



**Instituto Superior de Economia e Gestão**

UNIVERSIDADE TÉCNICA DE LISBOA

Mestrado em Decisão Económica e Empresarial

Relatório de estágio:

**“Análise dos tempos de resolução de incidentes no desenvolvimento de software”**

Ana Maria Pereira da Silva Filipe

Orientadores: Professora Doutora Nicoletta Rosati  
Professor Doutor João Paulo Carvalho

Presidente: Professor Doutor António Costa

Vogal: Professor Doutor José Passos

Setembro de 2011

## **Agradecimentos**

Aos meus pais e familiares pelo seu apoio constante.

Aos meus amigos pelo seu apoio prestado.

À Professora Doutora Nicoletta Rosati e ao Professor Doutor João Paulo Carvalho pelas excelentes ideias na orientação deste relatório e pela disponibilidade sempre demonstrada.

Aos meus colegas do estágio, que me ajudaram neste trabalho.

# Índice

<u>Agradecimentos</u>	<u>1</u>
<u>Índice</u>	<u>2</u>
<u>Introdução</u>	<u>3</u>
<u>1 – Apresentação da empresa <i>Quidgest</i></u>	<u>5</u>
<u>2 – Apresentação do estágio</u>	<u>6</u>
<u>3 – Formação realizada no estágio</u>	<u>7</u>
<u>4 – Actividades desenvolvidas no estágio</u>	<u>10</u>
4.1 – Introdução ao SQL	10
4.2 – Caracterização das variáveis que serviram de base o presente estudo	11
4.3 – Construção da base de dados	12
<u>5 – Análise dos dados</u>	<u>15</u>
5.1 – Estatísticas descritivas	15
5.2 – Análise gráfica	16
<u>6 – Apresentação dos modelos estatísticos</u>	<u>27</u>
6.1 – Introdução	27
6.2 – Modelo de Kaplan-Meier	28
<u>7 – Apresentação dos resultados de estimação</u>	<u>29</u>
7.1 – Análise de sobrevivência da variável “dias”	29
7.2 – Análise de sobrevivência com a variável “dias” como variável dependente e a variável “prioridade” ou “ano” como variável explicativa	30
7.3 – Análise de sobrevivência da variável “horas”	33
7.4 – Cálculo da função <i>hazard</i> para 2011, por prioridade	34
<u>8 – Conclusões</u>	<u>35</u>
<u>9 – Sugestões para trabalhos futuros</u>	<u>36</u>
<u>10 – Bibliografia</u>	<u>37</u>

## Introdução

O presente relatório tem como objectivo descrever as actividades realizadas durante o estágio de natureza profissional efectuado na empresa *Quidgest*, localizada em Lisboa, da área informática. As actividades foram exercidas no período de 1 de Fevereiro de 2011 a 30 de Junho de 2011.

A produção de *software* tem aumentado muito nos últimos anos devido à sua grande importância na sociedade contemporânea. Este desenvolvimento deve-se ao facto de o uso dos computadores ter aumentado bastante, como também as diversas áreas do conhecimento humano. Estes dois aumentos têm gerado uma crescente procura por soluções que automatizem os diversos processos.

Com o objectivo de melhorar a qualidade dos *softwares* e aumentar a produtividade no desenvolvimento de tais produtos surgiu a engenharia do software.

Com o desenvolvimento das novas técnicas de produção de *software*, este tem-se tornado cada vez mais importante, permitindo a informatização da maioria das actividades económicas.

A engenharia do *software* propõe métodos sistemáticos com o uso adequado de ferramentas e técnicas, que levam em consideração o problema a ser resolvido, as necessidades dos clientes e os recursos que estão disponíveis. O *software* tem a ver sobretudo com a matemática e a computação. É através destas disciplinas que é possível o desenvolvimento de *softwares* mais fiáveis, de melhor qualidade, com custo reduzido e alta produtividade. Sem dúvida que a qualidade e a produtividade levam a que os programadores desenvolvam cada vez mais os programas de *software* com o objectivo de tornarem a produção e a entrega do produto (*software*) mais céleres.

A qualidade dos produtos e dos processos são de extrema importância. A produtividade é um bom indicador para o progresso tanto a nível económico como a nível tecnológico de uma empresa. Através do crescimento continuado da produtividade, a competitividade pode aumentar. O aumento da produtividade não significa necessariamente aumento da competitividade. A produtividade tem a ver sobretudo com a produção de bens independentemente da qualidade e da inovação, enquanto que a competitividade está relacionada com a posição vantajosa ou não da empresa no mercado. Uma empresa pode estar a produzir produtos (aumento da produtividade) mas serem de má qualidade, logo a competitividade não

aumenta uma vez que há empresas que produzem os mesmos produtos que aquela empresa mas com a diferença da qualidade.

A produtividade no desenvolvimento do *software* pode apresentar inúmeras diferenças de empresa para empresa, na medida em que cada empresa pode desenvolver software em prazos mais pequenos do que outras dependendo de alguns factores ligados ao mecanismo de funcionamento da própria empresa como, por exemplo, o nível de experiência dos técnicos.

A experiência ganha no processo de resolução dos incidentes tem um papel fundamental na diminuição do número de incidentes pois os técnicos actuam directamente na solução e detecção dos problemas relacionados, neste caso, com o *software*.

Neste trabalho serão relatadas as actividades desenvolvidas durante o estágio.

Numa primeira abordagem, houve lugar à aprendizagem do *SQL – Structured Query language*, análise e compreensão do trabalho a realizar. O programa SQL é utilizado por profissionais de informática e é usado na *Quidgest* principalmente para resolver incidentes ou para detectar eventuais erros nos programas desenvolvidos pela empresa.

Em termos gerais, o objectivo deste relatório é estudar o tempo decorrido entre o início e o fim do processo de resolução do incidente.

Foram analisados dados reais fornecidos pela empresa relativamente aos anos de 2006 até 2011, perfazendo no total 3133 incidentes.

No primeiro capítulo será feita uma apresentação da empresa e no segundo capítulo uma apresentação do estágio.

O capítulo três descreve a formação recebida durante o estágio.

O capítulo quatro menciona as actividades desenvolvidas no estágio, nomeadamente, as características das variáveis em estudo e a construção da base de dados que servirá para a análise.

O capítulo cinco descreve a análise dos dados.

Os capítulos seis e sete prendem-se com a apresentação dos modelos estatísticos teórica e prática, respectivamente.

Os capítulos oito e nove apresentam as conclusões e sugestões para trabalhos futuros, respectivamente.

## **1 – Apresentação da empresa Quidgest**

A empresa *Quidgest* é uma empresa de consultoria e desenvolvimento de sistemas de informação de gestão que aposta na investigação em Engenharia do *software*.

A *Quidgest* é o maior produtor de *software* empresarial de origem nacional, pela variedade de soluções produzidas.

Com empresas já constituídas em Espanha, Reino Unido, Timor-Leste e Moçambique, a *Quidgest* tem investido com grande sucesso na internacionalização das suas actividades.

A empresa desenvolve Sistemas de Informação desde 1988 e em Portugal foi pioneira na informatização da Administração Pública. O seu sistema de gestão SINGAP (Sistema Integrado para a Nova Gestão da Administração Pública) é, hoje em dia, utilizado por mais de metade dos organismos públicos da Administração Central do Estado. A *Quidgest* trabalha com grandes empresas e organismos da Administração Pública e lidera a inovação na produção de *software*, tanto a nível nacional como internacional.

A *Quidgest* foi também pioneira na racionalização da prestação de cuidados de saúde, com a realização da prescrição electrónica de medicamentos da Portugal Telecom – Associação de Cuidados de Saúde.

Os sistemas de informação definem, actualmente, a forma de organização dos organismos. Eles têm a grande vantagem de poderem aumentar significativamente a produtividade dos recursos humanos que estão envolvidos na empresa, garantir o cumprimento das regras e medir a eficiência dos recursos usados, testar a validade das normas face aos procedimentos em curso e transmitir de uma forma rápida novas orientações de gestão.

A empresa tem actualmente cerca de 80 colaboradores. As suas principais áreas são a área de Investigação & Desenvolvimento, a área de Marketing & Comunicação e varias áreas de sistemas de informação, como sendo, Gestão Patrimonial e Aprovisionamento, Recursos Humanos, Gestão Financeira e Contabilidade, Gestão Documental e Projectos Especiais.

O *software* é produzido utilizando varias linguagens de programação, entre as quais, Genio, Visual C++, C#, JAVA, XML, SQL Server, SGBD's.

## **2 – Apresentação do estágio**

O objectivo do estágio é determinar o *SLA* (*Service Level Agreement*), ou seja, determinar o tempo decorrido entre o início e o fim do processo de resolução de um incidente. É importante ficar claro que nem sempre o técnico inicia a resolução do incidente no mesmo dia em que o cliente fez a reclamação. Um *SLA* é negociado entre duas partes onde uma é o cliente e a outra é o prestador de serviço. No nosso caso, o acordo é feito entre o cliente onde o incidente está a ocorrer e os técnicos responsáveis pelo incidente. O primeiro passo para atingir este objectivo é determinar o tempo que o processo de resolução do incidente demora até ser resolvido utilizando modelos de duração. Resolver os incidentes até à data acordada com o cliente nem sempre é fácil, por isso, deve haver uma monitorização dos tempos de resposta de cada interveniente. Deste modo, os prazos contratuais têm de ser cumpridos.

O principal objectivo em analisar os tempos de resposta dos incidentes é minimizar o impacto do incidente no cliente e permitir o bom funcionamento dos sistemas o mais depressa possível.

A gestão de um projecto deve incluir o cumprimento dos prazos previstos e a minimização de recursos (financeiros, mão-de-obra, equipamento), controlando-se a variância do tempo de resolução do incidente.

Os tempos de resolução são registados em dias, mas podemos representá-los como uma distribuição contínua, uma vez que por natureza o tempo é uma variável aleatória contínua.

### **3 – Formação realizada no estágio**

Nas semanas de 16 de Maio a 30 de Maio de 2011 realizou-se nas instalações da *Quidgest* uma formação de Genio na qual eu estive presente.

Esta formação foi destinada a todos os novos colaboradores tendo como objectivo a aproximação do desenvolvimento do Genio aos colaboradores.

O Genio da *Quidgest* é uma plataforma que gera automaticamente código para construir aplicações para diferentes ambientes e interfaces.

As interfaces do Genio são: *Windows*, *Web Services*, *Portais Web*, *Dispositivos Móveis*, *Office Add-ins*.

O Genio gera código em diversas linguagens de programação (JAVA, C++ e C#) e trabalha com os principais Sistemas de Gestão de Bases de Dados, como sejam, *SQL Server*, *Oracle* e *DB2*.

Com este tipo de tecnologia, o desenvolvimento de sistemas de informação funciona como um jogo de LEGO, ou seja, se se juntar todas as peças disponíveis é exequível construir qualquer tipo de sistema.

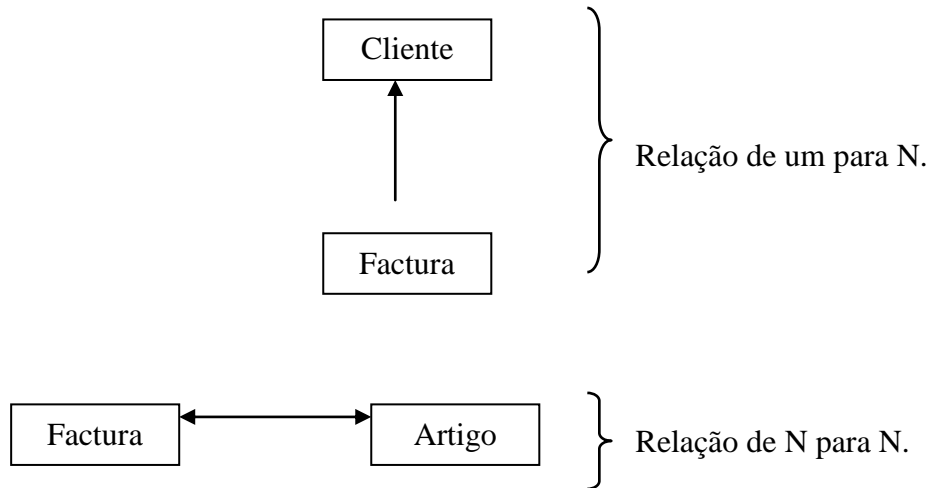
O primeiro dia foi sobre as funcionalidades básicas do Genio. Nesta sessão falou-se sobre os requisitos necessários para a instalação do Genio, a arquitectura, configuração.

