY DISTRITO, YEAR (DATA), MONTH(DATA)) SUB2 ON SUB2.ano=SUB1.ano and SUB2.mes=SUB1.mes
Order by sub1.ano,sub1.Mes'
,@parallel = 1
,@params= N'@a nvarchar(max), @b nvarchar(max), @c int, @d int'
,@a=@Especialidade
,@b=@Distrito
,@c=@AnoInicial
,@d=@AnoFinal
with result sets ((plot varbinary (max)));
end
```

Figura 17 - Código do stored procedure "previsao"

O *stored procedure* "rácio" calcula o rácio entre o número de atos médicos sobre o número de prestadores de saúde através da função *inner join* entre as duas *queries*.

```
ALTER proc [dbo].[rácio]
@Especialidade nvarchar (max),
@Distrito nvarchar (max),
@Ano int
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(scales)
library(dplyr)
library(ggthemes)
racio=InputDataSet
racio$racio = racio$Qt/racio$Total
df=data.frame(1:12)
colnames(df)="Mes"
df=left_join(df,racio, by = "Mes")
for(i in 1:nrow(df)){
if(is.na(df$Qt[i])==T){
df$racio[i]=0
}}
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
if(nrow(racio)>0){
plot=ggplot()+
geom_line(aes(x=df$Mes,y=df$racio), size=0.5)+
geom_line(aes(x=df$Mes, y=mean(racio$racio)), color= "red")+
geom_line(aes(x=df$Mes, y=sd(racio$racio)), color="blue")+
scale_x_continuous(breaks = 1:12)+
geom_point(aes(x=df$Mes,y=df$racio))+
scale_y_continuous(limits = c(0,max(df$racio)))+
labs(x="Meses", y="Racio")+
geom_text(aes(x = df$Mes[12], y = mean(racio$racio), label = "Media"), color = "red")+
geom_text(aes(x=df$Mes[12], y=sd(racio$racio), label = "Desvio-padrao"), color="blue")+
theme_gdocs()
print(plot)} else if (nrow(racio)<1) {
plot(c(0, 1), c(0, 1), ann = F, bty = "n", type = "n", xaxt = "n", yaxt = "n")
text(x = 0.5, y = 0.5, paste("Nao existem dados para os parametros selecionados.\n"),
cex = 1, col = "black") }
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what="raw()", n=1e6));'
, @input_data_1 = N'SELECT sub1.Mes, Total,Qt
FROM(SELECT Mes, COUNT(codprest) Total
FROM (SELECT distinct codprest,MONTH(data) as Mes
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b AND YEAR(DATA) = @c and Especial like @e) SUB1
GROUP BY Mes) SUB1
INNER JOIN (SELECT MONTH(DATA) Mes,COUNT(DISTRITO) Qt
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b AND YEAR(DATA) = @c and Especial like @e
GROUP BY DISTRITO, YEAR (DATA), MONTH(DATA)) SUB2 ON SUB2.MES=SUB1.Mes
Order by sub1.Mes'
,@parallel = 1
,@params= N'@a nvarchar(max), @b nvarchar(max), @c int'
,@e=@Especialidade
,@b=@Distrito
,@c=@Ano
with result sets ((plot varbinary (max)));
end
```

Figura 18 - Código do *stored procedure* "rácio"

O *stored procedure* “provincia” permite escolher uma especialidade para a consulta de especialidades por concelho.

```
ALTER proc [dbo].[provincia]
@Especialidade nvarchar (max)
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(RColorBrewer)
library(ggmap)
tespecialConce=InputDataSet
centro = c(-8.118896484375,39.53793974517628)
image_file=tempfile();
jpeg(filename=image_file, width=600, height=600, quality=100);
plot=ggmap(get_googlemap(center=centro, scale=2, zoom=7), extent="normal")+
geom_point(aes(x=tespecialConce$lon, y=tespecialConce$lat, size = (tespecialConce$Total)),data = tespecialConce, alpha= 0.4,
col="orangered4")+
labs(x="Longitude", y="Latitude", size = "Nr de atos")
print(plot)
dev.off();
OutputDataSet = data.frame(data=readBin (file(image_file, "rb"), what="raw()", n=1e6));'
, @input_data_1 = N'select id, lon, lat, Distritos.Distrito, count([GPSDADOS].[dbo].[Distritos].Distrito) as Total
from [GPSDADOS].[dbo].[Distritos]
inner join [GPSDADOS].[dbo].[Dados] on [GPSDADOS].[dbo].[Dados].DISTRITO=[GPSDADOS].[dbo].[Distritos].Distrito
where data between ''2016-01-01'' and ''2016-12-31'' and ESPECIAL like @s
group by lon,lat,[GPSDADOS].[dbo].[Distritos].Distrito, id
order by [GPSDADOS].[dbo].[Distritos].Distrito asc'
,@parallel = 1
,@params= N'@s nvarchar(max)'
,@s=@Especialidade
with result sets ((plot varbinary (max)));
end
```

Figura 19 - Código do stored procedure "provincia"

O *stored procedure* “tabela” permite auxiliar as descrições dos relatórios.

