Universidade Federal do Paraná

Rafaela Barbirato Ferreira

Análise automatizada de exames de urina utilizando imagens digitais de *dipsticks*

Curitiba PR
2016

Rafaela Barbirato Ferreira

Análise automatizada de exames de urina utilizando imagens digitais de *dipsticks*

Trabalho apresentado como requisito parcial à conclusão do Curso de Bacharelado em Informática Biomédica, setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Ciência da Computação.*

Orientador: Lucas Ferrari de Oliveira.

Curitiba PR
2016

TERMO DE APROVAÇÃO

RAFAELA BARBIRATO FERREIRA

ANÁLISE AUTOMATIZADA DE EXAMES DE URINA UTILIZANDO IMAGENS DIGITAIS DE *DIPSTICKS*

Trabalho de Conclusão de Curso apresentado ao Curso de Informática Biomédica da Universidade Federal do Paraná como requisito à obtenção do título de bacharel em Informática Biomédica, pela seguinte banca examinadora:

Orientador: _____

Prof. Dr. Lucas Ferrari de Oliveira
Departamento de Informática, Universidade Federal do Paraná

_____

Prof. Dr. David Menotti
Departamento de Informática, Universidade Federal do Paraná

_____

Prof. Dr. Eduardo Jaques Spinosa
Departamento de Informática, Universidade Federal do Paraná

Curitiba, 31 de Janeiro de 2016.

Agradecimentos

A Deus por ter me dado saúde e força para superar as dificuldades.

A esta universidade e seu corpo docente que me fizeram crescer ao longo dessa jornada.

Ao laboratório Lanac que me recebeu de portas abertas para que eu pudesse desenvolver meu trabalho.

Ao meu orientador Lucas Ferrari de Oliveira, pelo suporte no pouco tempo que lhe coube, suas correções e incentivos.

A minha mãe Marcia e ao meu pai Alberto por todo apoio, paciência e amor incondicional.

Aos meus amigos que estiveram ao meu lado em todos os momentos.

E a todos que direta ou indiretamente fizeram parte da minha formação, o meu muito obrigada.

Resumo

Este trabalho propõe um estudo sobre a análise de fitas reagentes (*dipsticks*) para exames de urina. A uroanálise proporciona informações sobre patologias renais e do trato urinário, detecção de doenças sistêmicas e metabólicas que podem estar relacionadas não apenas com os rins. Usualmente o exame químico de urina é feito com tiras reagentes, objetivando tornar a análise mais rápida, simples e econômica. Porém, obter equipamentos especializados que analisam os *dipsticks* é algo dispendioso e possui limitações. Este trabalho propõe um algoritmo para análise das fitas reagentes utilizando um *scanner* padrão para a captura das imagens. Dessa forma pretende-se, através de técnicas simples de processamento de imagens, e de aprendizado de máquina, obter o diagnóstico de forma rápida e eficiente. Inicialmente montou-se uma base de imagens, possibilitando o desenvolvimento e avaliação das técnicas de extração propostas para a analise das fitas. A metodologia utilizada baseou-se, inicialmente, na conversão das imagens do modelo RGB para o modelo de cores HSV, e posteriormente na obtenção das médias normalizadas de cada canal H, S e V. Paralelamente propôs-se a utilização de mais seis valores para auxiliar na classificação, sendo os máximos e mínimos de cada um dos canais H, S e V de cada imagem; e a utilização do modelo RGB como alternativa, visando possíveis melhoras nos resultados. Os atributos foram organizados em formato de arquivo ARFF para que os testes pudessem ser realizados pela ferramenta *Weka*. Para realização dos testes selecionou-se três classificadores, visando avaliar a melhor abordagem para o problema de classificação. Utilizou-se os algoritmos *k-NN (k-Nearest Neighbors)*, *MLP (Multi-layer Perceptron)* e o *SVM (Support Vector Machine)*. Para cada classificador, pode-se manipular alguns de seus parâmetros, visando melhores resultados de classificação. No caso do *k-NN*, apenas o número de vizinhos mais próximos foi alterado. Para o *MLP*, variou-se o número de camadas intermediárias, suas respectivas quantidades de neurônios, e posteriormente, a taxa de aprendizado, momento e ciclos de treinamento. No caso do classificador *SVM*, a manipulação se deu na escolha do *kernel* e dos parâmetros gama, C e grau do polinômio, quando disponíveis. Os três classificadores mostraram-se eficientes na classificação dos dados, com média de percentuais de acerto próximas. O *k-NN* obteve uma média de $90,67\%$, a menor dentre os algoritmos. O *MLP*, com a segunda melhor média obteve $91,21\%$ de taxa de acerto, sendo o *SVM* o que apresentou melhor resultado para a proposta, com percentual de acerto de $92,24\%$. Os experimentos desenvolvidos confirmaram que a técnica é viável.

Palavras-chave: Uroanálise, Fitas Reagentes, Aprendizado de Máquina.

Abstract

This paper describes a study on the dipsticks analysis for urine tests. The urinalysis provides information about renal function, urinary tract infections, detection of systemic and metabolic diseases, which can be related not only to the kidney. The chemical examination of urine is done with reagent strips to enhance and simplify the process of urinalysis. Equipments for automatic analysis already exists, however, it is expensive and has limitations. For the purposes of this work, a database of images was elaborated to enable the development and evaluate the performance of the information extraction techniques used for the reagent strips analysis. The information extraction was based, firstly, on the transformation of the images from RGB into HSV color model and subsequent average normalization of the values of each H, S and V channels. It was also proposed to use six more values to help the classification, these were the maximum and minimum of the H, S and V channels of each image; and the use of the RGB model, alternatively, searching for improvments in the results. The attributes were organized in the ARFF file so that the tests could be performed by the Weka tool. For the tests, three classifiers were selected, in order to evaluate the best approach to the classification problem. The algorithms used were k-NN (k- Nearest Neighbors), MLP (Multi-layer Perceptron) and the SVM (Support Vector Machine). For each classifier, it is possible to manipulate some particular parameters, aiming better classification results. In the case of k-NN, the only parameter to be varied was the number of nearest neighbors. For MLP, the number of layers and the number of neurons were varied, and later the learning rate, momentum and epoch. In the case of the SVM classifier, the manipulation was focused on the choice of the parameters kernel, gamma, C and the degree of the polynomial in the case that it is available. The data classification, of the classifiers, was proved to be efficient with slight differences on their averages of success rate. The k-NN obtained an average of 90.67%, the lowest compared to the other algotrithms. The MLP with the second best mean obtained 91.21% of success rate and the SVM was the best classifier for the proposal, with a success percentage of 92.24%. Experiments have confirmed that the technique is viable.