O segundo dia foi sobre o modelo relacional da base de dados. Em termos gerais, o modelo relacional, como o próprio nome indica, estabelece relações entre tabelas uma vez que os dados estão organizados em tabelas facilitando assim a consulta dos mesmos. No Genio utiliza-se as relações de um para um (1:1), de um para muitos (1:N) e de muitos para muitos (N:N). A relação de um para um (1:1) raramente é usada. A relação (1:N) significa que para um registo de uma tabela existe muitos registos de outra tabela. Um exemplo desta relação é o caso de um determinado cliente ao qual estejam associadas várias facturas. Por fim, a relação de muitos para muitos (N:N) significa que para um registo de uma tabela X correspondem N registos de uma tabela Y e a um registo da tabela Y correspondem N registos da tabela X. Quando existe esta relação recorre-se à relação de (1:N), ou seja, cria-se uma nova tabela que integra as duas fontes de informação.

No terceiro dia fez-se uma introdução à linguagem de programação *Web*. Numa primeira abordagem falou-se sobre interface, *Qweb*, metadados e comunicação-servidor. Como já foi dito anteriormente, o Genio gera código para construir aplicações para diversas interfaces. O Genio



pode gerar em três *layouts* padrão diferentes, a saber: menus laterais, menus horizontais e renderização com recurso a *jQuery*.



No quarto e quinto dias falou-se sobre formulários e menus. Quando se cria uma tabela, o Genio associa um formulário a essa tabela, perguntando se se pretende criar esse formulário ou não. Apesar de a plataforma criar um formulário associado a uma tabela, podem ser criados tantos formulários quantos os que se quiserem. Os menus dão acesso às funcionalidades da aplicação e estão definidos de forma hierárquica.

O sexto dia foi dedicado às fórmulas. As fórmulas internas servem para criar cálculos sem haver a necessidade de se utilizar as rotinas manuais. Basta ir buscar os valores necessários para que o Genio faça os cálculos.

No sétimo dia discutiram-se os níveis de acesso, as entradas permanentes de historial (EPH's) e a autenticação. Os níveis de acesso como o próprio nome indica definem uma hierarquia quando se pretende aceder a tabelas, *forms*, menus, campos. A partir do momento em que o utilizador faz o *login*, o Genio automaticamente adiciona o nome do utilizador ao historial, atribuindo-lhe certos registos. A autenticação é um dos pontos fulcrais de uma qualquer aplicação na medida em que a segurança para o utilizador ao fazer o *login* tem de ser garantida.

No oitavo dia falou-se sobre as rotinas manuais em *PL/SQL*, na arquitectura e no *Web services*.

No nono e último dias falou-se sobre as rotinas manuais para o *QWeb* e sobre a arquitectura.

Em cada uma das sessões houve lugar a um exemplo com o objectivo de se aplicar o que se aprendeu nas respectivas sessões. Com esse exemplo pretendia criar-se numa página Web um

inquérito, de modo a que as pessoas pudessem preenchê-lo nessa mesma página de uma forma mais célere. Todo este procedimento faz-se através da plataforma Genio.

## **4 – Actividades desenvolvidas no estágio**

### **4.1 – Introdução ao SQL**

O SQL é uma linguagem padrão para base de dados. A importância do SQL nos dias de hoje é indiscutível. A linguagem de programação está presente nos mais variados sistemas de informação.

O programa SQL tem por detrás o modelo relacional. Este modelo foi inventado em 1970 por Edgar Codd e foi utilizado pela maioria dos Sistemas de Geração de Base de Dados (SGBD). O modelo relacional é um modelo simples e intuitivo baseado no conceito matemático de relação. Há muitas operações que se podem utilizar para poder relacionar as várias tabelas. O programa SQL permite criar, alterar, remover todas as componentes de uma base de dados como por exemplo tabelas.

Nas Instituições é necessário e importante que a informação esteja organizada para que o acesso a esta seja o mais eficiente possível. A informação é armazenada em tabelas, pelo que este modelo é bastante útil uma vez que o objectivo é organizar os dados.

Um atributo ou lista de atributos pode ser declarado como chave em *SQL* utilizando chaves primárias e chaves estrangeiras. Nas chaves estrangeiras há um relacionamento entre tabelas enquanto que nas chaves primárias esse relacionamento não existe.

O nome do servidor onde está a informação usada para este relatório é o Jupiter, no entanto, a empresa utiliza vários servidores, nomeadamente, o *Antares*, o *Unuk*, o *Pulsar*, o *Perseus*, o *Felis*, o *Colossus*, o *Andromeda* e o *Allo*. O servidor Jupiter tem muitas bases de dados mas a que é mais útil para o presente relatório é a *QUI2011*. Esta base de dados é composta por diversas tabelas, designadamente, *quiincid*, *quitecni*, *quifactu*. Estas tabelas são constituídas por diversas colunas. A título de exemplo, a tabela *quiincid* tem a descrição do incidente, a tabela *quitecni* indica o nome de todos os técnicos que já passaram pela *Quidgest* e os que estão na empresa presentemente e a tabela *quifactu* dá-nos a indicação das facturas que são passadas aos clientes.

Para se poder fazer a análise dos tempos de resposta dos incidentes foi necessário fazer uma query à base de dados no programa SQL. A query é a seguinte:

```

select *, cast(encerradoEm-dataini as int) as dias from
  (select quiincid.priorida, quitecni.iniciais, quiarea.area,
    quiaccao.programa,
    quiincid.descrica, quiaccao.codincid, quiarea.sigla, round(sum(nhoras), 1)
    as horas, min(quiacao.data) as dataini
  from quiacao inner join quiincid on quiacao.codincid=quiincid.codincid
  left join quiarea on quiincid.codarea=quiarea.codarea
  left join quitecni on quiincid.codtecnt=quitecni.codtecni
  where quiincid.tptarefa='INCI' and quiincid.priorida <6
  group by quiincid.priorida, quitecni.iniciais, quiarea.area,
    quiacao.programa, quiincid.descrica, quiacao.codincid, quiarea.sigla) a
  inner join
  (select quiacao.codincid, min(quiacao.data) as encerradoEm
  from quiacao
  inner join quiincid on quiacao.codincid=quiincid.codincid
  where quiincid.tptarefa='INCI' and data is not null and
    quiacao.situacao in ('Fechado', 'Resolvido - avaliar eficácia')) b
  on a.codincid=b.codincid

```

Esta *query* é composta por duas subrotinas separadas com o *inner join* (o objectivo deste comando é concatenar os dados, ou seja, é apresentar as linhas que são comuns às tabelas *quiacao* e *quiincid*).

A primeira linha indica-nos o número de dias que cada incidente demorou até ser entregue, ou seja, a diferença entre a data de encerramento do incidente e a data do início da resolução do mesmo.

A primeira subrotina determina a data do início, a sigla, o código do incidente, o programa, a área, o técnico, a descrição, a prioridade e o número de horas. Esta *query* apresenta apenas os incidentes cuja prioridade é inferior a seis porque o tempo de entrega destes incidentes é urgente em relação aos incidentes cujo tempo de resposta seja superior a seis.

Para determinar o número de horas utilizou-se a função “*round*”. Esta função é usada para arredondar um campo numérico para um número de casas decimais específico. Neste caso, o número de horas vai aparecer com uma casa decimal. Quando se utiliza a função “*round*” é necessário usar a função de agregação “*sum*”.

#### **4.2 – Caracterização das variáveis que serviram de base no presente estudo:**

- Codincid: Código do incidente. Para cada incidente há um código diferente;
- Técnico: Nome do técnico responsável pela resolução do incidente;

- Técnico\_N: Atribuição de um número a cada técnico, por ordem alfabética;
- Sigla: É a sigla do programa/cliente;
- Sigla\_N: Atribuição de um número a cada sigla;
- Programa: Nome do programa em que se insere o incidente;
- Programa\_N: Atribuição de um número a cada programa, por ordem alfabética;
- Área: Área onde se insere o incidente;
- Descrição: Descrição técnica do problema específico a ser resolvido;
- Dias: Número de dias que o processo de resolução do incidente demorou até ser encerrado;
- Data\_ini: Data em que o cliente fez o pedido de resolução do incidente;
- Encerrado\_em: Data em que o processo do incidente foi encerrado;
- Prioridade: Atribuição de um nível de prioridade para cada incidente;
- Horas: Número de horas que efectivamente o incidente demorou a ser resolvido.

### **4.3 – Construção da base de dados:**

No início do estudo, a base de dados era constituída por 3193 incidentes.

Para construir a base de dados teve de se recorrer ao programa SQL, como já foi dito acima. Primeiramente, teve de se fazer um estudo sobre os comandos mais importantes para se poder ter todas as variáveis necessárias para se fazer a análise aos dados. De seguida, fez-se uma *query* que resultou nas seguintes variáveis: codincid, sigla, data\_ini, encerrado\_em, dias. As variáveis data\_ini e encerrado\_em foram convertidas em dia, mês e ano, para permitir uma análise mais aprofundada. Depois de analisar a *query* em questão detectou-se que esta informação era insuficiente para se poder fazer uma análise aos dados de uma forma mais profunda.

Deste modo, achou-se importante introduzir também as seguintes variáveis: programa, área, técnico responsável pelo incidente, descrição. Toda esta informação é importante para saber se há algum relacionamento entre variáveis, por exemplo saber se os técnicos com os tempos de resposta maiores são especializados em intervenções mais complexas. Como a cada programa pertence a uma e a uma só área as duas variáveis nunca são utilizadas ao mesmo tempo. A variável “descrição” tem um conteúdo informativo bastante grande, mas do ponto de vista

estatístico não foi possível utilizar a variável em questão, uma vez que não se conseguiu agrupar a mesma em categorias.

É importante salientar que durante o processo de resolução do incidente pode haver mais do que um técnico responsável pelo mesmo, na medida em que a resolução do incidente pode ser muito complexa.

Relativamente à variável “dias”, esta contém todo o processo de resolução do incidente incluindo os tempos mortos, como por exemplo o tempo de espera entre o pedido de resolução do incidente e o começo da mesma; a espera de *feedback* da parte do cliente; a eventual substituição do técnico que abriu o processo por outro. O valor “zero” nesta variável indica os processos que foram encerrados no próprio dia em que foi recebido o pedido. Geralmente, quando não têm feedback há mais de seis meses o técnico decide fechar o incidente.

Os incidentes têm de ser resolvidos o mais depressa possível de modo a que a empresa que detectou o erro não fique prejudicada. De facto, a gravidade de um incidente é medida segundo o impacto que o mesmo causa no processo de negócio de uma determinada instituição. Este tipo de problemas fez com que introduzíssemos uma outra variável, a prioridade. Como o próprio nome indica, esta variável diz-nos se a resolução do incidente é urgente ou não. Esta variável pode tomar um dos seguintes valores: zero, um, três, cinco, sete e nove, mas no nosso caso apenas se vai considerar os incidentes cuja “prioridade” toma os seguintes valores: zero, um, três e cinco. A empresa achou que não era relevante estudar os incidentes com prioridade igual a 7 e a 9, na medida em que estes incidentes não têm grande importância para a análise em questão.

A classificação zero na variável “prioridade” significa que ainda não foi atribuído um valor ao incidente. Se a variável “prioridade” toma o valor um, a sua resolução é muito urgente. Os incidentes com prioridade igual a três ou a cinco têm menos urgência para serem resolvidos.

