Universidade do Minho

Mestrado Integrado em Engenharia Informática

Perfil **Sistemas Inteligentes**

Unidade Curricular de **Computação Natural**

Edição **2016/2017**

REDES NEURONAIS

Autores:
Diana Oliveira (a67652)
Gil Gonçalves (a67738)
Pedro Duarte (a61071)
Pedro Lima (a61061)

Braga, 13 de Junho de 2017

Resumo

O presente relatório apresenta a descrição do desenvolvimento de um trabalho teórico-prático, realizando em grupo, relacionado com as temáticas abordadas nas aulas de Computação Natural, em particular a Rede Neuronal Artificial, constituída por uma secção introdutória de contextualização e definição do problema de investigação abordado, uma avaliação crítica do trabalho efetuado e algumas de referências bibliográficas consultadas durante a elaboração do trabalho.

O trabalho pretendia a utilização de Redes Neuronais Artificias (RNA) que previssem o valor da despesa por município, grupo de municípios e do conjunto dos municípios portugueses tendo em conta as despesas efetuadas durante os anos compreendidos entre o período de 2010 a 2015.

Conteúdo

0.1 Introdução

Uma Rede Neuronal Artificial (RNA) é um sistema computacional de base conexionista para a resolução de problemas sendo concebida com base num modelo simplificado do sistema nervoso central dos seres humanos e definida por uma estrutura interligada de unidades computacionais, designadas neurónios, com capacidade de aprendizagem. O neurónio é a unidade computacional de composição da RNA identificado pela sua posição na rede e caracterizado pelo valor do estado, por sua vez um Axónio é a via de comunicação entre os neurónio podendo ligar qualquer neurónio, incluindo o próprio. Essas ligações podem variar ao longo do tempo com a particularidade de a informação circular em um só sentido. Uma Sinapse é o ponto de ligação entre axónios e neurónios e o seu valor determina o peso (importância) do sinal a entrar no neurónio (excitativo, inibidor ou nulo). A variação no tempo determina a aprendizagem da RNA. O valor da ativação, que varia pelo tempo, é representado por um único valor sendo que a gama de valores varia com o modelo adotado (normalmente dependendo das entradas e de algum efeito de memória). O valor de transferência de um neurónio determina o valor que é colocado na saída (transferido através do axónio), este é calculado como uma função do valor de ativação (eventualmente com algum efeito de memória).

Há diferentes arquiteturas de RNA's, como *Feed forward*, *Feed forward* e Recorrente, bem como paradigmas com/sem supervisão de aprendizagem.

O treino de uma RNA corresponde à aplicação de regras de aprendizagem, por forma a fazer variar os pesos das ligações (sinapses). *Hebbian*, competitiva, estocástica, baseada na memória e gradiente decrescente são exemplos de regras de aprendizagem mais comuns.

Há algumas especificações a establecer como a quantidade de neurónios (na camada de entrada, na camada de saída, nas camadas intermédias), níveis (ou camadas) da RNA, ligações entre neurónios, topologia das ligações, esquema de atribuição e atualização dos pesos, funções (de transferência, de ativação, de aprendizagem) e métodos de Treino.

0.1.1 Motivação

A quantidade de dados, em exponencial expansão, que tem sido criada, gerada e armazenada a nível global é quase inconcebível. Isso significa que há inúmeras potencialidades de uso a ser extraídas desses dados, sustentando valiosas informações, seja qual for o objetivo, desde melhorias em eficiência a previsões de valores futuros.

Big Data é o termo que descreve o imenso volume de dados estruturados e não estruturados, sendo que o importante não seja a quantidade de dados mas sim o que fazer com eles.

Devido à problemática descrita, é impossível um ser humano conseguir analisar estes dados sem a ajuda de um modelo.