Keywords: Urine Analysis, Reagent Strips, Machine Learning.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Símbolos

dL	decilitro
k-NN	*k-Nearest Neighbors*
mg	miligrama
MLP	*Multilayer Perceptron*
μL	microlitro
neg	classe negativa
norm	classe normal
RBF	*Radial Basis Function*
SVM	*Support Vector Machine*
γ	gama, terceira letra do alfabeto grego
θ	theta, limiar (*threshold*)

Capítulo 1

Introdução

Ao longo dos anos, o processo de análise do exame de urina evoluiu, com a finalidade de se obter resultados mais confiáveis e precisos. As técnicas desenvolvidas não envolvem apenas o exame físico, mas também o estudo através dos microscópios, da coloração de Gram e das tiras impregnadas com reagentes, ou ainda conhecidas como *dipsticks* [Reine and Langston, 2005, Bolodeoku and Donaldson, 1996].

Existem no mercado instrumentos que executam a leitura das fitas reagentes, eliminando a subjetividade do olho humano na leitura das mudanças de cor, melhorando assim o grau de precisão [Ravel, 1995]. Mas apesar dos avanços tecnológicos ocorridos nos últimos anos, e da análise das fitas reagentes tornarem o diagnóstico mais rápido, simples e econômico, este processo ainda peca em alguns pontos.

Os laboratórios possuem apenas duas alternativas quando o assunto é a análise dos *dipsticks*. Ou optam por realizar a leitura das fitas manualmente através da análise visual, comparando a amostra com tabela de cores padrão, ou utilizam equipamentos especializados para a análise. Devido ao custo destes equipamentos, muitos laboratórios ficam restritos apenas à análise manual realizada pelos profissionais, um processo sujeito a erros de diagnóstico devido à subjetividade do olho humano.

Outro fator a ser considerado foram as limitações apresentadas pelo equipamento utilizado na análise das fitas, o *Uriscan Pro II* $^{\text{TM}}$. A primeira delas está relacionada com o fato de que o equipamento emite o diagnóstico em papel térmico, que com o passar do tempo tem sua impressão apagada. A segunda é que, segundo relatos dos profissionais, o equipamento possui uma interface lenta e não prática, o que desmotiva o seu uso. Para solucionar ambos os problemas, os profissionais passam manualmente todas as informações para um sistema próprio do laboratório. Essas são repassadas para este sistema para posterior impressão dos resultados, a ser entregue aos pacientes, e para armazenamento da informação por um período de cinco anos, conforme exige a lei correspondente.

No presente trabalho será abordada uma nova proposta, que tem por finalidade transformar o trabalho dos laboratórios que não possuem o equipamento para a análise das urinas e mudar a rotina daqueles que já possuem a análise automatizada, dispensando equipamentos sofisticados.

1.1 Objetivos

Este trabalho propõe-se a desenvolver uma metodologia para a análise de fitas reagentes, baseada em processamento de imagens.

1.1.1 Objetivo geral

Montar uma base de imagens adquirida via *scanner*, desenvolver e avaliar o desempenho das técnicas propostas para a extração de informações e análise das fitas reagentes.

1.1.2 Objetivos específicos

- Montar uma base de dados para cada item analisado da fita;
- Realizar a detecção dos *dipsticks* e as respectivas posições das áreas reagentes;
- Extrair as informações das áreas reagentes;
- Realizar testes com a base de dados criada;

1.2 Estrutura do trabalho

No primeiro capítulo é feita uma introdução aos objetivos da dissertação, bem como uma rápida exposição sobre o processo de análise dos exames de urina e seus problemas.

No segundo capítulo são apresentados os conceitos da uroanálise e das substâncias analisadas pelo método. É feita também uma introdução ao aprendizado de máquina, bem como alguns dos seus classificadores selecionados, da ferramenta *Weka* utilizada nos testes, e uma breve descrição dos modelos de cores HSV e RGB.

No capítulo três são apresentados os trabalhos relacionados.

O capítulo quatro aborda técnicas propostas para a extração de informações das imagens e técnicas para a análise das fitas reagentes, bem como os materiais utilizados no desenvolvimento da metodologia.

No quinto capítulo são apresentados os resultados obtidos, visando a validação da metodologia proposta. E em seguida, no capítulo seis são apresentadas as conclusões, análise final e sugestões.

Capítulo 2

Fundamentação

2.1 Uroanálise

No início da medicina o exercício da profissão ficava restrito apenas a observação e exame físico do paciente. Já os estudos laboratoriais apenas podiam ser realizados através das substâncias eliminadas pelo paciente, dentre elas a urina. Acredita-se que um dos procedimentos laboratoriais mais antigos, que tem sido utilizado desde o século XIX para diagnóstico de doenças, é a análise da urina [Strasinger, 2000, Amorim et al., 2009].

A análise da urina, ou conhecida também como uroanálise, foi o grande começo da medicina laboratorial. Referências ligadas ao estudo da urina foram encontradas em desenhos dos homens das cavernas e nos hieróglifos egípcios que representavam os médicos examinando um frasco de urina. Essa análise da urina era feita através de observações básicas, mas que traziam informações diagnósticas como a cor, turvação, odor e até mesmo a presença de açúcar em certas amostras, observada pela aproximação de formigas e demais insetos [Bolodeoku and Donaldson, 1996, Strasinger and Torquettitoloi, 1996].

A uroanálise foi definida pela Associação Brasileira de Normas Técnicas (ABNT, 2005, p.1) como "exame realizado numa amostra de urina humana para determinar os caracteres físicos e químicos e para verificar a presença de elementos figurados ou de outra origem."[ABNT, 2005].

Esse exame é o terceiro mais realizado em laboratórios clínicos e pode ser obtido sem nenhum procedimento invasivo. O mesmo fornece informações importantes, tanto no que diz respeito ao diagnóstico e monitoramento de doenças renais ou do trato urinário, quanto para a detecção de doenças sistêmicas e metabólicas que podem estar relacionadas não apenas com os rins [Chien et al., 2007, Strasinger, 2000].