Inicialmente, as observações com valor zero na variável “prioridade” tinham sido inseridas na base de dados mas após alguma reflexão achámos melhor tirá-las por duas razões:

- Primeira: se tivesse havido uma atribuição de um número à variável em causa, podia ser sete ou nove. Neste caso, estaríamos a incluir incidentes na base de dados que fogem à nomenclatura. Assim, poderíamos estar a enviar os resultados.
- Segunda: Como são apenas 53 incidentes que estão nesta situação, não faria sentido incluí-los na base de dados.

Depois de se fazer esta alteração, a nossa base de dados tem 3140 incidentes.

Retomando ainda a complexidade, introduziu-se na base de dados o número de horas com uma casa decimal para cada incidente. Esta variável apresenta apenas o tempo efectivo de resolução do incidente. É importante trabalhar com esta variável para se poder ter uma noção mais clara sobre a duração exacta que o incidente levou até ser resolvido.

Esta nova variável tem mais importância para a empresa do que propriamente para os clientes, ou seja, tem menos impacto para os clientes porque os clientes interessam-se apenas com o número de dias que o incidente demorou a ser resolvido.

Para se ver que com esta variável os resultados são mais precisos vejamos o caso de dois incidentes. Primeiro caso: para a variável “dias”, o processo de resolução de um incidente demorou zero dias e para a variável “horas” demorou 8 horas; Segundo caso: para a variável “dias”, o processo de resolução de um incidente demorou 51 dias e para a variável “horas” demorou apenas 2 horas. Isto acontece porque por vezes o processo de resolução do incidente pode ser aberto num determinado dia mas o técnico responsável pelo mesmo não começar a resolvê-lo nesse mesmo dia. Estas diferenças também têm a ver com o facto de haver incidentes que estão resolvidos mas o técnico está à espera de feedback por parte do cliente bastantes dias.

Depois de todas as variáveis relevantes estarem incluídas na base de dados, analisaram-se todos os incidentes com mais de 150 dias de duração para o fecho do processo, tendo-se detectado 25 situações em que o incidente foi fechado por não ter havido feedback há mais de seis meses. Esta análise teve como objectivo testar se efectivamente o incidente demorou ou não mais de 150 dias até ser resolvido. Dezoito casos não foram excluídos, mas alterou-se a data de encerramento para o dia em que o incidente ficou efectivamente resolvido, eliminando o tempo excedente à espera de *feedback*, portanto não foram excluídos da base de dados porque com a alteração, o número de dias ficou muito reduzido. Os restantes sete casos foram excluídos da base de dados por serem *outliers*.

Com estas situações pode concluir-se que o rigor no registo do fecho será na prática secundarizada em relação à resolução do incidente propriamente dita.

Assim sendo, a base de dados em estudo tem 3133 incidentes.

## 5 – Análise dos dados

### 5.1 – Estatísticas descritivas

Antes de começar com a estimação dos modelos de duração, vai proceder-se a uma análise gráfica dos dias que cada processo levou a ser encerrado para saber se há factores importantes para a determinação do tipo de modelação a usar, estatísticas descritivas e cruzamento de variáveis.

Quando os incidentes demoram mais tempo a serem resolvidos, isto pode ser devido ao facto do próprio problema ser difícil de resolver. Outro factor que pode explicar esta demora é o mês em que o incidente ocorreu, por exemplo, se um incidente ocorreu no mês de Agosto é provável que o técnico responsável pela resolução desse incidente esteja de férias, e que por conseguinte não consiga resolver o problema de imediato. Outro factor que também pode explicar o atraso na entrega da resolução é o facto de poder haver um parâmetro de heterogeneidade não observável entre incidentes, na medida em que pode haver incidentes que devido à sua complexidade tenham mais riscos associados do que outros relativamente à entrega dos mesmos. Pode haver vários técnicos responsáveis pela resolução do problema, uma vez que há incidentes muito complexos. Isto pode implicar também possíveis atrasos.

Segue-se uma breve descrição estatística em relação à variável “dias” (Murteira, Bento J. F. (1993)):

#### Univariate statistics

=====

Number of Observations: 3133

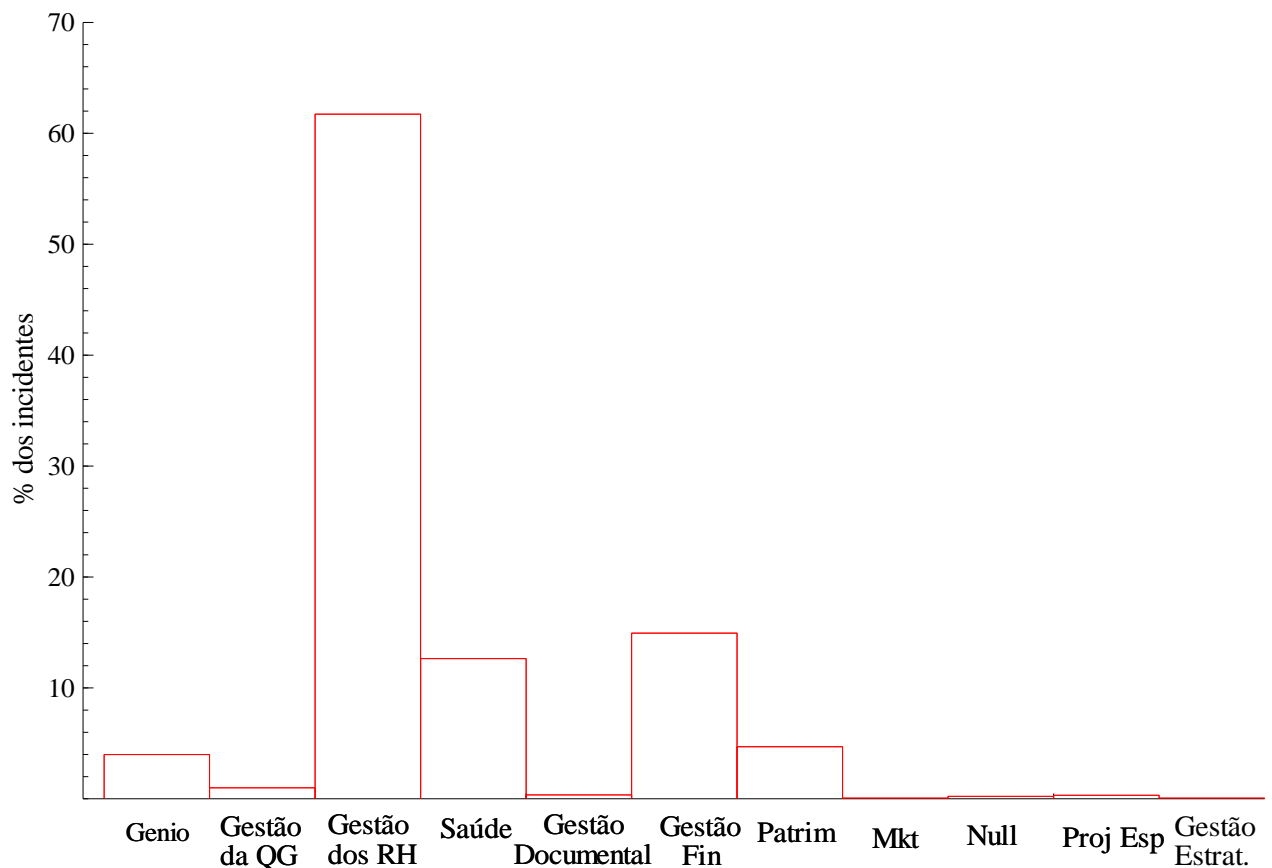
	Mean	Std Dev	Minimum	Maximum	Median
DIAS	9.00511	32.49243	0.00000	458.00000	0
	Sum	Variance	1º quartil	3º quartil	
DIAS	28213.00000	1055.75796	0	3	



Através da estatística descritiva pode dizer-se que, em média, a duração que um incidente demora a ser resolvido é aproximadamente 9 dias. Este resultado é bastante positivo uma vez que há incidentes que são mais complexos do que outros. O número de dias pode variar entre zero e 458 dias, como se pode constatar na informação acima. Quanto às medidas de dispersão, para a variável em estudo a variância é aproximadamente 1056 e o desvio padrão é aproximadamente 33. Os técnicos resolveram pelo menos 25% dos processos no próprio dia em que o cliente fez o pedido (1º quartil). Em relação ao 3º quartil, pelo menos 75% dos processos foram resolvidos em menos de três dias.

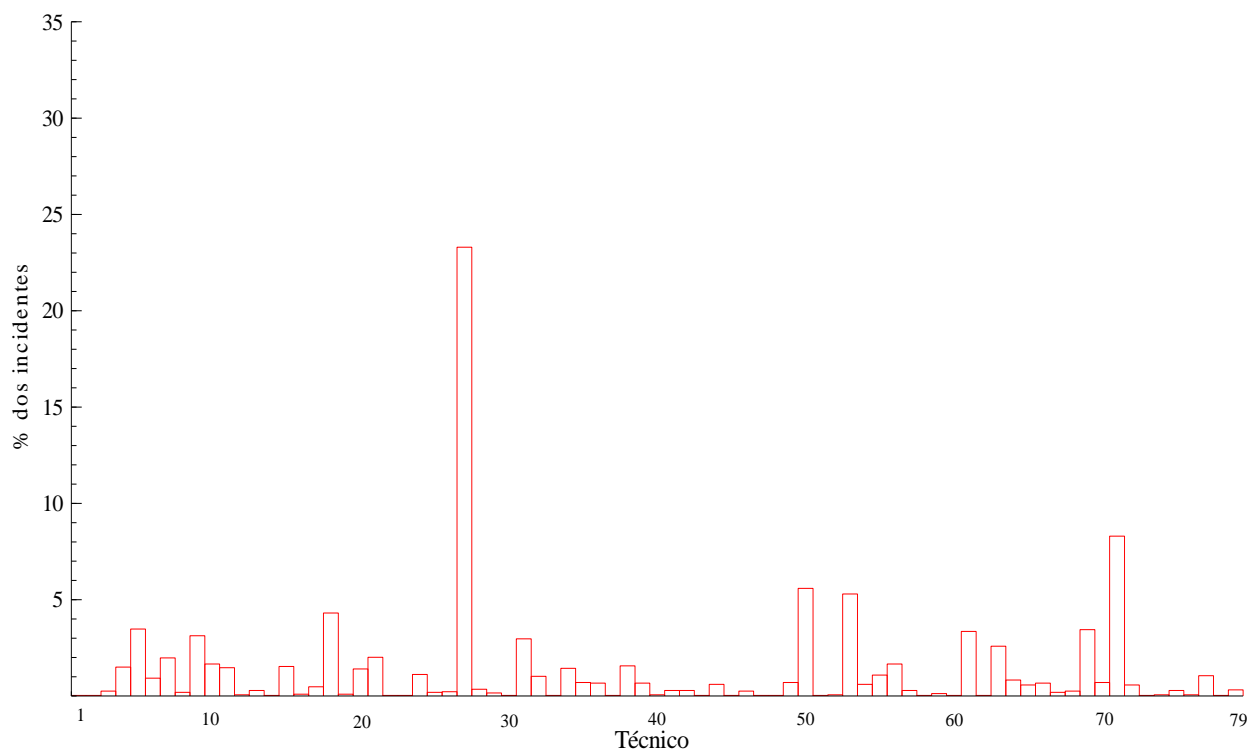
## 5.2 – Análise gráfica

**Figura 1:** Distribuição de frequências dos pedidos de resolução de incidentes, por área:



Olhando para a figura 1 pode dizer-se que há áreas mais susceptíveis de terem mais incidentes que outras. Em termos percentuais, as áreas de Recursos Humanos e de Gestão Financeira são as que têm mais incidentes. A área de Sistemas de Gestão Estratégica apenas tem dois incidentes.