Sendo os instrumentos fornecidos pela equipa docente compostos por dados reais,

assim como controlada liberdade para escolha de um estudo específico acerca dos mesmos, este facto apresentou-se como uma motivação para o grupo em compreender de facto o problema e os seus dados de modo a partir para uma vertente de desenvolvimento que realmente *"nos disse-se algo"*.

0.1.2 Objetivos

O objetivo do trabalho seria a utilização de Redes Neuronais Artificias (RNA) para prever o valor da despesa por município, grupo de municípios e do conjunto dos municípios portugueses tendo em conta as despesas efetuadas durante os anos compreendidos entre o período de 2010 a 2015.

Para atingir este objetivos foi necessário seguir esta metodologia de etapas de desenvolvimento:

- Análise exploratória dos dados: aplicando métodos de análise às séries temporais com o objetivo de identificar as dependências entre os dados usando conhecimentos adquiridos na unidade curricular de *Aprendizagem e Extração de Conhecimento*;
- Particionar os *datasets* em dados de aprendizagem (treino) e de teste;
- Desenvolvimento de uma arquitetura da RNA que se adequa-se ao problema em causa;
- Criação de padrões de treino;
- Teste da RNA com o conjunto de dados anteriormente reservados para teste

0.1.3 Estrutura do relatório

- ***Capítulo 1*** Introdução aos modelos usados
- ***Capítulo 2*** correspondeste à fase de análise exploratória dos dados e do seu pré-processamento;
- ***Capítulo 3*** Comparação entre os modelos usados.
- ***Capítulo 4*** Conclusão, discussão de resultados e trabalho futuro.

Capítulo 1

Modelos usados

1.1 Regressão linear

Em estatística regressão linear é uma equação para se estimar a condicional (valor esperado) de uma variável y, dados os valores de algumas outras variáveis x.

A regressão, em geral, trata da questão de se estimar um valor condicional não esperado.

A regressão linear é chamada "linear"porque se considera que a relação da resposta às variáveis é uma função linear de alguns parâmetros. Os modelos de regressão que não são uma função linear chamam-se modelos de regressão não-linear. Sendo uma das primeiras formas de análise regressiva a ser estudada rigorosamente, e usada extensamente em aplicações práticas. Isso acontece porque modelos que dependem de forma linear dos seus parâmetros desconhecidos, são mais fáceis de ajustar que os modelos não-lineares aos seus parâmetros.

Os modelos de regressão linear apenas lidam com valores numéricos, logo para se usar estes tipos de modelos é necessário converter os valores todos para numéricos. Depois é preciso tratar dos valores desconhecidos ou o modelo irá dar resultados que podem levar ao engano do utilizador.

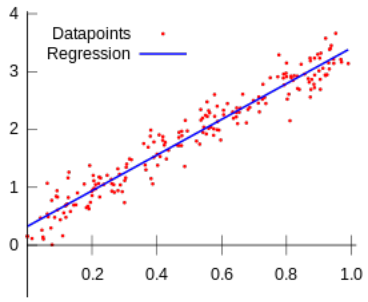

Figura 1.1: Exemplo de uma regressão linear

1.2 Redes neuronais

Uma rede neuronal e um sistema computacional para a resolução de problemas, desenhado com base num modelo simplificado do sistema nervoso central humano. Definida por uma estrutura interligada de unidades computacionais, designadas neurónios, com capacidade de aprendizagem.

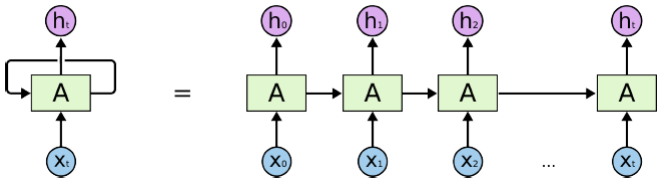

Figura 1.2: Exemplo de uma RNN

Existem vários tipos de Redes Neuronais entre eles:

- Redes Neuronais *FeedForward* de uma só camada;
- Redes Neuronais *FeedForward Multi-Camada*;
- Redes Neuronais Recorrentes.