```
ALTER proc [dbo].[tabela]
@Especialidade nvarchar (max),
@Distrito nvarchar (max),
@DataInicial date,
@DataFinal date
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(RColorBrewer)
library(ggmap)
tespecialConce=InputDataSet
OutputDataSet = data.frame(tespecialConce);'
, @input_data_1 = N'select [GPSDADOS].[dbo].[Dados].CONCELHO, count([GPSDADOS].[dbo].[Dados].CONCELHO)as [Total]
from [GPSDADOS].[dbo].[Dados]
inner join [GPSDADOS].[dbo].[Concelhos] on [GPSDADOS].[dbo].[Concelhos].[CONCELHO]=[GPSDADOS].[dbo].[Dados].[CONCELHO]
inner join [GPSDADOS].[dbo].[Distritos] on [GPSDADOS].[dbo].[Distritos].[Distrito]=[GPSDADOS].[dbo].[Dados].[DISTRITO]
inner join [GPSDADOS].[dbo].[Centros] on [GPSDADOS].[dbo].[Centros].[Distrito]=[GPSDADOS].[dbo].[Dados].[DISTRITO]
where [GPSDADOS].[dbo].[Dados].DISTRITO like @a and [GPSDADOS].[dbo].[Dados].[ESPECIAL] like @b and data between @c and @d
group by [GPSDADOS].[dbo].[Dados].CONCELHO
order by [GPSDADOS].[dbo].[Dados].CONCELHO'
,@parallel = 1
,@params= N'@a nvarchar(max),@b nvarchar(max), @c date, @d date'
,@a=@Distrito
,@b=@Especialidade
,@c=@DataInicial
,@d=@DataFinal
with result sets ((Concelhos nvarchar (max), Total nvarchar (max) ));
end
```

Figura 20 - Código do stored procedure "tabela"

O stored procedure "tabela1" permite auxiliar as descrições dos relatórios.

```
ALTER proc [dbo].[tabela1]
@Especialidade nvarchar (max)
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'tespecialConce=InputDataSet
OutputDataSet = data.frame(tespecialConce);'
, @input_data_1 = N'select concelhos.concelho,count([GPSDADOS].[dbo].[Concelhos].CONCELHO) as Total
from [GPSDADOS].[dbo].[Concelhos]
inner join [GPSDADOS].[dbo].[Dados] on [GPSDADOS].[dbo].[Dados].CONCELHO=[GPSDADOS].[dbo].[Concelhos].CONCELHO
where data between ''2016-01-01'' and ''2016-12-31'' and ESPECIAL like @e
group by lon,lat,[GPSDADOS].[dbo].[Concelhos].CONCELHO, id
order by [GPSDADOS].[dbo].[Concelhos].CONCELHO asc'
,@parallel = 1
,@params= N'@e nvarchar(max)'
,@e=@Especialidade

with result sets ((Concelhos varchar (max), Total nvarchar (max)));
end
```

Figura 21 - Código do stored procedure "tabela1"

O stored procedure "tabela2" permite auxiliar as descrições dos relatórios.

```
ALTER proc [dbo].[tabela2]
@DataInicial date,
@DataFinal date
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'tprestador=InputDataSet
OutputDataSet = data.frame(tprestador);'
, @input_data_1 = N'select DISTRITO, count(DISTRITO) as [Qt]
from(select distinct codprest, [GPSDADOS].[dbo].[Dados].DISTRITO, Lon, Lat
FROM [GPSDADOS].[dbo].[Dados]
inner join [GPSDADOS].[dbo].[Distritos] on [GPSDADOS].[dbo].[Dados].DISTRITO=[GPSDADOS].[dbo].[Distritos].Distrito
where data between @a and @b) as subquery
group by DISTRITO
order by DISTRITO'
,@parallel = 1
,@params= N'@a date, @b date'
,@a=@DataInicial
,@b=@DataFinal

with result sets ((Distritos varchar (max), Total nvarchar (max)));
end
```

Figura 22 - Código do stored procedure "tabela2"

O stored procedure "tabela3" permite auxiliar as descrições dos relatórios.

```
ALTER proc [dbo].[tabela3]
@Especialidade nvarchar (max),
@DataInicial date,
@DataFinal date
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'tatosDist=InputDataSet
OutputDataSet = data.frame(tatosDist);'
, @input_data_1 = N'select [GPSDADOS].[dbo].[Dados].[codigo distrito] as cod,[GPSDADOS].[dbo].[Distritos].Distrito, count([GPSDADOS].[dbo].[Distritos].Distrito) as Qt
from [GPSDADOS].[dbo].[Dados]
inner join [GPSDADOS].[dbo].[Distritos] on [GPSDADOS].[dbo].[Distritos].(Distrito)=[GPSDADOS].[dbo].[Dados].(DISTRITO)
where [GPSDADOS].[dbo].[Dados].data between @b and @c AND ESPECIAL like @e
group by [GPSDADOS].[dbo].[Dados].[codigo distrito], [GPSDADOS].[dbo].[Distritos].Distrito
ORDER BY [GPSDADOS].[dbo].[Dados].[codigo distrito]'
,@parallel = 1
,@params= N'@e nvarchar(max), @b date, @c date'
,@e=@Especialidade
,@b=@DataInicial
,@c=@DataFinal

with result sets ((ID nvarchar (max),Distritos varchar (max), Total nvarchar (max)));
end
```

Figura 23 - Código do stored procedure "tabela3"

O *stored procedure* "tabeladescr" permite auxiliar as descrições dos relatórios.

```
ALTER proc [dbo].[tabeladescr]
@Especialidade nvarchar (max)
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'descricao=InputDataSet
OutputDataSet = data.frame(descricao);'
, @input_data_1 = N'SELECT DISTINCT DESCRICA, count(DESCRICA) as Total
FROM Dados
where ESPECIAL LIKE @a
group by DESCRICA
ORDER BY DESCRICA'
, @parallel = 1
, @params= N'@a nvarchar(max)'
, @a=@Especialidade
with result sets (([Tipos de atos medicos] nvarchar (max), Total nvarchar (max)));
end
```

Figura 24 - Código do stored procedure "tabeladescr"

O *stored procedure* "tabelaout" permite auxiliar as descrições dos relatórios.