**Figura 2:** Distribuição de frequências dos incidentes resolvidos por cada técnico:

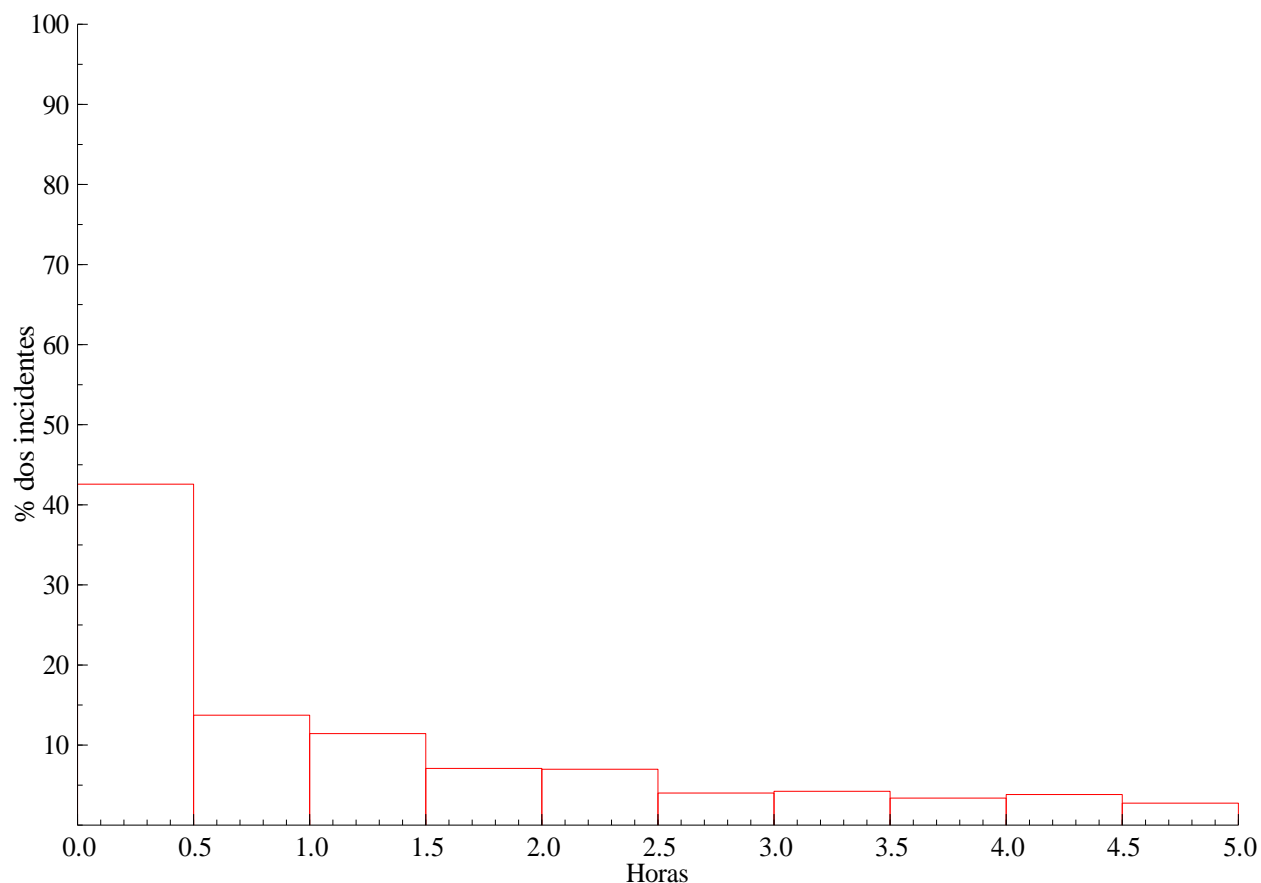


O técnico 27 teve 729 incidentes a seu cargo, ou seja, cerca de 23% do total dos incidentes. A título de curiosidade, grande parte dos incidentes que este técnico resolvia fazia parte do programa 70. O técnico 27 tem muitos processos de resolução de incidentes que levam muitos dias até serem encerrados. Assim pode concluir-se que este técnico muitas vezes resolve incidentes muito complexos.

Há muitos técnicos que têm uma percentagem muito pequena de processos de resolução de incidentes. Por exemplo, o técnico 47 apenas resolveu um incidente com prioridade um. Este processo de resolução do mesmo demorou 173 dias.

Apenas 6 técnicos têm um maior número de processos de incidentes para resolver.

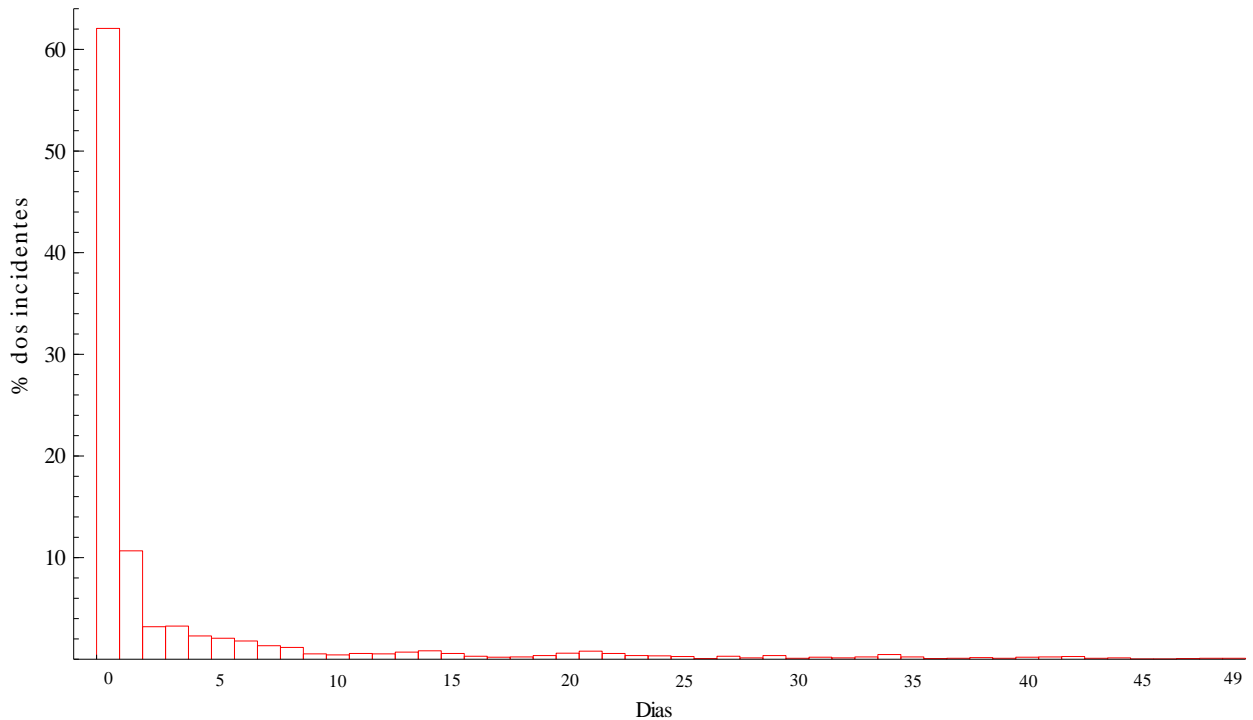
**Figura 3:** Histograma do número de horas efectivas de resolução do incidente (incidentes com menos de cinco horas):



Como cerca de 90% dos processos levaram menos de cinco horas efectivas a serem resolvidos, achou-se oportuno mostrar apenas os processos que demoraram menos de cinco horas.

Quarenta e dois e meio por cento desses incidentes foram resolvidos no máximo em meia hora.

**Figura 4:** Histograma com o número de dias que o processo de resolução do incidente levou até ser encerrado (processos com menos de 50 dias):



Primeiramente procedeu-se à construção do histograma com o número de dias que o processo de resolução do incidente foi encerrado.

O número máximo de dias é de 458, mas como cerca de 95 % dos incidentes levaram menos de 50 dias a serem resolvidos, achou-se oportuno apresentar um histograma apenas com os incidentes que demoraram menos de 50 dias.

Como se pode ver no histograma acima, cerca de 60% dos processos de incidentes analisados demoraram zero dias a serem resolvidos, ou seja foram encerrados no mesmo dia em que chegou o pedido da parte do cliente. Esta conclusão que acabámos de retirar é muito positiva para a empresa.

**Tabela 1:** Análise temporal (por ano):

<b>Período</b>	<b>Nº de casos incid</b>	<b>Nº médio dias resolução</b>	<b>Percentil 75</b>	<b>Máximo dias</b>
<b>Iº Sem. 2006</b>	112	12,87	8,25	263
<b>IIº Sem. 2006</b>	224	6,67	3	189
<b>Iº Sem. 2007</b>	410	7,47	2,75	299
<b>IIº Sem. 2007</b>	244	4,84	1	168
<b>Iº Sem. 2008</b>	400	12,25	5	327
<b>IIº Sem. 2008</b>	162	16,08	2	237
<b>Iº Sem. 2009</b>	477	9,15	4	412
<b>IIº Sem. 2009</b>	354	8,72	1	458
<b>Iº Sem. 2010</b>	215	14,36	6	288
<b>IIº Sem. 2010</b>	277	6,31	1	173
<b>Iº Sem. 2011</b>	258	4,87	5	55

Antes de começar a interpretar a tabela é importante salientar que apenas há dados desde Abril no primeiro semestre de 2006. Em relação ao primeiro semestre do ano 2011 há dados até Março. Em termos gerais, os anos de 2007 e de 2009 foram os anos em que o número médio de dias de resolução foi baixo, e em contrapartida o ano de 2008 foi o ano onde esse número foi mais elevado.

Na passagem do primeiro semestre para o segundo semestre para os anos de 2006 e 2007 houve uma descida acentuada no número médio de dias de resolução do incidente. Esta descida pode justificar-se pela chegada de novos técnicos ou pela presença de novos programas.

Os primeiros semestres de 2007, de 2008 e de 2009 foram três semestres com muitos processos de resolução de incidentes. Pela tabela, apesar de haver muitos processos, eles foram encerrados numa média de aproximadamente 7, 12 e 9 dias, respectivamente.