A aprendizagem realizada por uma Rede Neuronal, esta assente em três importantes paradigmas:

Aprendizagem Supervisionada - de uma forma resumida são fornecidas as respostas corretas a rede, sendo que esta rede aprende a partir de padrões, assumindo-se a existência de um "professor"que ajuda na aprendizagem da mesma, dizendo quais os "caminhos"correto;

Aprendizagem por Reforço - nesta aprendizagem, assume-se também a existência de um "professor"tal como na aprendizagem supervisionada, sendo que a resposta correta não e apresentada a rede;

Aprendizagem Não Supervisionada - este tipo de aprendizagem não recebe qualquer indicação externa sobre uma possível resposta, sendo a aprendizagem realizada pela descoberta das características dos dados.

Capítulo 2

Análise exploratória e tratamento dos dados

2.1 Metodologia *CRISP-DM*

No desenvolvimento deste trabalho aplicou-se a metodologia *CRISP-DM* pois contempla todos os passos a seguir em *data mining*.

O modelo de ciclo de vida é composto de seis fases com setas indicando as dependências mais importantes e frequentes entre as fases. A sequência das fases não é rigorosa. De fato, a maioria dos projetos vão e voltam entre as fases, conforme necessário.

O modelo *CRISP-DM* é flexível e pode ser facilmente adaptado a algumas preferências.

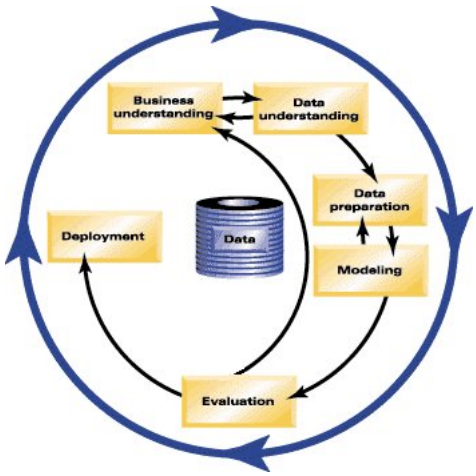

Figura 2.1: Metodologia *CRISP-DM*

2.2 Estudo do Negócio

É nesta fase que se determina o objetivo do negócio, faz-se uma avaliação da situação atual do negócio, define-se um objetivo para o negócio e produz-se um plano para o projeto.

Para este trabalho foi disponibilizado vários *datasets* contendo as despesas de várias cidades portuguesas. De forma a não prever o resultado para todas as cidades, tomamos a opção de no nosso trabalho passar só por prever as despesas anuais para a **região norte**. Através do modelo de previsão será possível as várias câmaras controlarem e projetarem o seu panorama financeiro de modo a assegurarem um desenvolvimento sustentável.

2.3 Estudo dos dados

Esta fase consistiu na recolha, descrição, exploração e análise da qualidade dos dados. Como foram disponibilizados vários *datasets* começou-se por efetuar uma leitura, fazendo uma análise atributo a atributo, de forma a perceber quais seriam os seus tipos (quantificador, categórico, binário, etc). Como os atributos eram todos os mesmos nos vários *datasets* o estudo acabou por ser simples. Partiu-se então para uma análise conferindo se os atributos não apresentavam grande dispersão dos dados, ou seja, se os valores se concentravam maioritariamente numa dada região e se se verificava a existência de valores nulos e *outliers*.

2.4 Preparação dos dados

Nesta fase selecionou-se os atributos mais relevantes face ao problema inicialmente proposto.

Criou-se as seguintes duas variáveis:

- **Variável "região"**, que apenas contém a palavra "Norte", para assim se distinguir quais as cidades que pertencem a região norte do país.
- **Variável "data"**, para assim distinguir os vários atributos dos diferentes *datasets*.