Desde quando a uroanálise foi introduzida como exame de rotina em 1827, pelo médico Richard Bright, houve muitos progressos no que diz respeito à facilidade de coleta, simplicidade de execução do exame e quantidade de informações a serem obtidas. Informações estas que podem estar relacionadas a diversas atividades metabólicas e processos patológicos do organismo, sejam estes fisiológicos ou anatômicos [Strasinger, 2000]. O fato da urina ter uma aparência completamente normal não significa que ela não possa conter alterações. Mesmo a presença de sangue ou proteína, por exemplo, pode ser apenas microscópica, não sendo possível a sua identificação por qualquer outro meio que não através do exame laboratorial.

A partir do século XX, com o desenvolvimento do conhecimento científico-tecnológico obtido ao longo das décadas, a análise do exame de urina evoluiu, com a finalidade de se obter resultados cada vez mais confiáveis e precisos, isto é, com maior

poder de exclusão ou inclusão de doenças. Dentre as diversas técnicas desenvolvidas, a mais utilizada hoje é a das tiras impregnadas com reagentes [Reine and Langston, 2005, REPRESENTANTES, 2004].

O exame mais realizado na urina é denominado de Exame de Rotina da Urina, ou conhecido também como EAS (Elementos Anormais e Sedimentares). O mesmo é divido em três etapas. A primeira etapa consiste no teste físico, realizado a olho nu. Analisam-se as características gerais da urina, como sua coloração, aspecto e cheiro. Na segunda etapa é feita a pesquisa de elementos anormais, que corresponde à pesquisa química feita na urina. A terceira e ultima etapa é feita a sedimentoscopia, que corresponde ao exame microscópico da urina, compreendendo a observação, identificação e quantificação do material insolúvel presente na amostra [Lopes, 2004]. O método químico de tiras reagentes é utilizado como uma triagem das amostras de urina para a determinação do pH, da densidade e para a pesquisa de elementos anormais, que fazem parte do protocolo do exame de urina de rotina e que compõem a segunda etapa desse processo.

A tira é constituída por um suporte plástico contendo áreas impregnadas com reagentes químicos, conforme mostra a Figura 2.1. Quando as áreas de química seca entram em contato com a urina, uma reação de cor se desenvolve a qual permite a interpretação quase que imediata desses parâmetros químicos, em cerca de um a dois minutos [Lima et al., 1992].

Figura 2.1: Fita reagente da marca *Uriscan*™.
Fonte: o autor.

A fita apresentada na Figura 2.1 possui doze áreas, mas sendo a última apenas uma área chamada de compensação, para indicar o final da leitura da fita. Logo, a mesma analisa no total onze parâmetros, que serão descritos na próxima seção.

2.2 Substâncias analisadas

As fitas reagentes utilizadas nesse trabalho são da marca *Uriscan*™. A mesma analisa onze elementos, que são descritos a seguir com seus respectivos significados clínicos [Lopes, 2004].

Bilirrubina é um analito importante na suspeita de doenças hepáticas e na investigação das causas de icterícia, que é o nome dado a alteração da coloração para amarelo dos olhos, da pele e de mucosas.

Urobilinogênio é filtrado pelos rins e excretado na urina em uma concentração aproximada de $1,0$mg/dL. Sua presença em níveis maiores pode indicar hepatopatias, que são doenças do fígado, e distúrbios hemolíticos.

Cetonas são o resultado da metabolização das gorduras, ou seja, os mesmos são produzidos quando o organismo possui dificuldade em obter a energia através da glicose. Comumente a produção de cetonas é baixa e não estão presentes na urina. Sua detecção pelo *dipstick* pode ser a causa de exercícios físicos intensos, estados febris, jejum prolongado ou até mesmo dietas rigorosas.

Proteínas presentes no sangue são, em sua maioria, grandes demais para serem filtradas pelos rins e, por isso, não é comum detectar sua presença na urina. Quantidades não significativas de proteínas na urina podem ser resultado de diversas causas simples, tais como presença de febre, desidratação ou estresse emocional, até causas mais graves como doença renal e infecção urinária. Porém, a presença de grandes quantidades na urina quase sempre indicam alguma doença relacionada aos rins.

Sangue assim como as proteínas, possui uma quantidade desprezível na urina e por isso não consegue ser detectado pelo exame da fita. Logo, o normal é que não haja presença do analito. Sua detecção pelo *dipstick* indica hematúria e pode ocorrer por diversos fatores como infecção, pedras nos rins, tumores de bexiga ou renal, doenças renais graves, ou até mesmo um falso positivo que pode acontecer nas mulheres que colheram a urina enquanto estavam no seu período menstrual.

Nitrito é proveniente da metabolização do nitrato. A urina é rica em nitratos e quando há presença de bactérias na mesma, esses nitratos são transformados em nitritos. Porém nem todas as bactérias são capazes de metabolizar o nitrato, logo, uma urina positiva para nitritos é uma indicação indireta da presença de bactérias. A presença desse analito na urina pode auxiliar no diagnóstico de infecção urinária.

Glicose filtrada pelos rins é reabsorvida de volta para o sangue, por isso, o normal é não haver evidências da mesma na urina. Sua presença pode ser um indicativo de doença nos túbulos renais, o que significa que apesar de não haver excesso de glicose na urina, os rins não conseguem impedir sua perda. A glicose também ajuda no controle e diagnóstico precoce da Diabetes Mellitus, se associada a presença de elevadas taxas de glicose no sangue.

pH é utilizado no diagnóstico de distúrbios eletrolíticos sistêmicos de origem metabólica ou respiratória. Tendo em vista que o rim é o principal meio de excreção dos ácidos do organismo, a urina é naturalmente ácida, variando entre valores próximos de $6,0$. Valores maiores do que $6,0$ podem indicar presença de bactérias que alcalinizam a urina. Nesse caso, pode-se ter um quadro de infecção urinária.

Densidade é utilizada na avaliação do estado hídrico do paciente. O valor de densidade indica o nível de concentração de substâncias sólidas diluídas presentes na urina. Sabendo-se que a água pura possui densidade igual a 1000 mg/dl, a urina que tem seu valor próximo a este significa que ela está diluída. Os valores normais de densidade concentram-se entre 1005 mg/dl e 1035 mg/dl. Urinas com valores próximos a 1035 mg/dl estão muito concentradas, ou seja, indicam a alta concentração das substâncias sólidas diluídas.

Leucócitos são as células de defesa do organismo, por isso, sua pesquisa é útil para diagnóstico de processos infecciosos ou com outros processos inflamatórios do trato urinário. Logo, uma urina normal tem resposta negativa para os leucócitos.