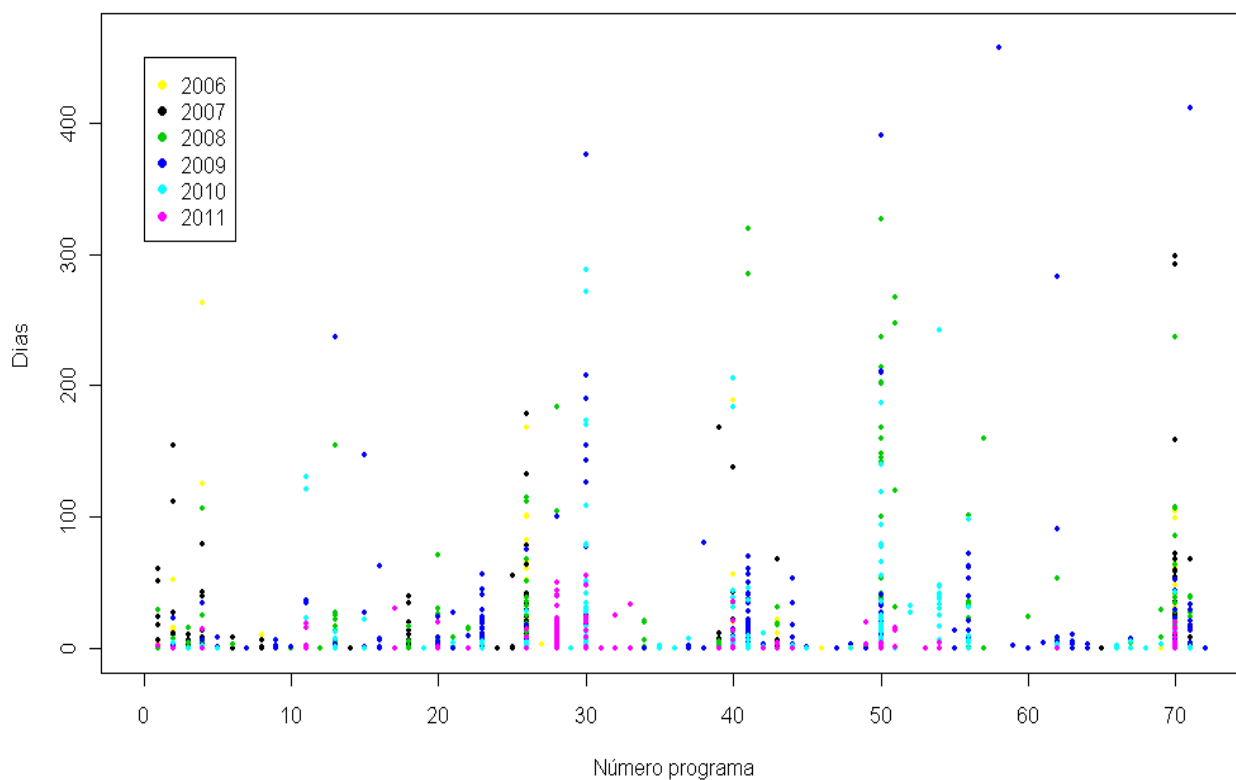
A partir de 2009, houve uma queda acentuada no número médio de dias, excepto no primeiro semestre de 2010. Em particular, é importante salientar que a partir do primeiro semestre de 2010 houve um decréscimo muito acentuado e parece que esta tendência decrescente se está a manter, já que no início de 2011, o número médio de dias de resolução continua a diminuir. Assim sendo, está a haver um esforço em manter a tendência decrescente nos resultados.

Nota-se também uma diminuição do número máximo de dias a partir de 2009, que poderá depender de vários factores, como por exemplo a contratação de novos técnicos ou a

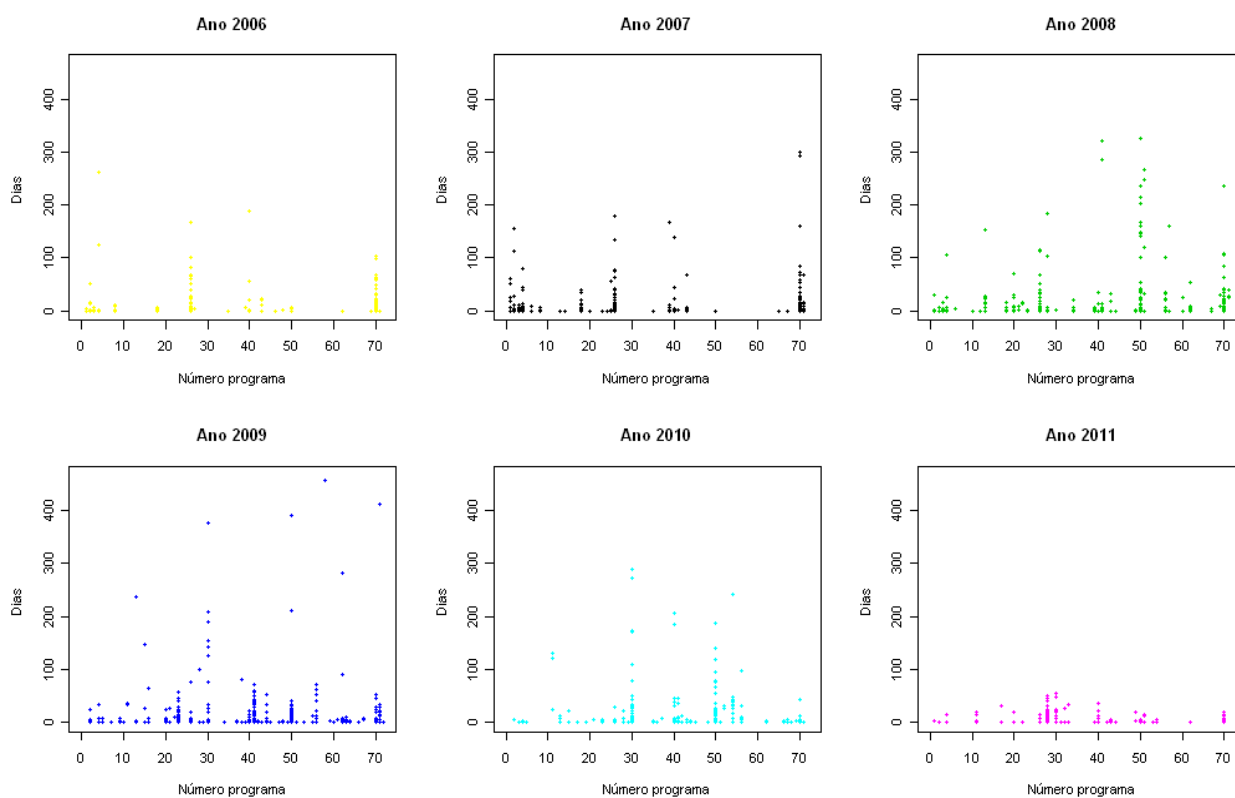
eliminação/substituição de programas mais problemáticos, ou ainda o pedido de *feedback* mais rápido da parte dos clientes.

Com a coluna denominada por percentil 75 fica claro que há três períodos em que os técnicos resolveram pelo menos 75 % dos processos dos incidentes em menos de um dia. O primeiro semestre de 2006 é o que tem o valor mais alto.

**Figura 5:** Gráfico com o número de dias em função da variável “programa”:



**Figura 6:** Gráfico com o número de dias em função da variável “programa”, por ano:



Olhando para as figuras 5 e 6, pode dizer-se que grande parte dos incidentes demora poucos dias a ser resolvido. A experiência ganha na resolução dos incidentes tem um papel fundamental na diminuição do número de incidentes e do tempo de resolução pois actuam directamente na solução e detecção dos problemas relacionados, neste caso, com o software.

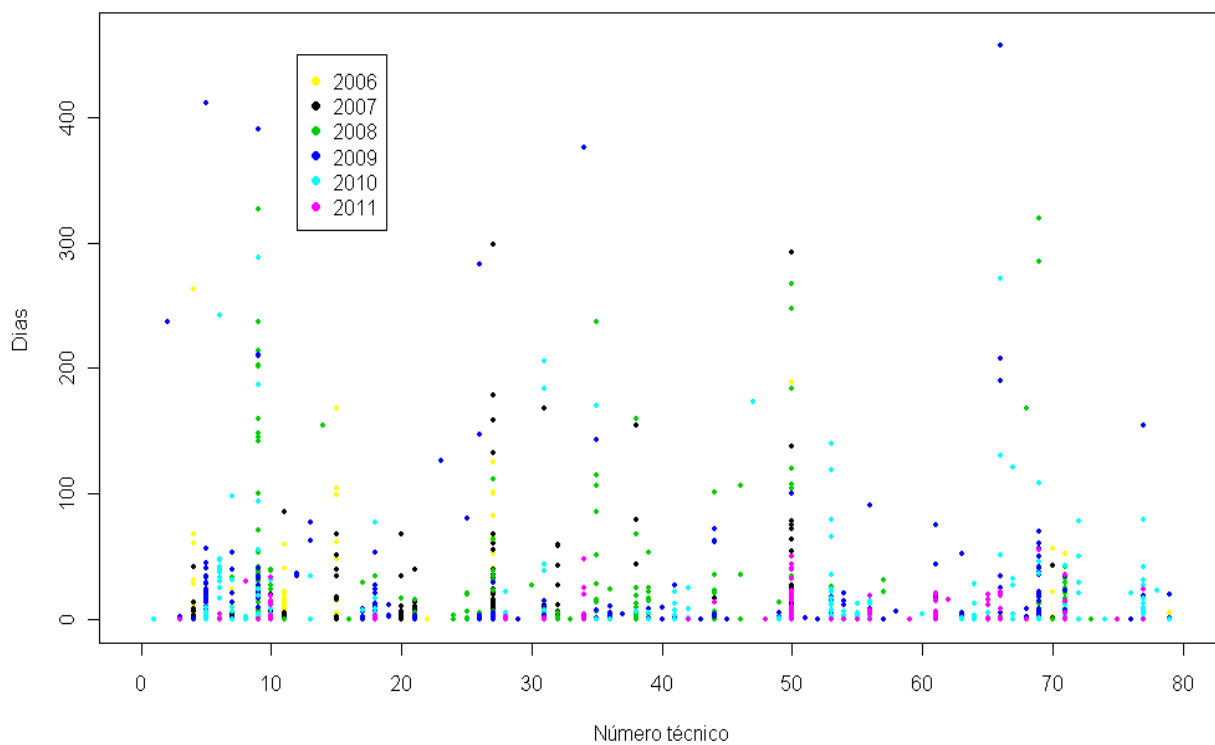
Relativamente aos dois gráficos acima pode salientar-se que os programas que deram tempos mais longos no passado ainda existem, mas deram tempos de resolução mais curtos em 2011 (por ex. os n. 26, 30, 50, 70). Isto pode ser porque os incidentes eram menos graves, ou porque o programa foi melhorado, ou ainda porque os processos de resolução foram melhorados.

Não foi encontrada evidência da eliminação no tempo dos programas mais problemáticos, mas suspeita-se que tenham sido resolvidas questões estruturais desses programas, o que levou ao fecho mais rápido dos incidentes relativos aos mesmos nos últimos anos.

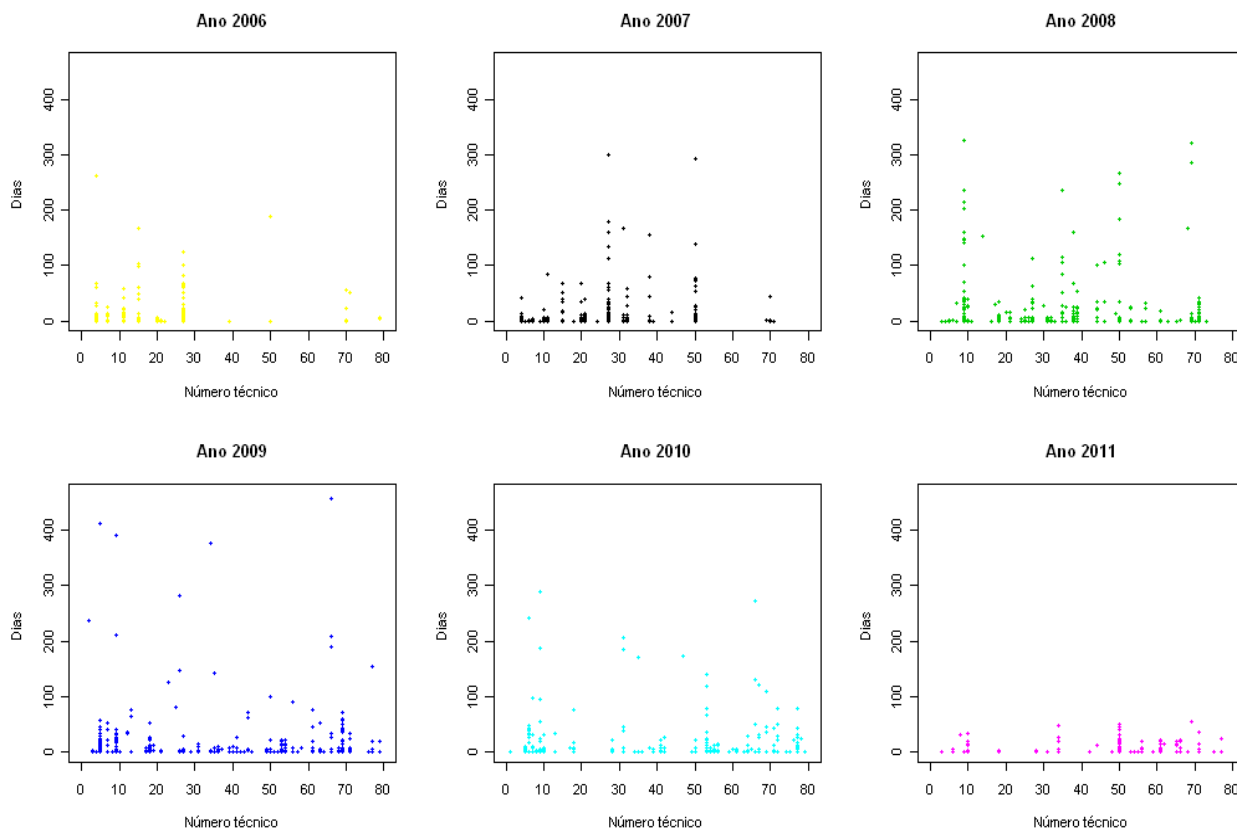
Há também programas que são mais susceptíveis de terem mais incidentes do que outros.