Para além disso, efetuando-se uma limpeza aos dados, sendo removidos os seguintes:

- TERRENOS_HABITACOES;
- DSC_AUTARQUIA;
- Idk;

- EDIFICIOS;
- Regiao;
- INVESTIMENTOS_OUTROS_INVESTIMENTOS_BENS_DE_CAPITAL;
- CODIGO_PERIODO.

Eliminou-se o atributo ***TERRENOS_HABITACOES*** após uma análise feita ao documento disponibilizado pela **Direção-Geral das Autarquias Locais(DGAL)** onde eram apresentados os terrenos como uma fonte de rendimento, facto esse que não interessava para o problema, optando-se pela sua remoção, sendo que pela mesma razão eliminou-se o atributo ***EDIFICIOS***.

O atributo ***DSC_AUTARQUIA*** foi eliminado porque não fazia sentido ter a descrição da autarquia quando existe um atributo chamado ***CODIGOINE*** que identificava inequivocamente cada cidade, sendo que pela mesma razão eliminou-se o atributo ***Idk***.

O atributo ***Regiao*** foi eliminado devido ao facto de se selecionar as câmaras que eram da região norte deixar de ser importante para a análise do problema.

Foi removido o atributo ***INVESTIMENTOS OUTROS INVESTIMENTOS BENS DE CAPITAL*** porque no *dataset* de 2014 este atributo aparecia a nulo. Existem várias maneiras de prever um atributo mas como só tinha valores nulos substituir pela moda ou média estava fora de questão, assim como contratar um perito e retirar valores de outra base de dados era improvável, visto não ter sido encontrada outra base de dados com estes valores. A única alternativa seria tentar estimular o valor através de um modelo(e.g. **regressão linear**).

Para isso utilizou-se o seguinte pseudo-código.

```
formula=INVESTIMENTOS_OUTROS_INVESTIMENTOS_BENS_DE_CAPITAL ~ DESPESA_AQUISICAO_BENS+
  DESPESA_TOTAL+
  INFRAESTRUTURAS_BASICAS+
  ACESSIBILIDADES+
  JUROS_ENCARGOS+
  DESPESA_COM_PESSOAL+
  OUTROS_INVESTIMENTOS_BENS_DE_CAPITAL+
  DESPESA_CORRENTE+
  CODIGOINE+
  TRANSFERENCIAS_OUTRAS_DESPESAS_CAPITAL+
  Data

predit.model<-predict(fit,newdata=test,type="response")


plot(test$INVESTIMENTOS_OUTROS_INVESTIMENTOS_BENS_DE_CAPITAL,predit.model,col='blue',
main='Real vs predicted lm',pch=18, cex=0.7)
abline(0,1,lwd=2)
legend('bottomright',legend='LM',pch=18,col='blue', bty='n', cex=.95)
```

O gráfico obtido foi o seguinte

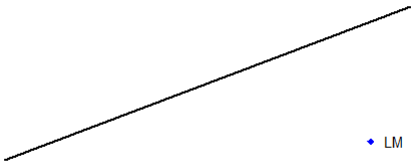

Figura 2.2: Gráfico resultante

Como é possível observar não é possível prever este atributo, logo teve que ser removido senão iria provocar interferência nos dados.

Após a filtração dos dados pelo período anual o atributo **CODIGO_PERIODO** foi então removido, visto que já não ia interferir com o modelo.

2.5 Modelação, Avaliação e Implementação

2.5.1 1º tentativa

Posto as fazes de **Estudo de Negócio, Estudo dos Dados e Preparação dos Dados** chegou a altura de escolher um modelo para a previsão de resultados. Visto que as minhas variáveis de entrada e a variável que se ia tentar prever eram todas numéricas o modelo usado foi o modelo de regressão linear. Ao executar o seguinte pseudo-código

```
formula=DESPESA_TOTAL ~ DESPESA_AQUISICAO_BENS+
  INFRAESTRUTURAS_BASICAS+
  ACESSIBILIDADES+
  JUROS_ENCARGOS+
  DESPESA_COM_PESSOAL+
  OUTROS_INVESTIMENTOS_BENS_DE_CAPITAL+
  DESPESA_CORRENTE+
  CODIGOINE+
  TRANSFERENCIAS_OUTRAS_DESPESAS_CAPITAL+
  Data+TRANSFERENCIAS_CORRENTES

  fit<- glm(formula ,family = gaussian,data = treino)

predit.model<-predict(fit,newdata=test,type="response")
```