Ácido Ascórbico é um analito de importante análise por dois motivos. Primeiro, pois ele pode alterar os resultados da fita reagente, principalmente na detecção de hemoglobina e glicose. E segundo porque resultados inesperados podem ser falsos negativos ou positivos devido a vitamina C.

As tiras trazem resultados semiquantitativos, que podem ser representados por cruzes, ou ainda através de estimativas, como é possível observar na Figura 2.2. As estimativas são representadas através de escalas em mg/100mL, WBC/μL - (*White Blood Cells*) para a contagem de Leucócitos ou em RBC/μL - (*Red Blood Cells*) para contagem de hemácias no sangue.

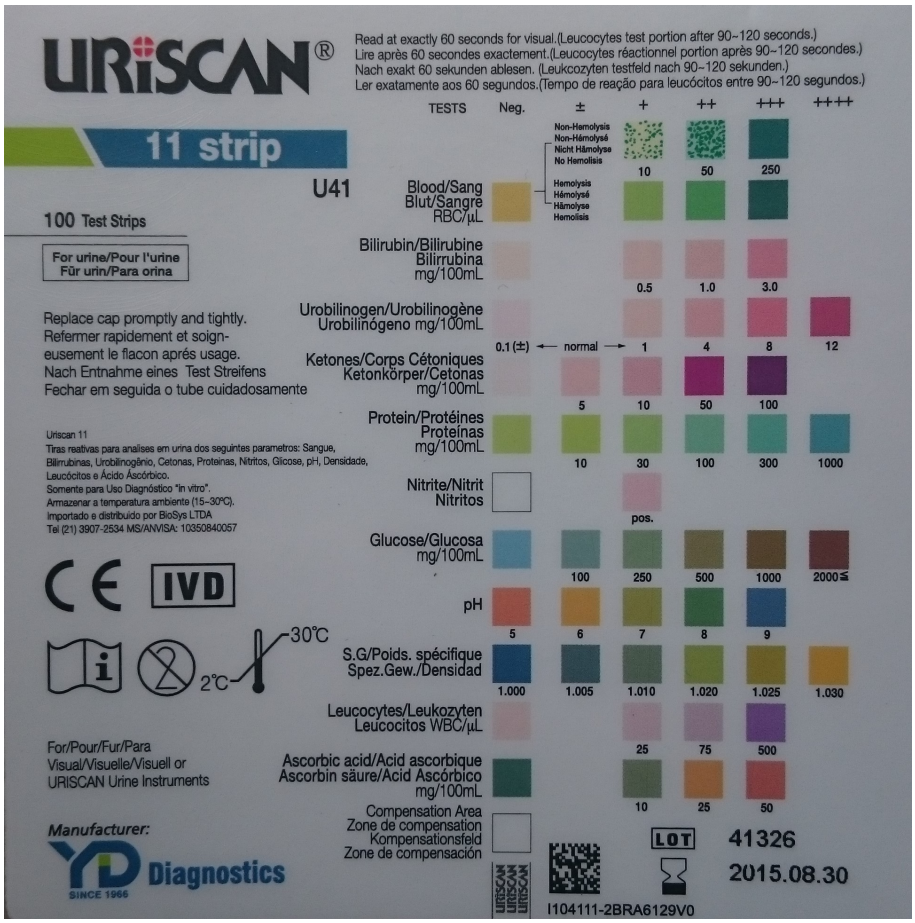

Figura 2.2: Padrões para os parâmetros químicos da marca *Uriscan*$^{\text{TM}}$.
Fonte: o autor.

É necessário observar que apesar de simples, o exame químico qualitativo realizado utilizando-se as fitas reagentes pode dar muitas respostas quando o assunto é o diagnóstico. Apesar da sua análise não ser cem por cento precisa, devido a erros durante a coleta da urina e a elementos que podem mascarar outros, como a menstruação para as hemácias e a presença de ácido ascórbico para a glicose, ela pode dar muitas informações relacionadas não só ao sistema urinário, mas também ao organismo como um todo.

2.3 Aprendizado de máquina

É uma sub-área da inteligência artificial que tem por objetivo desenvolver técnicas computacionais capazes de aprender, ou seja, de adquirir conhecimento. Um sistema de aprendizado nada mais é do que um programa computacional que toma decisões baseado em experiências obtidas ao longo de soluções bem sucedidas de problemas anteriores [Monard and Baranauskas, 2003]. Mesmo o aprendizado de máquina sendo uma ferramenta poderosa na aquisição automática do conhecimento, é necessário saber que não há

um único algoritmo que tenha os melhores resultados para todos os problemas. Logo, é preciso observar as limitações e a metodologia dos diversos algoritmos para sua aplicação em problemas específicos.

A indução é um tipo de inferência lógica que propicia a obtenção de conclusões genéricas de um conjunto de exemplos. A ideia é generalizar o raciocínio obtido através de um conceito específico. A generalização de um classificador é definida como a capacidade de prever de maneira correta o rótulo de novos dados. Quando o modelo se especializa nos dados utilizados na fase de treinamento e apresenta baixas taxas de acerto quando confrontado com novos dados, tem-se o que se chama de super-ajustamento ou *overfitting*. Quando as taxas de acerto são baixas ainda na fase de treinamento, configura-se uma condição de sub-ajustamento ou *underfitting*. Isso pode ocorrer devido a baixa representatividade dos dados da base de treinamento ou quando o modelo, ou hipótese, obtido é muito simples [Monard and Baranauskas, 2003].

O aprendizado através da indução é feito a partir do raciocínio sobre exemplos fornecidos por um processo externo ao sistema de aprendizado. Esse tipo de aprendizado divide-se em supervisionado e não-supervisionado [Monard and Baranauskas, 2003].

A aprendizagem supervisionada é aquela onde os dados possuem um atributo classe, que especifica a classe à qual uma determinada instância pertence. Existem diversos métodos que trabalham com esse tipo de aprendizado, dentre eles, as técnicas preditivas são comumente usadas, pois elas tentam predizer a classe de um exemplo ainda não visto com base nos exemplos utilizados no treinamento. A partir de um subconjunto de atributos, presente no conjunto de dados, os classificadores podem determinar o rótulo de um exemplo. Para rótulos de classes não numéricas, ou discretas, o problema é conhecido como classificação. Já para valores contínuos, o problema é compreendido como regressão [Damasceno, 2015].