**Figura 7:** Gráfico com o número de dias em função da variável “técnico”:



**Figura 8:** Gráfico com o número de dias em função da variável “técnico”, por ano:

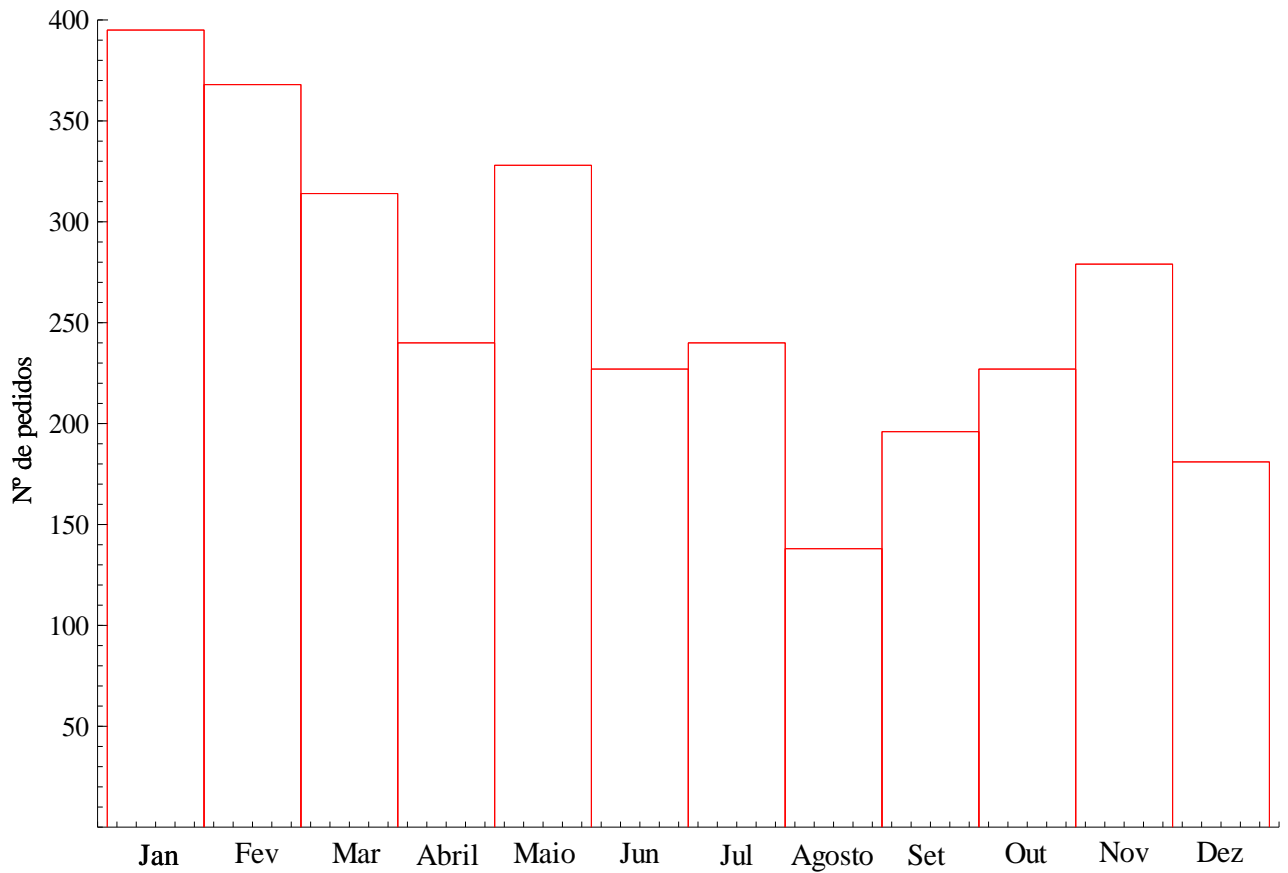


Quando se olha para o gráfico acima a maior parte dos técnicos demora muito pouco tempo a resolver os problemas. No entanto ainda há uma grande percentagem de incidentes que levam bastantes dias para serem resolvidos.

Não se evidenciou uma alteração dos tempos devida à entrada de novos técnicos, mas sim uma diminuição dos tempos para os mesmos técnicos por anos.

De acordo com a figura 7, pode dizer-se que houve alguns técnicos com tempos maiores, como por exemplo os números 9, 27, 50 e 69. Em 2011, estes técnicos tiveram tempos mais curtos, por isso não saíram da empresa. No entanto, alguma coisa se alterou, pois a partir de 2010 começaram a desaparecer os incidentes mais problemáticos. Este acontecimento pode ser devido a que os clientes sejam mais exigentes nos prazos de entrega ou os técnicos receberem incidentes menos graves. O único técnico que não aparece em 2011 é o 27. Isto não quer dizer que ele se tenha ido embora, pode ser que ainda não tivesse recebido nenhum incidente para resolver, uma vez que os dados acabam em Abril.

**Figura 9:** Distribuição mensal dos pedidos feitos pelos clientes:



Ao olhar para este histograma fica claro que o mês de Janeiro é o mês em que os técnicos recebem mais pedidos. Este facto é compreensível porque é o mês em que as empresas têm mais trabalho e qualquer erro que haja no *software* precisa de ser corrigido, rapidamente. A seguir tem-se os meses de Fevereiro, Março, Maio e Novembro. Os outros apresentam valores mais ou menos idênticos, com excepção do mês de Agosto. Como já foi dito anteriormente é de esperar que o mês de Agosto seja o mês com menos pedidos devido ao facto de muitas empresas estarem de férias.

De acordo com o gráfico, o mês de Janeiro teve cerca de 420 incidentes para serem tratados enquanto que, o mês de Agosto teve apenas de cerca de 150.

## 6 – Apresentação dos modelos estatísticos

### 6.1 – Introdução

O objectivo deste estudo é estudar os incidentes que têm de ser resolvidos, analisando os tempos que decorrem até à conclusão do processo de resolução.

Seja  $T$  a variável aleatória que representa a duração e  $t$  uma particular concretização de  $T$ .

No presente estudo, o estado inicial é o início da resolução do incidente e  $T$  será o tempo medido em dias até um técnico resolver o incidente e encerrar o processo depois de ter recebido o feedback da parte do cliente.

Em termos práticos, a análise de sobrevivência é um conjunto de processos estatísticos muito utilizados na análise dos dados para a qual a variável dependente é o tempo que decorre até que um determinado acontecimento se verifique. No nosso caso particular, o acontecimento a verificar é o fecho do incidente.

A base de dados em estudo tem dados completos, ou seja, não tem dados censurados ou truncados pois temos o tempo de sobrevivência exacto para cada incidente.

A função de distribuição de  $T$  é dada por:

$$F(t) = P(T \leq t), t > 0,$$

ou seja, representa a probabilidade da duração ser inferior ou igual a  $t$ .

Assim sendo, a função de sobrevivência  $S(t)$  é definida como o complementar de  $F(t)$ :

$$S(t) = 1 - F(t) = P(T > t),$$

ou seja, representa a probabilidade da duração ser superior ao momento  $t$ .

Uma função muito utilizada nos modelos é a função *hazard* [ $h(t)$ ] que mede a taxa à qual os incidentes ficam encerrados. Formalmente,  $h(t)$  é definida por:

$$h(t) = \lim_{\Delta t \rightarrow 0} \frac{P(t \leq T < t + \Delta t \mid T \geq t)}{\Delta t}.$$

A função *hazard* é interpretada como sendo a intensidade de transição e pode ser expressa como:

$$h(t) = \frac{f(t)}{S(t)},$$

onde  $f(t) = F'(t)$  é a função densidade de probabilidade de T.

## **6.2 – Modelo de Kaplan-Maier**

O modelo de Kaplan-Maier é um modelo de estimação não paramétrico da função de sobrevivência. Neste trabalho, pretendemos estimar a função de sobrevivência da variável “dias” (Cameron, A. Colin e P. K, Trivedi (2008)).

A estimação Kaplan-Maier é:

$$\hat{S}_{KM}(t) = \prod_{t_i < t} \frac{r(t_i) - d(t_i)}{r(t_i)},$$

onde  $r(t_i)$  e  $d(t_i)$  são o número de acontecimentos (incidentes ainda por resolver no tempo  $t_i$ ) e o número de mortes (no caso presente é o numero de incidentes resolvidos no tempo  $t_i$ ), respectivamente.

É importante salientar que o valor zero na variável “dias” inclui um intervalo infinitesimal, uma vez que esta variável pode ser considerada como sendo uma variável aleatória contínua. Relativamente à variável “horas”, esta também considera um intervalo infinitesimal.

Para se estimar este modelo em R vamos utilizar a função *survfit* (Venables W. N (2002) e S-PLUS 7.0). Esta função dá-nos um *output* que inclui as estimações da duração, dos erros padrão e dos intervalos de confiança.

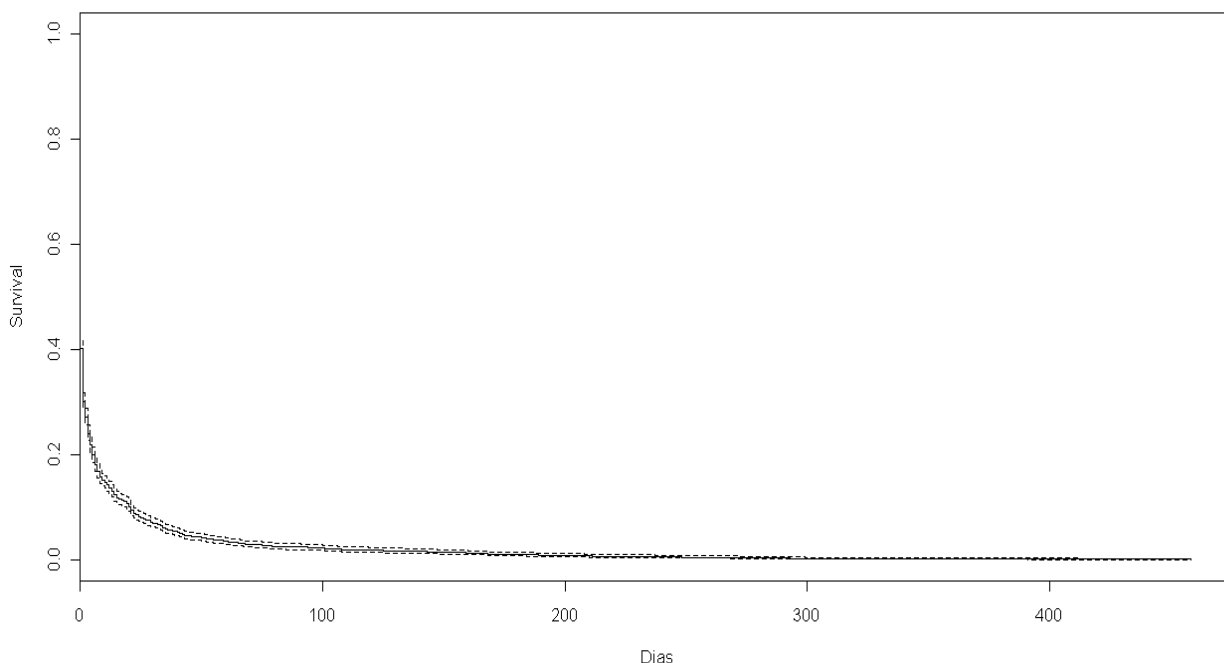
## 7 – Apresentação dos resultados de estimação

### 7.1 – Análise de sobrevivência da variável “dias”

Para a variável “dias” cerca de 80% dos processos de resolução dos incidentes foram encerrados em menos de 6 dias. A taxa de sobrevivência é de 20%, ou seja, 20% dos incidentes passaram para o nível seguinte, isto é, encerraram depois do sexto dia.