apareceu a seguinte mensagem "Warning message: In predict.lm(object, newdata, se.fit, scale = 1, type = ifelse(type == : prediction from a rank-deficient fit may be misleading".

Ao executar a função **summary(fit)** verificou-se que o atributo **TRASNFERENCIAS_CORRENTES** estava a dar problemas.

Este erro deve-se ao facto de as variáveis não serem todas linearmente independentes, logo o R irá de descartar essa variável, ou seja, para o modelo definido está variável não serve para previsão. **TRASNFERENCIAS_CORRENTES**.

```
Coefficients: (1 not defined because of singularities)
                                        Estimate         Std. Error  t value
(Intercept)                        592660263.58165383 186023008.30346665  3.18595
DESPESA_AQUISICAO_BENS                    0.14038047         0.11495210  1.22121
INFRAESTRUTURAS_BASICAS                   1.06118722         0.15445426  6.87056
ACESSIBILIDADES                           1.43605873         0.09662490 14.86220
JUROS_ENCARGOS                            0.18530991         0.32571514  0.56893
DESPESA_COM_PESSOAL                       0.43325599         0.13024442  3.32648
OUTROS_INVESTIMENTOS_BENS_DE_CAPITAL      1.05938388         0.09362256 11.31548
DESPESA_CORRENTE                          0.93354464         0.09551076  9.77424
CODIGOINE                               -30.32632402       199.30490517 -0.15216
TRANSFERENCIAS_OUTRAS_DESPESAS_CAPITAL    0.76041340         0.04340336 17.51969
Data                                -294721.65399119     92453.02485097 -3.18780
TRANSFERENCIAS_CORRENTES                          NA                 NA       NA
```

Figura 2.3: Erro resultante

Como está fase tinha falhado, voltou-se então a fase de preparação dos dados e optou-se por descartar está variável.

2.5.2 2º tentativa

Ao remover o atributo **TRASNFERENCIAS_CORRENTES** a fórmula inicial também teve que ser alterada, passando assim a ficar com o seguinte aspeto:

```
formula=DESPESA_TOTAL ~ DESPESA_AQUISICAO_BENS+
  INFRAESTRUTURAS_BASICAS+
  ACESSIBILIDADES+
  JUROS_ENCARGOS+
  DESPESA_COM_PESSOAL+
  OUTROS_INVESTIMENTOS_BENS_DE_CAPITAL+
  DESPESA_CORRENTE+
  CODIGOINE+
  TRANSFERENCIAS_OUTRAS_DESPESAS_CAPITAL+
  Data
```

Depois ao executar o seguinte pseudo-código:

```
fit<- glm(formula ,family = gaussian,data = treino)
predit.model<-predict(fit,newdata=test,type="response")

plot(test$DESPESA_TOTAL,predit.model,col='blue',main='Real vs predicted lm',pch=18, cex=0.7)
abline(0,1,lwd=2)
legend('bottomright',legend='LM',pch=18,col='blue', bty='n', cex=.95)
```

irá aparecer o seguinte resultado:

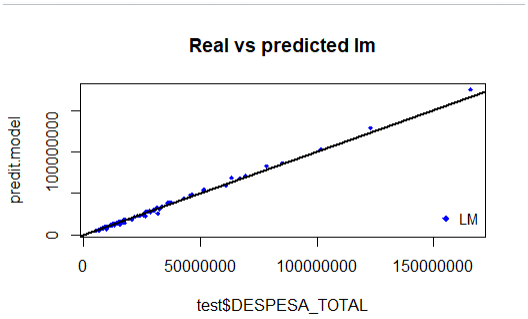

Figura 2.4: Gráfico resultante do modelo

Como é possível observar o existem muitos pontos azuis perto da reta isto significa que o modelo previu bons resultados.