Em contrapartida, no aprendizado não supervisionado os exemplos fornecidos são analisados e agrupados de maneira tal que o grau associativo entre os elementos do mesmo grupo é alto e entre os grupos diferentes é baixo. A ideia é encontrar a estrutura particular dos dados, compondo-os em grupos ou *clusters* [Backer, 1995, Damasceno, 2015]. A clusterização é um método onde a análise exploratória dos dados é utilizada para auxiliar na classificação.

O tipo de aprendizado abordado neste trabalho é o supervisionado. Logo, dado um conjunto de exemplos rotulados na forma (x_i, y_i), onde x_i representa a instância e y_i o seu rótulo, a ideia é obter um modelo ou hipótese. Esse modelo é que será responsável e capaz de predizer o rótulo de novos dados. Esse processo onde um classificador é induzido a partir de uma amostra de dados é denominado treinamento. Tal metodologia pode ser aplicada através da utilização de uma aplicação chamada *Weka*, que será abordada a seguir.

2.4 Weka

O *Weka*, ou *Waikato Environment for Knowledge Analysis* é formado por um conjunto de implementações de algoritmos de diversas técnicas de aprendizado de máquina com uma interface gráfica e intuitiva, suportando mais classificadores com o auxílio de bibliotecas externas, como é o caso do *SVM*, visto mais adiante [University of Waikato, 2016].

Para a utilização desta aplicação é necessário que os dados estejam de forma organizada em um determinado formato. O formato utilizado pelo *Weka* é o ARFF (*Attribute-Relation File Format*) ou Formato de Arquivo Atributo-Relação. A figura 2.3

representa um exemplo desse tipo de arquivo, que contém duas partes. A primeira parte contém o nome da relação, com a marcação @relation, seguida dos atributos (@attribute) com um nome e seus respectivos domínios. No exemplo da figura os atributos foram nomeados com valores de 0 a 5 e podendo assumir valores apenas do tipo real. Na última linha de representação dos atributos (@attribute class) são descritas as possíveis classes as quais os dados podem pertencer, que nesse caso são $(10, 25, 50, \text{neg})$.

A segunda parte compõem os dados em si e inicia-se a partir da marcação @data. Cada instância é representada por meio dos atributos separados por vírgula, que para o dado exemplo são 6. Quando o tipo de aprendizado é supervisionado, a classe a qual a instância pertence é representada como último atributo. Enquanto que para o aprendizado não supervisionado, a classe não é representada.

```
@relation Ascorbic_pattern
@attribute 0 real
@attribute 1 real
@attribute 2 real
@attribute 3 real
@attribute 4 real
@attribute 5 real
@attribute class  {10, 25, 50, neg}

@data
135,73,117,8,155,109,10
135,93,68,19,152,116,10
159,65,75,11,201,116,10
```

Figura 2.3: Exemplo do padrão de arquivo ARFF.
Fonte: o autor.

2.5 Classificadores

Foram selecionados três classificadores para os testes. O intuito foi de se avaliar a melhor abordagem para o problema da classificação das imagens das tiras reagentes, previamente processadas:

- ***k-NN (k-Nearest Neighbors)***
- ***MLP (Multi-layer Perceptron)***
- ***SVM (Support Vector Machine)***

Cada uma das técnicas será abordada em mais detalhes a seguir.

2.5.1 *k - Nearest Neighbors (k-NN)*

O algoritmo *Nearest Neighbor* foi proposto por Cover e Hart em 1967 [Cover and Hart, 1967]. É simples conceitualmente, de fácil implementação, pois não possui processamento na fase de treinamento e é uma das primeiras opções para estudo de classificação quando há pouco ou nenhum conhecimento prévio sobre a distribuição dos dados [Peterson, 2009].

O classificador de vizinhos mais próximos faz sua previsão baseada em informações locais. A sua ideia geral baseia-se em encontrar os k exemplos mais próximos do exemplo que ainda não foi classificado e, baseado nos rótulos desses exemplos mais próximos, decidir qual será a classe relativa ao exemplo que está sendo avaliado [Ferrero, 2009]. Na Figura 2.4 temos um exemplo de classificação do ponto vermelho, que deve ser classificado como a classe de quadrados azuis ou como a classe de triângulos verdes. Se $k = 3$, representado pelo círculo de linha contínua, o ponto será atribuído à classe de triângulos porque há 2 exemplos dessa classe e apenas 1 da classe quadrado. Se $k = 5$ (círculo tracejado) o ponto será atribuído à classe dos quadrados, pois há 3 quadrados e apenas 2 triângulos dentro do círculo externo.

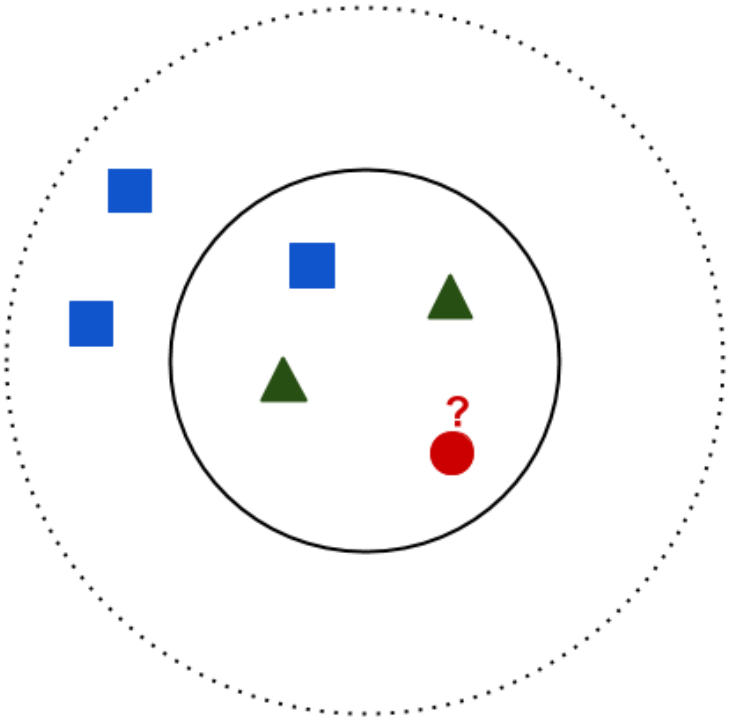

Figura 2.4: Exemplo de classificação com o *k-NN*.
Fonte: o autor.