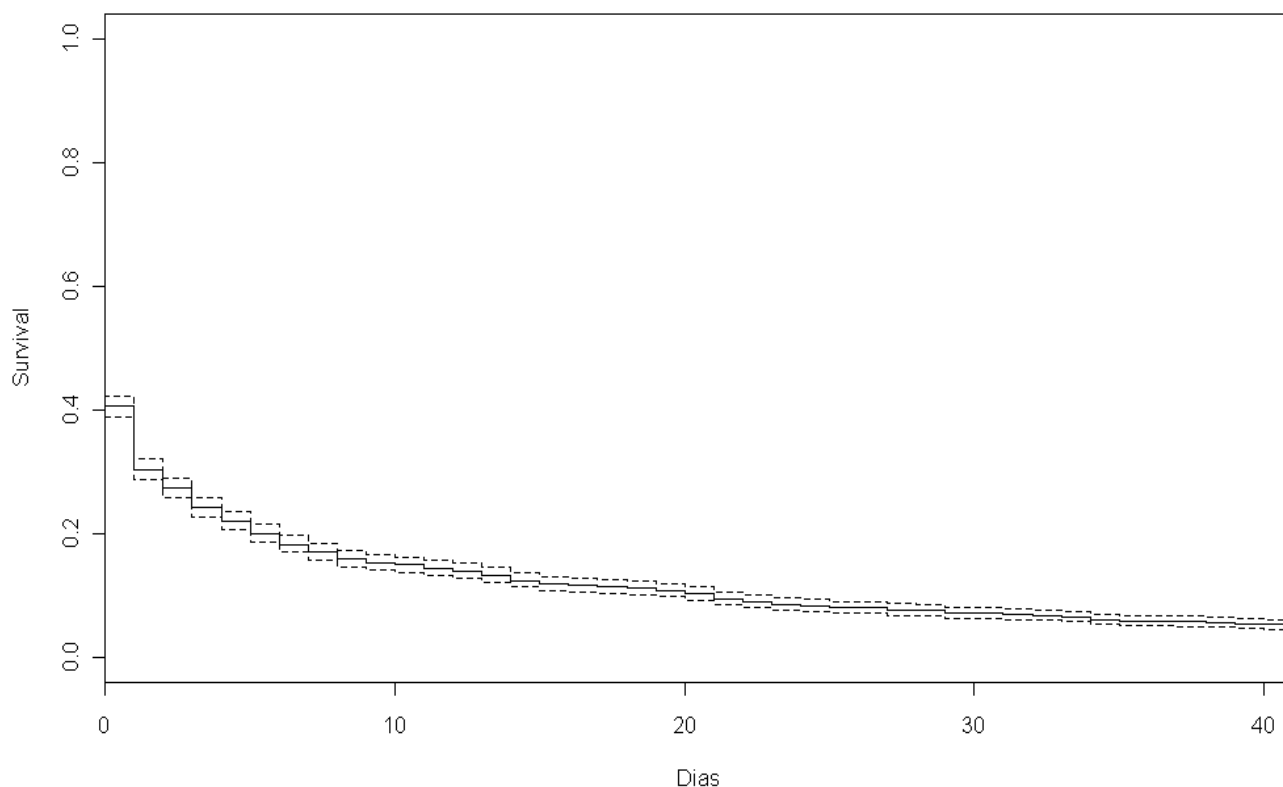
Aproximadamente 90% dos processos foram encerrados dentro de 20 dias e 95 % foram fechados em menos de 42 dias.

O gráfico de sobrevivência pode ver-se de seguida:



Para se fazer uma análise mais rigorosa achou-se adequado ampliar o gráfico até aos 41 dias, já que é neste ponto que cerca de 95 % dos processos já estão encerrados. A maior parte dos processos é fechado em menos de 41 dias. Apenas 5 % (taxa de sobrevivência) dos processos são encerrados em pelo menos 42 dias. Esta informação é bastante positiva para a empresa.

O gráfico de sobrevivência ampliado é o seguinte:



Neste gráfico mais ampliado que o outro pode constatar-se que 60 % dos processos são encerrados no próprio dia em que o cliente fez o pedido.

## **7.2 – Análise de sobrevivência com a variável “dias” como variável dependente e a variável “prioridade” ou “ano” como variável explicativa**

A variável “prioridade” pode influenciar bastante a variável dependente pois quanto mais urgente for a resolução do incidente, menos serão os dias necessários para que o processo de resolução do incidente fique encerrado.

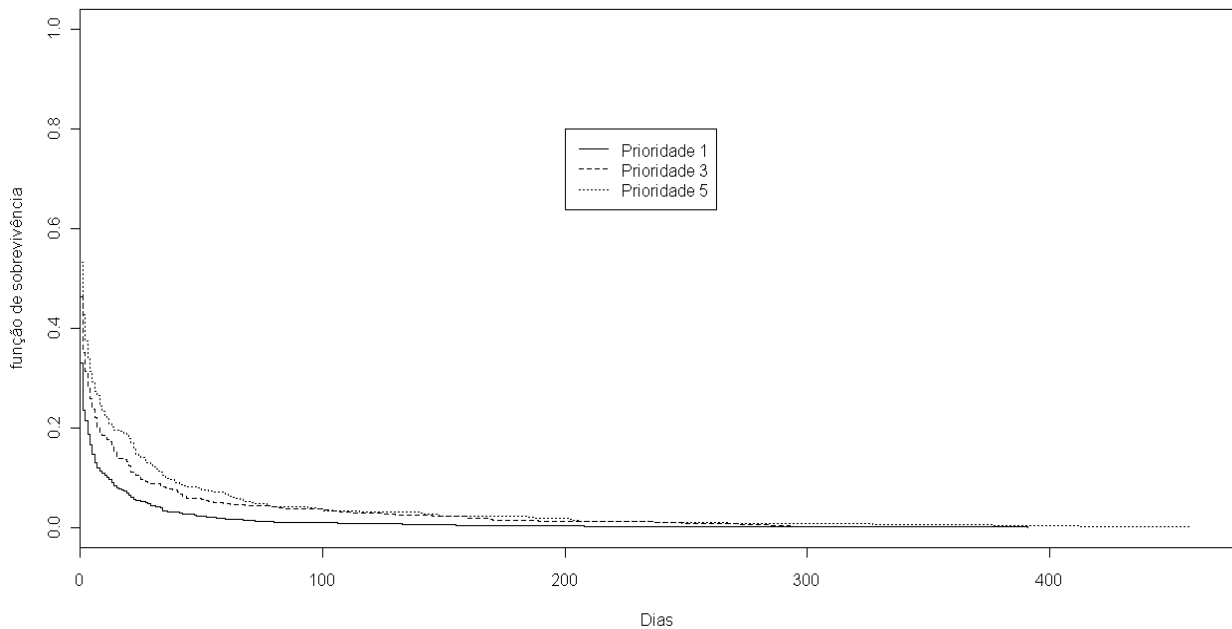
De acordo com o gráfico de sobrevivência, a curva mais a baixo é referente à prioridade igual a 1, a do meio refere-se à prioridade igual a 3 e a de cima à prioridade igual a 5.

Quanto mais a resolução do incidente for urgente, menor é a taxa de sobrevivência. Para a prioridade igual a 1, cerca de 77 % dos processos são resolvidos no próprio dia. Isto quer dizer que só 23 % dos processos demoram um dia ou mais até serem resolvidos.

Para a prioridade igual a 3, cerca de 54 % dos processos são resolvidos em zero dias, ou seja, 46 % dos processos são resolvidos em pelo menos um dia.

Como já foi dito, a taxa de sobrevivência para a prioridade igual a cinco é maior que as outras, uma vez que há pouca urgência na resolução do incidente. Essa taxa é igual a 53 %, aproximadamente. Isto significa que só 47 % dos processos foram encerrados no próprio dia em que foi feito o pedido, bastante inferior ao valor de 60 % calculado anteriormente sobre todos os incidentes.

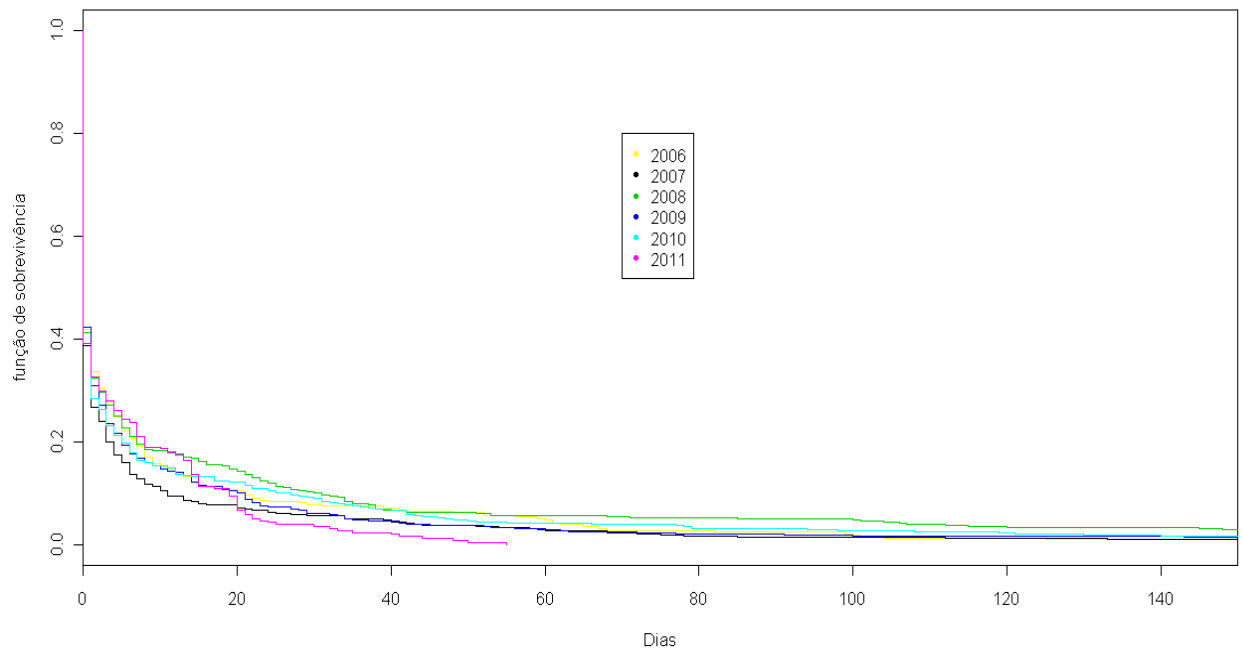
O gráfico de sobrevivência para a variável “prioridade” é:



O próximo gráfico é respeitante à variável “ano”. Vai-se fazer a mesma análise para a variável em questão.

Apenas se vai mostrar o gráfico de sobrevivência já ampliado para durações menores que 150 dias, uma vez que a partir de aqui as curvas estão todas sobrepostas.