```
ALTER proc [dbo].[tabelaout]
@Especialidade nvarchar (max),
@Distrito nvarchar (max),
@AnoInicial int,
@AnoFinal int
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(scales)
library(dplyr)
library(ggthemes)
library(forecast)
racio=InputDataSet
racio=filter(racio, racio$ano>(c-1) & racio$ano<=(d+1))
racio$racio = racio$Qt/racio$Total

if (nrow(racio)>0){
for(j in 0:(max(racio$ano)-min(racio$ano))){
for(i in 1:nrow(racio)){
if(racio$ano[i] == min(racio$ano)+j){racio$meses[i]=racio$Mes[i]+12*j}
}
}
meses=1:(max(racio$ano)-min(racio$ano)+1)*12
meses=data.frame(meses)
e=left_join(meses, racio, by="meses")
dados=data.frame(e$meses, e$racio)
colnames(dados)=c("meses", "racio")

for(i in 1:nrow(dados)){
if(is.na(dados$racio[i])!=T){
dados$racio[i]=0
}
}
if(nrow(dados)>0){
dados = ts(dados$racio, frequency = 12, start = c(c, 1), end= c(d,12))
Q1=quantile(dados, probs = 0.25)
Q3=quantile(dados, probs = 0.75)
iqr=Q3-Q1
low=Q1-1.5*iqr
high=Q3+1.5*iqr
df = data.frame(as.numeric(low), as.numeric(high))
colnames(df)=c("Low", "High")} else if (nrow(dados)<1) {} else if (nrow(racio)<1) {}
df = data.frame("Nada a apresentar", "Nada a apresentar")
colnames(df)=c("Low", "High")}
OutputDataSet = data.frame(df);'
, @input_data_1 = N'SELECT sub1.ano, sub1.Mes, Total, Qt
FROM (SELECT ano, Mes, COUNT(codprest) Total
FROM (SELECT distinct codprest, MONTH(data) as Mes, year(data) as ano
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a) SUB1
GROUP BY ano, Mes) SUB1
INNER JOIN (SELECT MONTH(DATA) Mes, year(data) as ano, COUNT(DISTRITO) Qt
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a
GROUP BY DISTRITO, YEAR (DATA), MONTH(DATA)) SUB2 ON SUB2.ano=SUB1.ano and SUB2.mes=SUB1.mes
Order by sub1.ano, sub1.Mes'
, @parallel = 1
, @params= N'@a nvarchar(max), @b nvarchar(max), @c int, @d int'
, @a=@Especialidade
, @b=@Distrito
, @c=@AnoInicial
, @d=@AnoFinal
with result sets ((Low nvarchar (max), High nvarchar (max)));
end
```

Figura 25 - Código do stored procedure "tabelaout"

O *stored procedure* "conclusao" calcula o menor erro quadrático médio dos métodos de previsão utilizados, consoante o resultado é apresentada uma mensagem de conclusão.

```

ALTER proc [dbo].[conclusao]
@Especialidade nvarchar (max),
@Distrito nvarchar (max),
@AnoInicial int,
@AnoFinal int
as
begin
exec sp_execute_external_script
@language = N'R'
, @script = N'library(ggplot2)
library(scales)
library(dplyr)
library(ggthemes)
library(forecast)
racio=InputDataSet
racio=filter(racio,racio$ano>(c-1) & racio$ano<(d+1))
racio$racio = racio$Qt/racio$Total

if (nrow(racio)>0){
for(j in 0:(max(racio$ano)-min(racio$ano))){
for(i in 1:nrow(racio)){
if(racio$ano[i] == min(racio$ano)+j){racio$meses[i]=racio$Mes[i]+12*j}
}
}
meses=1:(max(racio$ano)-min(racio$ano)+1)*12)
meses=Data.frame(meses)
e=left_join(meses,racio,by="meses")
dados=Data.frame(e$meses,e$racio)
colnames(dados)=c("meses","racio")

for(i in 1:nrow(dados)){
if(is.na(dados$racio[i])!=T){
dados$racio[i]=0
}}

if(nrow(dados)>0){
dados = ts(dados$racio, frequency = 12, start = c(c, 1), end= c(d,12))
HW = HoltWinters(dados);
HWforecast = forecast(HW, h=24);
ETS = ets(dados);
ETSforecast = forecast(ETS, h=24);
EHW = accuracy(HWforecast$fitted,dados)
EETS = summary(ETS)