2.6 Redes Neuronais

Posto a construção do modelo anterior criou-se um modelo de redes neuronais na tentativa de se obter melhores resultados utilizando os mesmos atributos.

2.6.1 Preparação dos dados

Primeiramente criou-se uma variável chamada **data** que irá conter os dados de treino e de teste. Esta variável foi criado com o objetivo de se obter os valores máximo e mínimos dos dados, quer dos dados de treino quer nos dados de teste. Com a obtenção dos valores máximo e mínimo foi possível fazer com que a escala varia-se entre [0;1], em todos os atributos presentes.

2.6.2 Modelação, Avaliação e Implementação

Primeira execução com a camada =c(10,50,50,50,10,10)

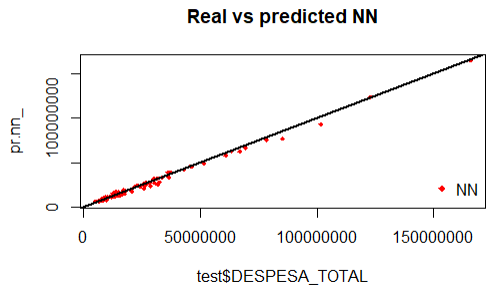

Figura 2.5: Gráfico resultante da primeira camada

e com um MSE= 5643588538023.05.

Segunda execução com a camada= c(10,50,50,50,10,20,1)

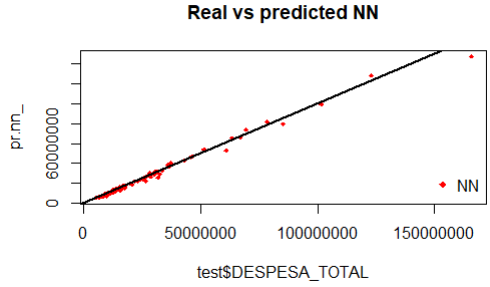

Figura 2.6: Gráfico resultante da segunda camada

e com um MSE=8352055699984.25.

Terceira execução com a camada= c(10,50,50,50,100,1)

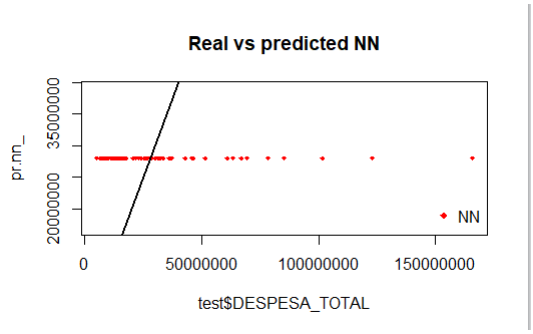

Figura 2.7: Gráfico resultante da terceira camada

e com um MSE=707001261084604.

Quarta execução com a camada= c(10,50,50,50,10,10)

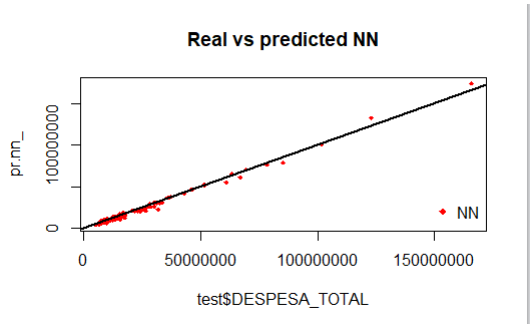

Figura 2.8: Gráfico resultante da quarta camada

e com um MSE=8141197833226.74.