Neste gráfico, vê-se que a curva roxa de 2011 desce mais depressa que as outras, ou seja, parece que os tempos de resolução estão a diminuir não só nos primeiros dias, mas também depois. Esta análise teria de ser confirmada depois de ter dados do ano todo.

Os anos de 2006 e 2009 tiveram uma taxa de sobrevivência de 42 % dos processos de incidentes. Isto quer dizer que 58 % dos processos foram resolvidos no próprio dia.

Em 2008, a taxa de sobrevivência foi de 41 % dos processos de incidentes. Houve mais processos que foram resolvidos em menos de um dia, em relação a 2006 e 2009.

Em 2010 e 2011 houve uma descida significativa na taxa de sobrevivência. Para os dois anos, esta foi de 39 % dos processos de incidentes. Neste caso, 61 % dos processos foram resolvidos em menos de um dia.

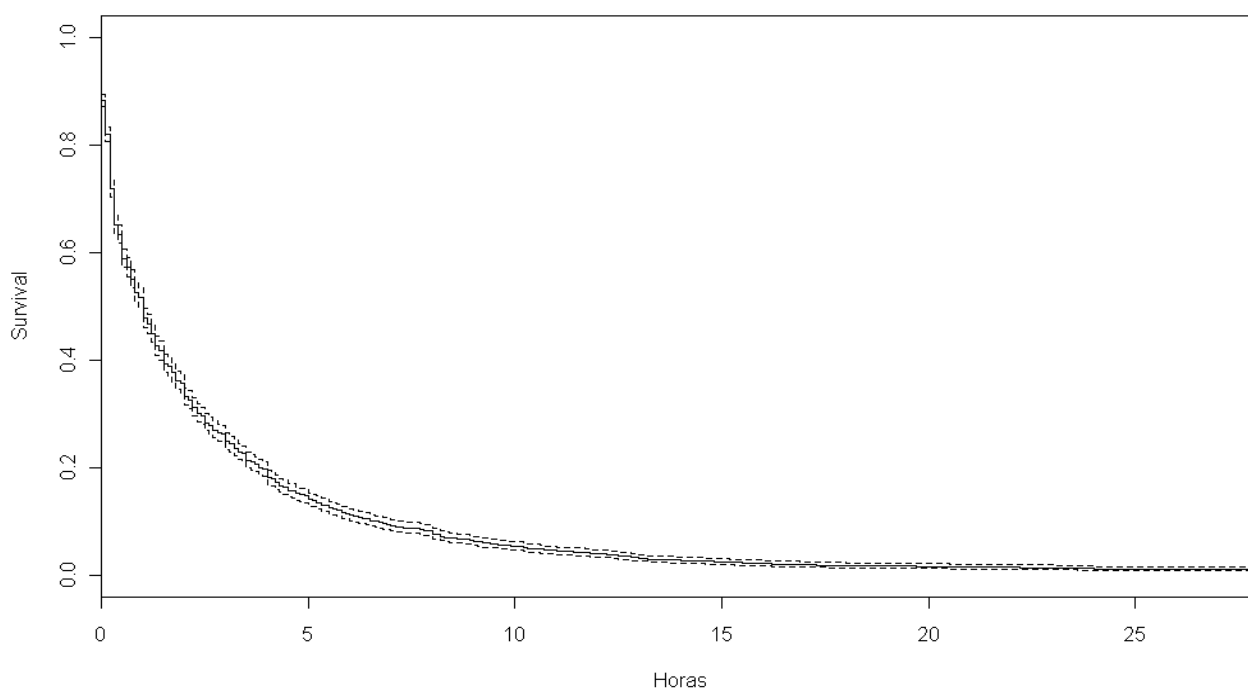
O ano de 2007 foi o ano em que a taxa de sobrevivência foi mais baixa (cerca de 38 %), e por conseguinte houve mais processos encerrados no próprio dia (cerca de 62 %).

Com este gráfico, confirma-se a análise feita anteriormente, em relação à variável “dias”.

### 7.3 – Análise de sobrevivência da variável “horas”

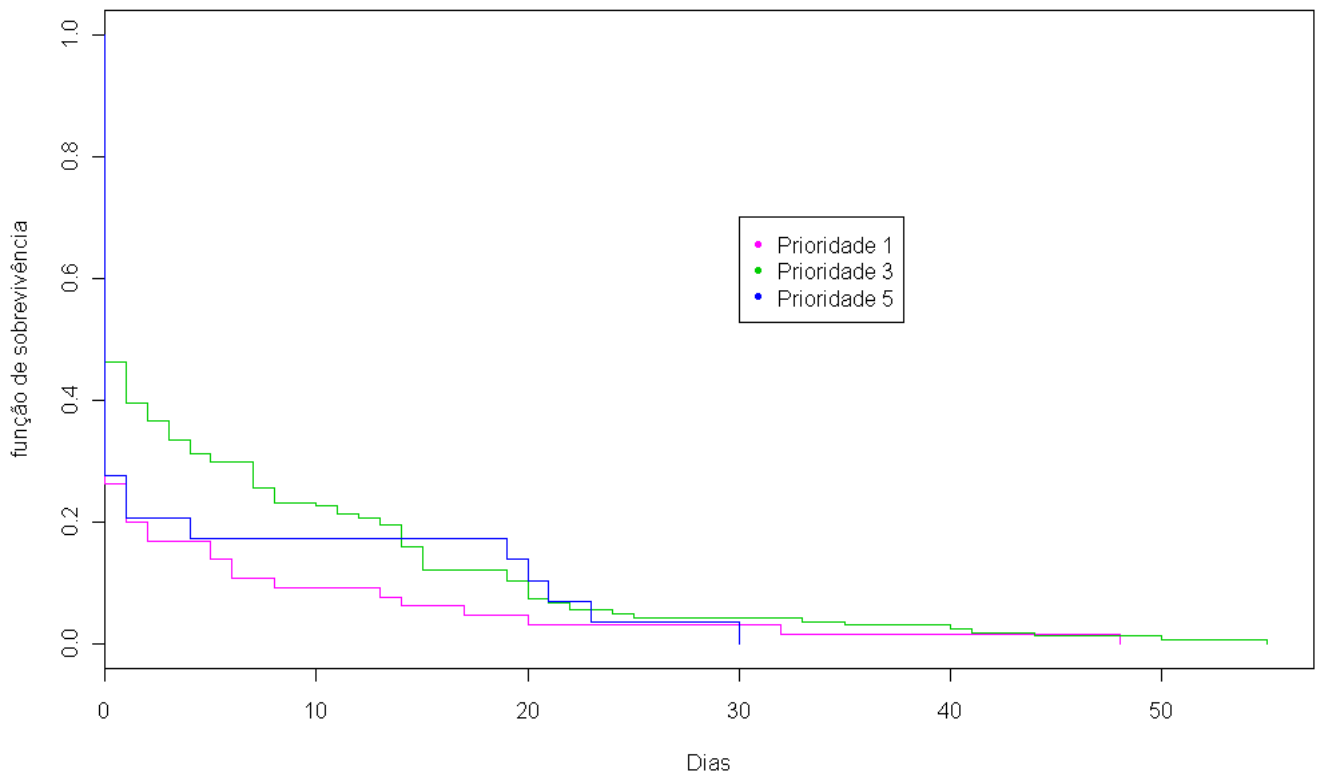
Apesar do número máximo para a variável “horas” ser 107.3, achámos conveniente só apresentar o gráfico ampliado até às 28 horas.

Assim, cerca de 86 % dos processos são fechados em menos de 5 horas. Para se ter uma noção mais precisa da rapidez na resolução dos incidentes, cerca de 99 % dos processos foram resolvidos em menos de 28 horas. Pode ver-se o gráfico que expressa estes valores de seguida:



Também é importante salientar que quarenta e oito por cento dos processos são resolvidos em menos de uma hora, aproximadamente. Oitenta por cento dos processos de incidentes são resolvidos em menos de quatro horas.

## 7.4 – Cálculo da função hazard para 2011, por prioridade



Através deste gráfico fica claro que os incidentes com um nível de prioridade 1 têm uma taxa de sobrevivência menor em relação às outras prioridades.

Não foi possível fazer previsões com estes dados, uma vez que o modelo utilizado é não paramétrico. No entanto, calculou-se o tempo esperado para a resolução de um incidente que entre agora. Este cálculo é apresentado de seguida:

- Para a prioridade 1: 2,8 dias
- Para a prioridade 3: 5,8 dias
- Para a prioridade 5: 4,1 dias.

Em relação à variável “programa”, pode também calcular-se o tempo esperado de resolução por programa. Por exemplo:

- Para o programa 20: 2,1 dias
- Para o programa 28: 10,35 dias
- Para o programa 70: 2,43 dias.

## **8 – Conclusões**

Estudaram-se fundamentalmente as variáveis “dias” (variável dependente), “prioridade”, “horas”, “técnico” e “programa”.

De acordo com a análise que se fez ao longo deste relatório verificámos que os processos de resolução dos incidentes são bastante céleres. Sessenta por cento dos processos de incidentes são resolvidos no próprio dia. Relativamente à variável “horas”, pode concluir-se que mais de 40 % dos processos são resolvidos em menos de meio hora.

Tendo-se observado uma tendência nos anos para um fecho dos processos mais célere, procedeu-se a uma análise dos tempos por programa e por técnico nos vários anos. Não se evidenciou uma alteração dos tempos devida à entrada de novos técnicos, mas sim uma diminuição dos tempos para os mesmos técnicos por anos.

Analogamente, não foi encontrada evidência da eliminação no tempo dos programas mais problemáticos, mas suspeita-se que tenham sido resolvidas questões estruturais desses programas, o que levou ao fecho mais rápido dos incidentes relativos aos mesmos nos últimos anos. A diminuição dos tempos pode também ser devida a uma política da empresa de maior exigência de feedback rápido da parte do cliente, sendo que no passado a falta de celeridade no feedback tinha atrasado em alguns casos o fecho dos processos.

Utilizou-se o modelo de Kaplan-Maier com a finalidade de poder fazer uma análise de sobrevivência aos dados.

Pela análise de sobrevivência que se fez para as três variáveis pode concluir-se que a taxa de sobrevivência dos processos de resolução dos incidentes não é muito elevada. Isto quer dizer que a percentagem de processos que não sobrevivem é alta, ou seja, há muitos processos que são fechados muito rapidamente. Por exemplo, sessenta por cento dos processos de incidentes são encerrados no próprio dia em que o cliente fez o pedido. Também é bastante positivo ilustrar que até aos 41 dias, já cerca de 95 % dos processos já estejam encerrados.

## **9 – Sugestões para trabalhos futuros**

Dado que não foi possível usar a variável “descrição” por não se conseguir categorizar a mesma, propomos à empresa classificar a variável em questão em categorias oportunas, de forma a recolher os dados futuros de uma forma mais simples de analisar, em termos estatísticos. Uma ideia de uma possível categorização é o tipo de problema. No entanto, para a empresa esta variável pode não ter muita influência no que toca aos prazos de entrega, uma vez que para a empresa, os prazos são calculados utilizando outras informações, tais como o técnico, ou o programa.

Seria também interessante estudar a aplicação de modelos paramétricos, como por exemplo, o modelo de Cox com *hazard* proporcional com o objectivo de poder fazer previsões e utilizar os modelos de duração para determinar o tempo de resposta para um incidente que entre agora, usando como variáveis explicativas a “prioridade”, o “programa”, o “técnico” e o “ano”.

## **10 – Bibliografia**

- Cameron, A. Colin e P. K, Trivedi (2008). *Microeconometrics: Methods and Applications*, Cambridge University Press, New York, 7ª edição;
- Murteira, Bento J. F. (1993), *Análise exploratória de dados – Estatística Descritiva*, MacGraw Hill;
- *S-PLUS 7.0 Guide to Statistics, Volume 2*, Insightful Corporation, Seattle, Washington;
- Venables, W. N. e Ripley, B. D. (2002), *Modern applied statistics with S*, New York, 4ª edição, pp.353-385;