(if(EHW[5]<EETS[5]){
x="Observando os erros absolutos medios em percentagem, podemos concluir que o melhor modelo segundo este criterio (o menor), e o modelo: Holt-Winters"
})
else if(EETS[5]<EHW[5]){
x="Observando os erros absolutos medios em percentagem, podemos concluir que o melhor modelo segundo este criterio (o menor), e o modelo: Alisamento Exponencial"
})
} else if (nrow(dados)<1) {x = "Nao existem dados"
}} else if (nrow(racio)<1) {x = "Nao existem dados"
}}
OutputDataSet = Data.frame(x);'
, @input_data_1 = N'SELECT sub1.ano,sub1.Mes, Total,Qt
FROM(SELECT ano,Mes, COUNT(codprest) Total
FROM (SELECT distinct codprest,MONTH(data) as Mes,year(data) as ano
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a) SUB
GROUP BY ano,Mes) SUB1
INNER JOIN (SELECT MONTH(DATA) Mes,year(data) as ano,COUNT(DISTRITO) Qt
FROM GPSDADOS.DBO.Dados
WHERE DISTRITO LIKE @b and Especial like @a
GROUP BY DISTRITO, YEAR (DATA), MONTH(DATA)) SUB2 ON SUB2.ano=SUB1.ano and SUB2.mes=SUB1.mes
Order by sub1.ano,sub1.Mes'
, @parallel = 1
, @params= N'@a nvarchar(max), @b nvarchar(max), @c int, @d int'