Esse tipo de classificador representa cada exemplo como um ponto dado em um espaço d-dimensional, onde d é o número de atributos. Dado um exemplo de teste, calcula-se sua proximidade com o resto dos pontos de dados no conjunto de treinamento. O problema do algoritmo está em ser suscetível a ruídos para valores pequenos de k e a inclusão de vizinhos distantes ao exemplo de teste em questão para valores muito grandes de k [Kramer, 2013].

Classificar um exemplar desconhecido usando o algoritmo *k-NN* consiste em primeiramente determinar o grau de similaridade entre o exemplo em questão e os outros exemplos do conjunto de treinamento. Para tal, existem diversas formas, entre as quais estão as medidas de distância.

Normalmente quando o conjunto de dados é descrito por atributos numéricos, as medidas de distância são comumente utilizadas no cálculo do grau de similaridade, onde a menor distância representa a maior similaridade. Sabendo-se que os exemplos são descritos através de ponto no espaço multi-dimensional, Minkowsky definiu uma maneira

genérica de calcular a distância entre dois pontos nesse espaço de acordo com o parâmetro d, equação 2.1 [Kramer, 2013]:

$$dist(p,q) = (\sum_{i=1}^{n} |p_i - q_i|^d)^{\frac{1}{d}} \tag{2.1}$$

Quando $d = 1$ a medida é conhecida como distância de Manhattan, e quando $d = 2$, ela define a distância Euclidiana [Ferrero, 2009]. Obtidas as distâncias, a meta é classificar o exemplo desconhecido atribuindo a ele o rótulo representado mais frequentemente dentre as k amostras mais próximas, utilizando-se o voto majoritário entre os rótulos de classe.

Esse algoritmo é encontrado no *Weka* com o nome *IBk*, no conjunto de classificadores denominados *lazy*.

2.5.2 *Multilayer Perceptron (MLP)*

A Rede Neural Artificial é um dos ramos da Inteligência Artificial composta por sistemas que, de certa forma, lembram a estrutura do cérebro humano. As Redes neurais artificiais são modelos matemáticos compostos por unidades de processamento chamadas de neurônios artificiais, que são responsáveis pelo cálculo de funções matemáticas [Falcão et al., 2013].

A arquitetura ou topologia dessas unidades podem variar de acordo com o número de camadas ou com os tipos de conexão. A rede pode ter camada única ou múltiplas camadas. Quanto aos tipos de conexão, a topologia pode ser do tipo acíclica (*feedfoward*), que são redes onde suas camadas estão organizadas em uma ordem e os neurônios de uma determinada camada estimulam apenas os neurônios das camadas posteriores, ou cíclica (*feedback*), onde os neurônios podem estimular outros da sua camada ou de camadas anteriores. Na maioria dos modelos, essas conexões da rede normalmente estão associadas a um peso, que têm por finalidade armazenar o conhecimento adquirido no processamento e servem como ponderação na entrada recebida pelos neurônios da rede [Falcão et al., 2013].

Essa técnica chama atenção por ter a competência de predizer informações mais sutis a respeito da distribuição dos dados, quando comparado a métodos estatísticos tradicionais, e pela sua capacidade de transformar limiares de decisão não-lineares em um espaço de características [Silva, 2005]. A ideia principal desse modelo é extrair combinações lineares dos exemplos de entrada como características derivadas, e utilizar essas características como base para a modelagem dos novos dados [Friedman et al., 2001].

A instituição do neurônio como uma estrutura básica do cérebro se deu em 1911 por Santiago Ramon y Cajal. Essa estrutura foi definida como a responsável por enviar e receber informações, que se dão através de sinais elétricos, conhecidos como impulsos nervosos. Logo, pode-se entender o neurônio como um dispositivo que possui muitas entradas e apenas uma saída [Falcão et al., 2013].

Baseando-se nisso, Waren McCulloch e Walter Pitts propuseram em 1943 um modelo matemático artificial de um neurônio biológico, conforme a Figura 2.5 [McCulloch and Pitts, 1943].

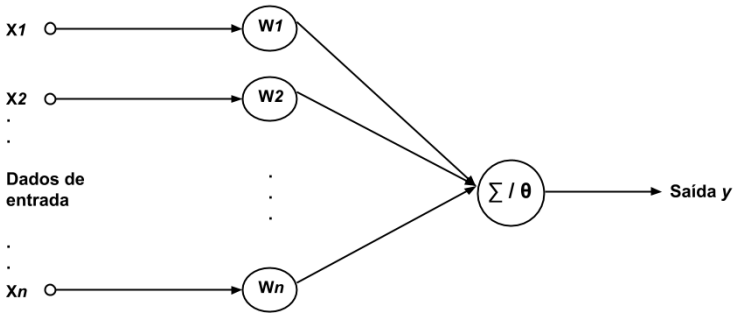

Figura 2.5: Modelo de um neurônio artificial segundo a proposta de McCulloch e Pitts. Adaptado de [McCulloch and Pitts, 1943].

Esse modelo é composto por terminais de entrada $(x_1, x_2, ...x_n)$, pesos sinápticos $(w_1, w_2, ...w_n)$, um somatório $(\sum)$, um limiar θ e um sinal de saída y. Os terminais são alimentados por informações de entrada da rede ou por sinais de saídas provenientes de outros neurônios. Essas informações são ponderadas por meio dos pesos sinápticos fixos da rede e que, posteriormente, são utilizadas pelo somatório. O cálculo do somatório é feito a partir dos valores de $x_i w_i$ que serão comparados com um limiar θ. Se esse somatório for igual ou superior ao limiar, o sinal de saída (y) recebe valor 1, caso contrário a saída receberá valor 0 [McCulloch and Pitts, 1943].

No decorrer dos anos, muito discutiu-se sobre o aprendizado tanto das redes biológicas quanto das artificiais. As novas propostas tinham como base que o aprendizado dos neurônios biológicos dava-se através do reforço das conexões sinápticas entre neurônios em estado de excitação. Logo, em redes artificiais o aprendizado seria dado por conta das variações dos pesos sinápticos, que antes eram fixos [Falcão et al., 2013].

Assim, no final da década de 1950, Frank Rosenblatt definiu um novo modelo de rede neural, o *Perceptron*. Esse modelo era composto por uma camada e demonstrou que os nodos da rede de McCulloch e Pitts dotados de pesos sinápticos ajustáveis poderiam ser treinados, com o objetivo de classificar certos tipos de padrões. Esse modelo é o mais simples tipo de rede neural, conhecido como um classificador linear. Isso significa que os tipos de problemas solucionados por este tipo de rede devem ser linearmente separáveis [Falcão et al., 2013].