Quinta execução com a camada= c(10,50,50,10,10)

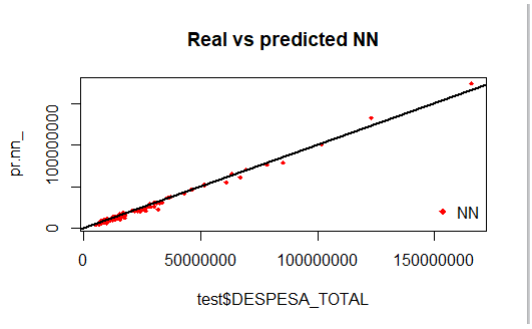

Figura 2.9: Gráfico resultante da quinta camada

e com um MSE=9299402215069.18.

Capítulo 3

Comparação entre os dois modelos

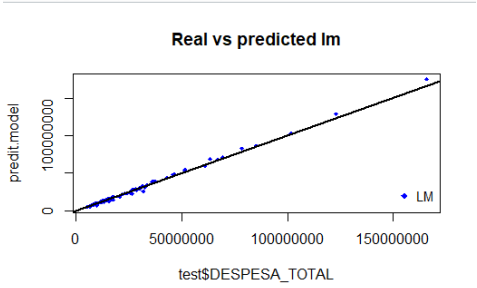

Figura 3.1: Flower one.

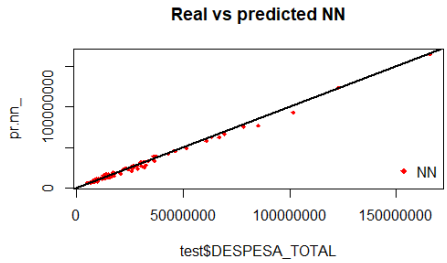

Figura 3.2: Flower two.

Quando visto os dois modelos é possível verificar que o modelo de regressão linear tem o modelo mais acertado, visto que existem mais pontos perto da reta do que existe no modelo de redes neuronais. Caso não fosse possível verificar tal coisa é sempre possível verificar pelo MSE onde o do modelo de regressão dá 4037487401648.12 enquanto o do modelo de redes neuronais dá 5643588538023.05, logo o do modelo de regressão linear para aqueles dados apresenta melhores resultados.

Capítulo 4

Conclusão e trabalho futuro

O termo *big date* está presente nos dias que correm muito devido ao facto de hoje em dia ser possível recolher informação praticamente de todo lado. O mais importante de ter muitos dados é conseguir extrair informação útil para o negócio.

Para se conseguir extrair informação útil dos dados é preciso que os mesmos estejam tratados corretamente, senão o que pode acontecer é que a previsão feita pelo modelo esteja errada o que pode levar a acontecimentos desastrosos para a empresa. Por isso é que a metodologia *CRISP-DM* é uma metodologia importante para quem trabalha na parte do *data mining*.

Antes sequer de se começar a analisar os dados é importante saber o que se vai fazer com eles e quem impacto isso vai trazer para o negócio. Passado isto é importante fazer um estudo pormenorizado dos dados, saber o seu domínio, se são numéricos, categóricos, binários, etc. Posto isto parte-se para a parte da preparação dos dados onde é selecionado os dados importantes para o modelo, se for caso é também nesta parte que há substituição de valores omissos, ou remoção, se for o caso balancear os atributos e tratar de valores fora do normal chamados *outliers*. Depois entra a parte de escolher um modelo,avalia-lo e por fim implementa-lo.

Para trabalho futuro fica tentar melhorar o modelo, por exemplo adicionar novos valores, retirar valores que estejam a mais e procurar novos modelos que apresentem melhores resultados.

Bibliografia

[1] *Redes Neuronais Artificiais*: César Analide, Paulo Novais e José Neves

[2] https://pt.wikipedia.org/wiki/Regress%C3%A3o_linear

[3] https://pt.wikipedia.org/wiki/Rede_neural_artificial