,@a=@Especialidade
,@b=@Distrito
,@c=@AnoInicial
,@d=@AnoFinal
with result sets ((Conclusao nvarchar(max)));
end

```

Figura 26 - Código do stored procedure "conclusao"

Anexo II - Relatórios

O relatório "NrPrestadoresDist" apresenta o número de prestadores de saúde por distrito num período de tempo possível à escolha, com dois parâmetros à escolha, a data inicial e a data final. O *dataset* é composto pelos *stored proceures* "prestador" e "tabela2".

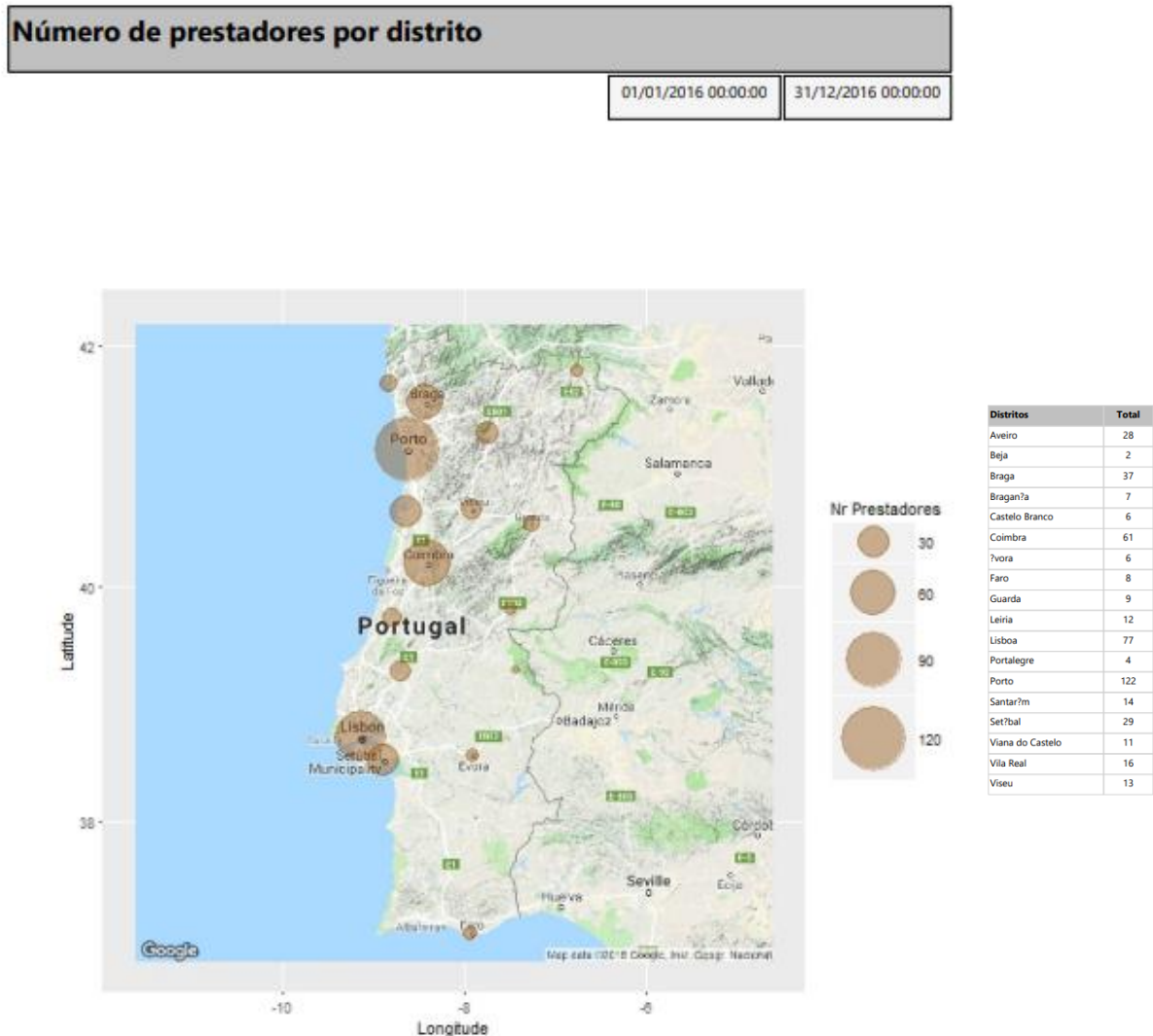


Figura 27 - Output do relatório "NrPrestadoresDist"

O relatório "NrAtosConcelho" apresenta o número de atos médicos por cada concelho conforme a especialidade escolhida. O *dataset* é composto pelo *stored procedure* "oftalmologia".

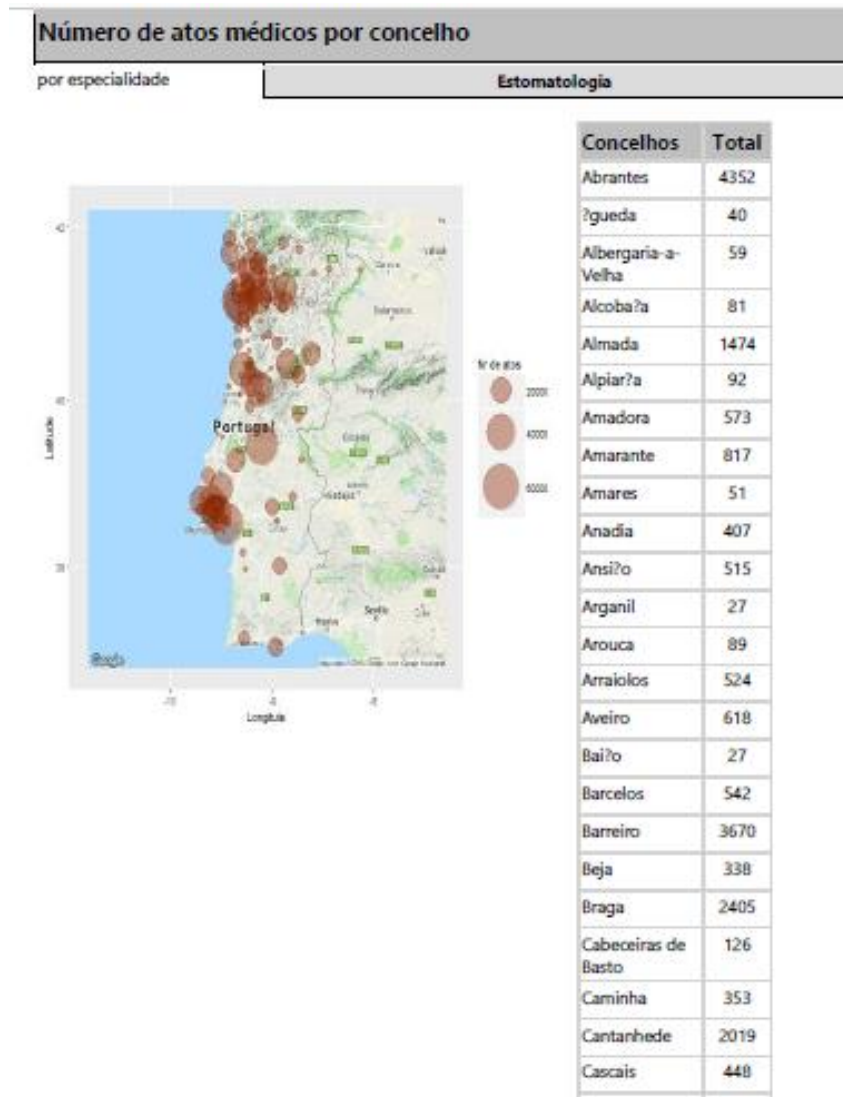


Figura 28 - Output do relatório "Nr Atos Concelho"

O relatório "Concelhos" apresenta o mapa de atos médicos em cada concelho do distrito e especialidade pretendida num período de tempo específico. O *dataset* é composto pelos *stored procedures* "concelho" e "tabela".

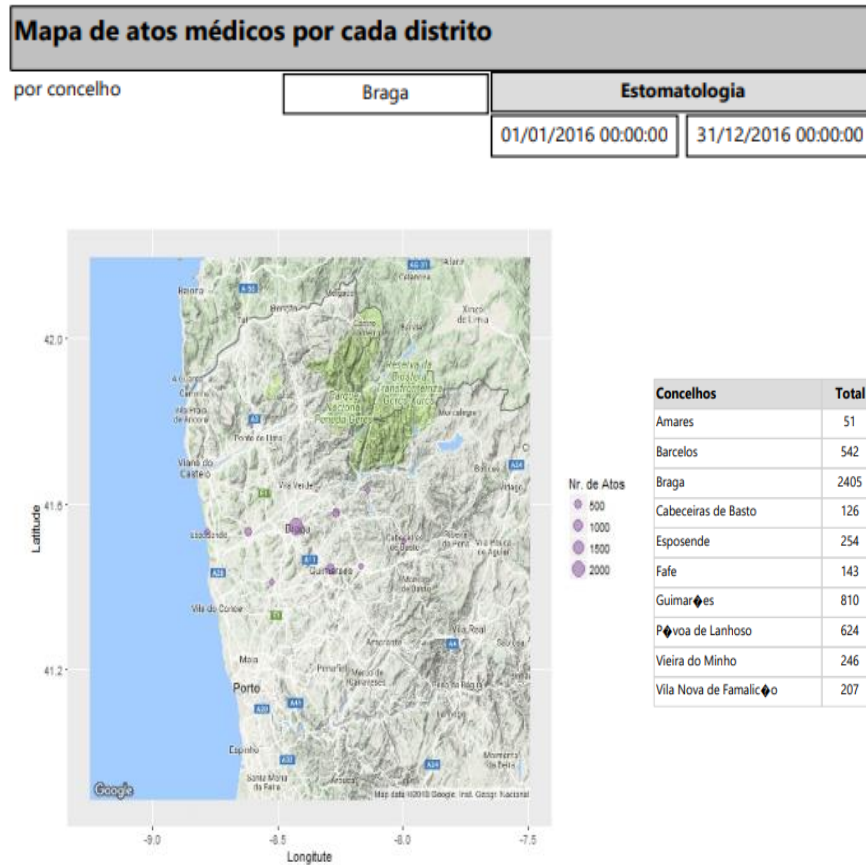