O *Multilayer Perceptron* ou *MLP* é um *perceptron* de várias camadas, como sugere seu nome, com conexões do tipo *feedfoward*. Essas camadas são intermediárias, ou seja, estão entre a camada de entrada e a de saída. Esse modelo foi proposto para resolver problemas complexos e não lineares, os quais não poderiam ser resolvidos com o modelo básico de neurônio [Falcão et al., 2013]. A Figura 2.6 é um modelo de uma *MLP* com três entradas, duas camadas intermediárias com quatro neurônios e uma camada de saída com um neurônio.

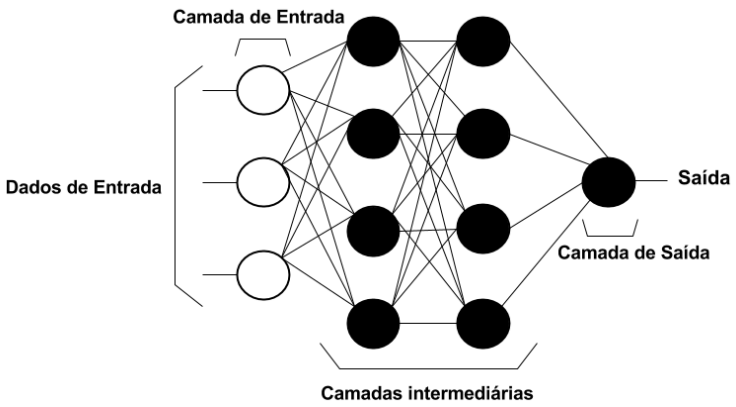

Figura 2.6: Modelo de uma *MLP* com duas camadas intermediárias.
Adaptado de [Falcão et al., 2013]

Os neurônios das redes *MLP* diferem-se do modelo de Waren McCulloch e Walter Pitts por não obter apenas um sinal de saída binário gerado pelo somatório, mas sim, por permitirem a saída de qualquer valor. Dessa forma, foram desenvolvidos modelos de neurônios artificiais onde os valores ponderados da entrada sofrem a aplicação de uma função de ativação, ou também conhecida como *bias* [Fiorin et al., 2011]. A Figura 2.7 é um exemplo de um neurônio genérico dentro da estrutura de uma rede *MLP*.

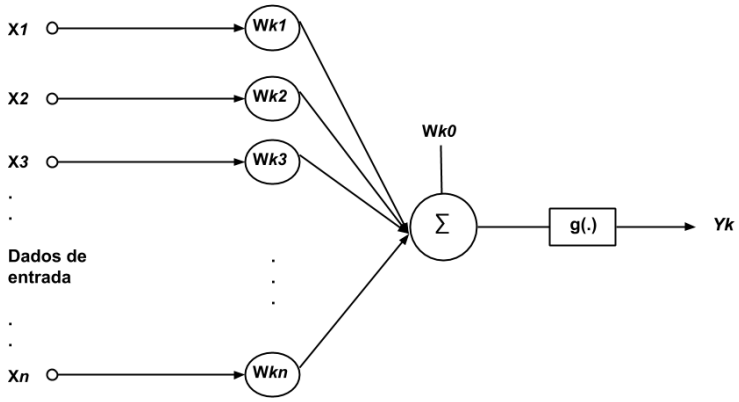

Figura 2.7: Modelo de um neurônio artificial não linear de uma rede *MLP*.
Adaptado de [Simões, 2000].

Onde:

$x_1, x_2, ...x_n$ é o vetor de entrada no neurônio k;

w_{ki} é o peso sináptico da entrada i do neurônio k;

w_{k0} é o *bias*, peso sináptico da entrada de polarização do neurônio k;

$g(.)$ é a função de ativação do neurônio;

y_k é a resposta ou saída do neurônio k.

Nesse modelo, o *bias* (w_{k0}) tem o propósito de expandir ou reduzir a entrada líquida da função de ativação $g(.)$. Essa função é responsável por produzir a saída y_k do

neurônio k e pode ser definida como do tipo reta, degrau, sigmóide ou tangente hiperbólica [Simões, 2000].

O treinamento nesse tipo de rede se dá na correção dos pesos sinápticos, para que as relações entre as características e as classes sejam estabelecidas de tal maneira que haja a melhor discriminação possível entre os diferentes padrões [Souza, 1999]. Tratando-se das redes de *perceptron* de Múltiplas Camadas, o treinamento é feito através do algoritmo de retropropagação do erro ou *error backpropagation* proposto por Rumelhart, Hinton e Williams em meados da década de 1980 [Rumelhart et al., 1985].

O aprendizado em uma rede de retropropagação é realizado através da alteração dos pesos sinápticos entre as conexões das camadas baseado em um fator de correção. Esse algoritmo possui duas etapas de execução. Na primeira delas, tem-se a propagação dos dados de entrada através das camadas até a obtenção dos resultados de saída. Em um segundo momento, compara-se os resultados obtidos na saída com os resultados esperados. Caso esses resultados sejam divergentes, é calculado um fator de correção que é retro-alimentado pela rede. Essa retroalimentação visa o ajuste dos pesos sinápticos para a obtenção de uma resposta mais próxima da desejada. O treinamento, com este algoritmo, leva um número maior de etapas para funcionar corretamente devido ao *warm-up* necessário para corrigir os pesos [Silva, 2005].

Na realização do treinamento, existem diversos parâmetros que devem ser ajustados. Dentre esses parâmetros, os mais importantes são a taxa de aprendizado, momento, quantidade de camadas escondidas e de neurônios em cada camada e os ciclos de treinamento ou *epochs*.

Os ciclos de treinamento definem o número de vezes que os dados do treinamento serão apresentados à rede. A taxa de aprendizado fornece a relevância da mudança dos pesos sinápticos entre os ciclos. Isso significa que uma taxa baixa apresenta um aprendizado lento, porém mais estável, enquanto que uma taxa alta fornece um aprendizado mais rápido, só que mais instável [Haykin, 2001].

Tendo a taxa de aprendizado fixa, pode-se utilizar o parâmetro momento como um meio de tornar a aprendizagem rápida e livre do perigo de instabilidade. Esse parâmetro tem relação com a variação dos pesos sinápticos provenientes do ciclo anterior. Isso significa que quanto maior for a variação dos pesos mais longe a rede está do resultado esperado, e logo, maior será o incremento oferecido pelo momento na velocidade do aprendizado [Haykin, 2001].