Figura 29 - Output do relatório "Concelhos"

O relatório "Registo Mensal" apresenta o registo mensal dos atos médicos durante um ano, sobre uma especialidade e um distrito. O *dataset* é composto pelo *stored procedure* "registo".

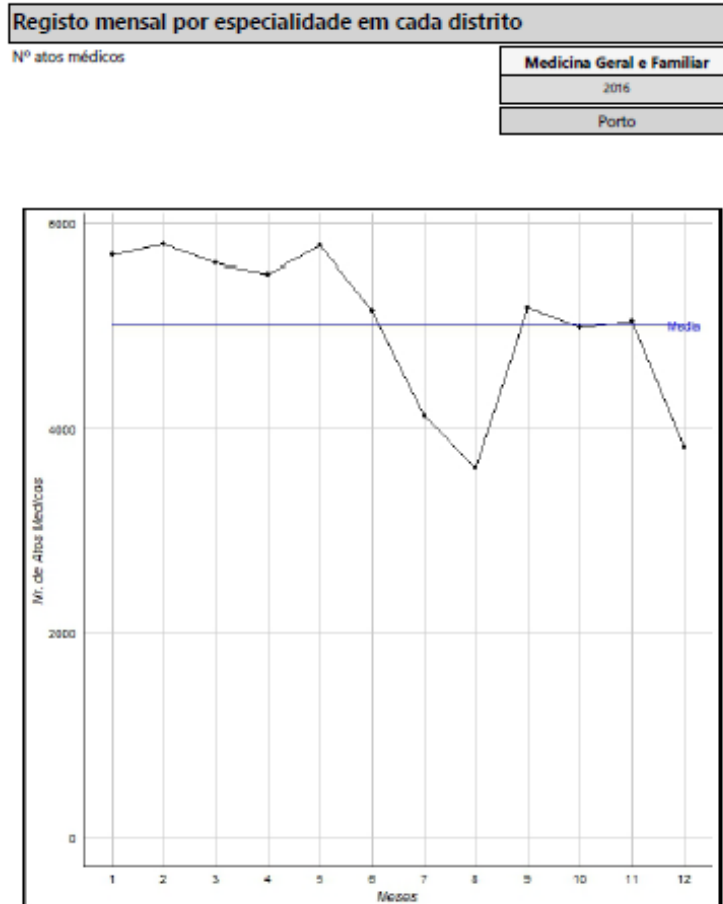


Figura 30 - Output do relatório "Registo Mensal"

O relatório "Descrição" apresenta a lista de cada tipo de ato médico conforme a especialidade escolhida. O *dataset* é composto pelo *stored procedure* "tabeladesc".

Resumo de atos médicos por cada Especialidade	
Medicina Geral e Familiar	
Tipos de atos medicos	Total
Ultrassonangiografia Carotidea E Vertebral	997
Estomago E Duodeno Com Terendlembourg	98
Tempo de protrombina (TP, Quick, INR)	9794
Massagem Manual Local	97
Electromiograma (EMG)	96
TC dos ouvidos	95
Rast Test.IgE Especifica P/dete.alergenio (ria Ou	949
Ambos os joelhos em carga antero-posterior	94
Dehidroepiandrosterona Sulfato = Dhea So4	94
Eosinofilos Pesquisa	92
Calcaneo - 2 Incidencias	91
Coluna Coccigea - 2 Incidencias	90
00111009	9
Anca unilateral em carga, uma incidencia	9
Anticorpos para Herpes simplex I- IgG+IgM	9
Anticorpos para Herpes simplex II- IgG+IgM	9
Articulaçao acromio-clavicular, bilateral	9
Colecistectomia Com EsfincteroSplastia	9
Ecografia Prostatica Trans-rectal	9
Electrodiagnostico Multiplo (varias Regioes)	9
Electromagnetismo	9
Estimpanom.func.trompa Eustaquio (med.c/ponte Ad	9

Figura 31 - Output do relatório "Descrição"

O relatório "Rácio" apresenta os dados da fórmula rácio e tem como parâmetros um distrito, uma especialidade, um ano 1 e um ano 2. O *dataset* é composto pelo *stored procedure* "teste1". O gráfico apresenta ainda a média anual dos respetivos anos.

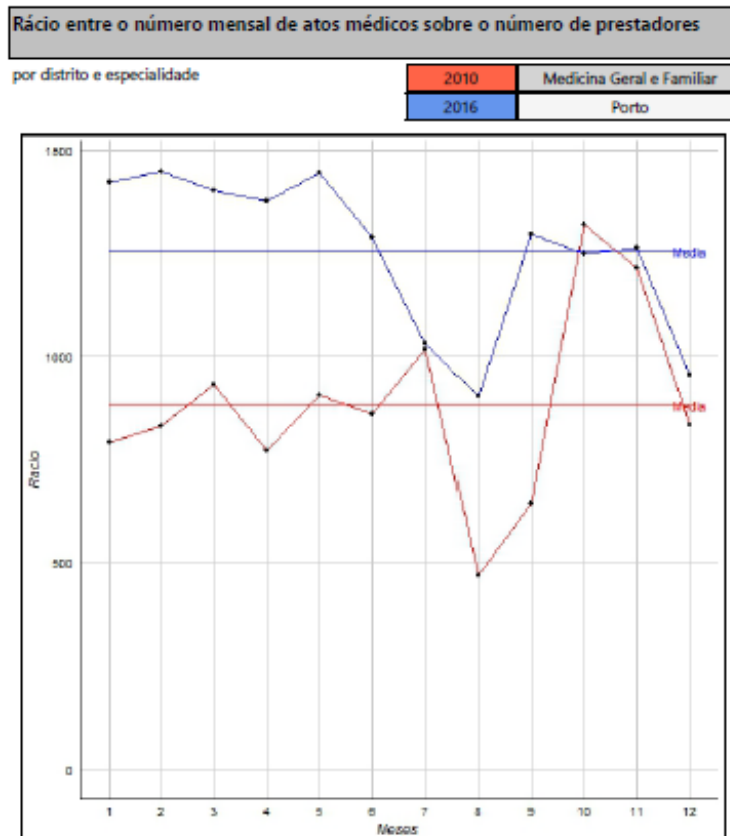


Figura 32 - Output do relatório "Rácio"

O relatório "Série Temporal" apresenta os dados do rácio numa série sobre uma especialidade e um distrito. O *dataset* é composto pelo *stored procedure* "st".

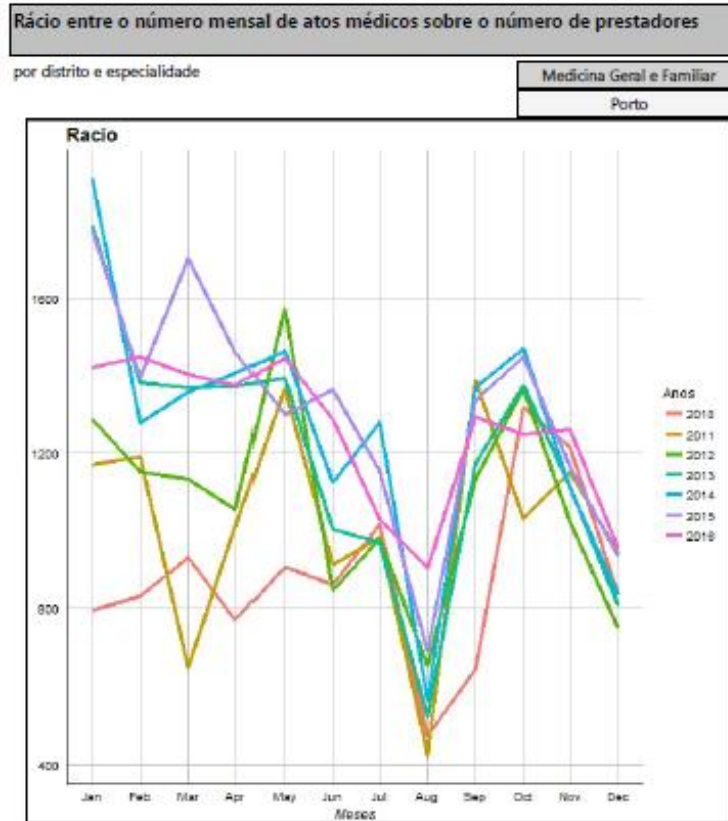


Figura 33 - Output do relatório "Série Temporal"

O relatório "Decomposição" decompõe os dados reais nas componentes de uma série temporal. Tem como parâmetros o distrito, a especialidade, o ano inicial e o ano final. O *dataset* é composto pelo *stored procedure* "decompose".

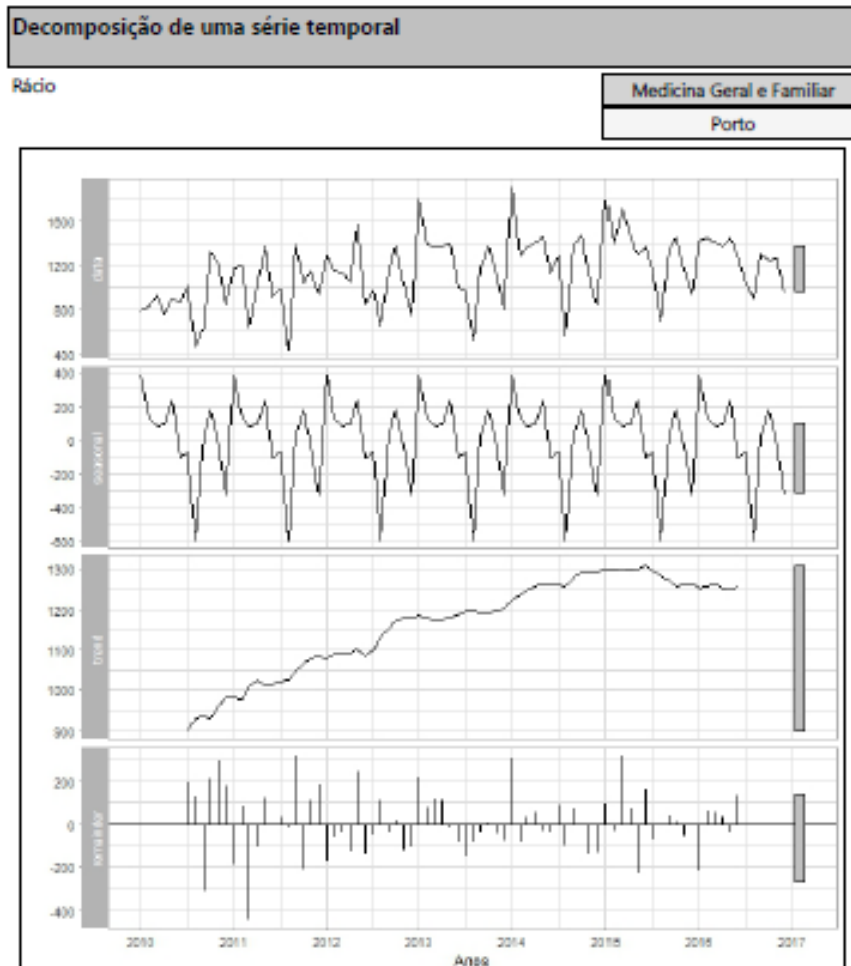


Figura 34 - Output do relatório "Decomposição"

O relatório "Outlier" revela o limite superior e inferior para análise da série do rácio. A sua parametrização é constituída pela especialidade e pelo distrito. O *dataset* é composto pelo *stored procedure* "outlier" e "tabelaout".

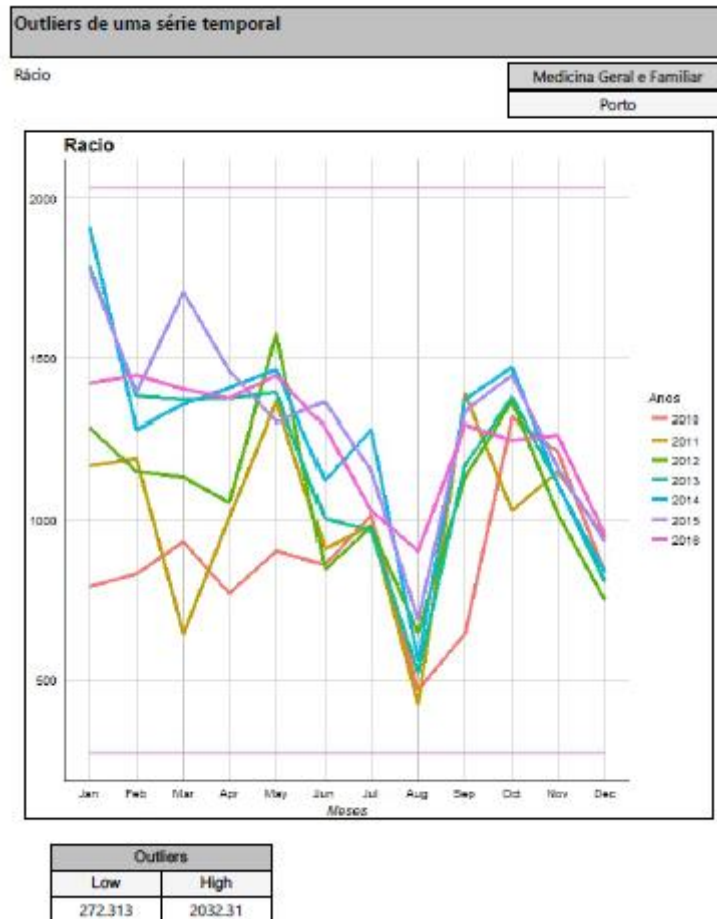


Figura 35 - Output do relatório "Outlier"

O relatório "Previsão AE" prevê os valores futuros de uma série temporal. Tem como parâmetros o distrito, a especialidade, o ano inicial e o ano final. O *dataset* é composto pelo *stored procedure* "alisamentoex" e "conclusao".

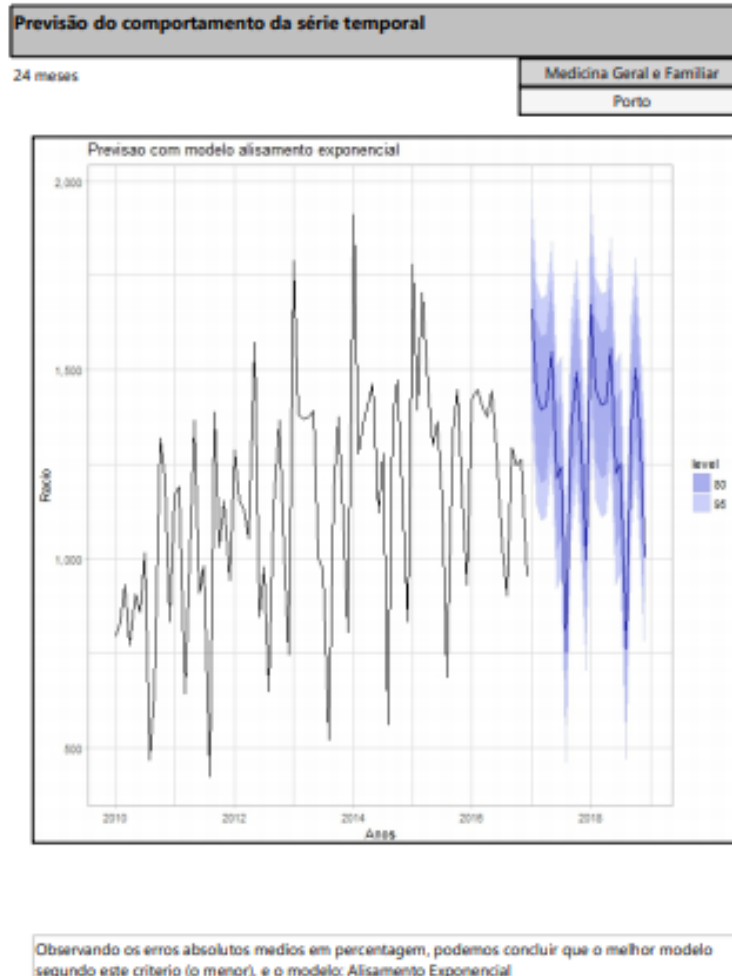


Figura 36 - Output do relatório "Previsão AE"

O relatório "Previsão HW" prevê os valores futuros de uma série temporal. Tem como parâmetros o distrito, a especialidade, o ano inicial e o ano final. O *dataset* é composto pelo *stored procedure* "previsao" e "conclusao".

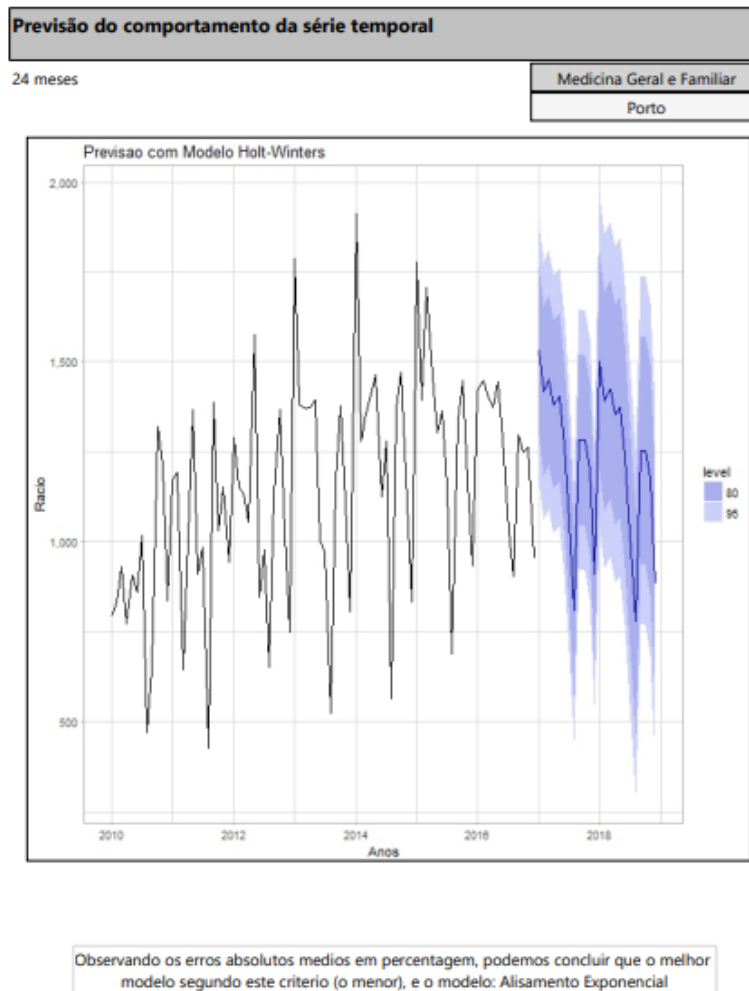


Figura 37 - Output do relatório "Previsao HW"