Com relação a quantidade de camadas escondidas, não é recomendado que se utilize um grande número. Isso porque a cada vez que o erro obtido no treinamento é utilizado para atualizar os pesos sinápticos da camada anterior, ele se torna menos preciso ou útil quando existem muitas camadas. Tratando-se da quantidade de neurônios nas camadas escondidas, esse parâmetro provê a capacidade de aprendizagem da rede. Na manipulação desse parâmetro é importante avaliar a quantidade de neurônios, pois uma quantidade baixa pode tornar o aprendizado difícil e uma quantidade muito alta pode deixar o modelo propenso ao aprendizado de ruídos dos dados e, assim, apresentar baixo poder de generalização [Haykin, 2001].

O *Multilayer Perceptron* está disponível no *Weka* no agrupamento de classificadores chamado *functions*.

2.5.3 *Support Vector Machine (SVM)*

Máquinas de Vetores de Suporte *(SVMs - Support Vector Machines)* é uma técnica de aprendizado que se destaca pelos resultados obtidos, comparáveis e, às vezes, superiores aos obtidos por outras técnicas, como as redes neurais artificiais [Joachims, 1998]. Essa técnica tem tido sucesso em diversas áreas como na categorização de textos [Joachims, 2002], na bioinformática [Noble et al., 2004] e na análise de imagens [Pontil and Verri, 1998].

O *SVM* é um algoritmo de aprendizado de máquina que aplica diretamente os conceitos da teoria do aprendizado estatístico, desenvolvida por Vapnik [Vapnik and Vapnik, 1998]. Essa teoria estabelece condições matemáticas que devem ser seguidas, com o objetivo de obter-se classificadores com boa capacidade de generalização.

O problema de classificação pode estar restrito a um problema binário. Nessa situação o objetivo é separar as duas classes através de uma função induzida pelos elementos de treinamento, chamada de *kernel*. Se tivermos um *kernel* linear e p pontos, pode-se ter então $(p-1)$ hiperplanos que irão separá-lós. Considerando a Figura 2.8, dentre os hiperplanos existentes, somente um maximiza a margem, ou seja, somente um hiperplano define o maior distanciamento entre as duas classes. Esse hiperplano é chamado de hiperplano de separação ótimo, e espera-se que ele possua uma generalização melhor que os outros possíveis.

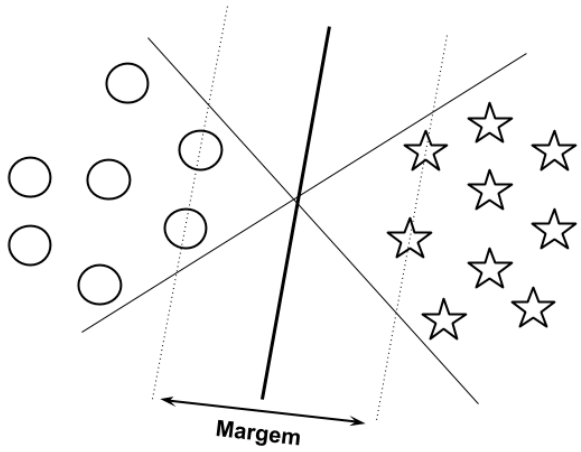

Figura 2.8: Hiperplano de separação ótimo.
Fonte: o autor.

Uma função linear pode não ser o suficiente para a tarefa de separação de dados, uma vez que muitos problemas acabam não sendo linearmente separáveis. Nesses casos, com o intuito de melhorar a separabilidade linear, estendeu-se a definição dos hiperplanos por meio do artifício chamado de *kernel trick*. A ideia desse *kernel* consiste em criar funções que mapeiam o espaço original em um espaço de maior dimensão denominado espaço de características, onde os dados passam a ser linearmente separáveis.

Apesar do *SVM* apresentar um bom poder de generalização, é de grande importância para o seu desempenho a seleção do *kernel* e de seus parâmetros particulares. Dentre os *kernels* existentes, a seguir são descritos três deles, onde $k(x, y)$ é a definição do *kernel* para os vetores de características x e y [Sklearn, 2016]:

- ***RBF***, ou *Radial Basis Function*, é um kernel do tipo Gaussiano e pode ser expresso da seguinte forma $k(x, y) = \exp(-\gamma\|x - y\|^2)$, onde γ (gama) deve ser maior do que

0 e $\|x - y\|^2$ pode ser reconhecido como o quadrado da distância Euclidiana entre os dois vetores de características x e y.

- **Polinomial**, é expresso por $k(x, y) = (x^\top y + c_0)^d$, onde d é o grau do polinômio e c_0 é um parâmetro livre que equilibra a influência dos termos de ordem superior nos termos de ordem mais baixa no polinômio.

- **Linear**, é o mais simples dos kernels disponíveis e pode ser expresso por $k(x, y) = x^\top y$. O *kernel* linear é um caso especial do *kernel* polinomial, em que a dimensão é definida como 1 e o termo independente $c_0 = 0$.

Além do *kernel*, a escolha adequada dos parâmetros podem resultar em melhoras na acurácia dos resultados. Os parâmetros de maior influência são [Sklearn, 2016]:

- **C** ou custo, é conhecido também como parâmetro de margem e é comum a todos os *kernels* do *SVM*. Esse parâmetro pode tanto aumentar quanto reduzir a penalidade para os erros de classificação. Ele determina um ponto de equilíbrio razoável entre a maximização da margem e a minimização do erro de classificação. Um alto valor para C atribui uma penalidade maior para uma classificação incorreta, reduzindo a quantidade de erros. Um baixo valor para C maximiza a margem de modo que o hiperplano é menos sensível a erros no conjunto de aprendizado.

- **Gama** pode ser visto como o inverso do raio de influência de amostras selecionadas pelo modelo. Esse parâmetro define quão longe a influência de um único exemplar do treino pode atingir na classificação. Com alto valores de gama, os exemplos mais próximos serão afetados. O mesmo está disponível apenas para os *kernels* RBF e polinomial.

- **Grau do polinômio** define o grau do polinômio utilizado pelo *kernel* polinomial.

No *Weka*, o *SVM* está disponível através da instalação da biblioteca externa *LibSVM*, no agrupamento de classificadores chamado *Functions*. O procedimento de instalação está descrito em [Chang and Lin, 2016].

2.6 Modelo de Cores

Os sistemas de cores são uma representação n-dimensional da imagem em que cada *pixel* configura um ponto, que possui, geralmente, três dimensões.

2.6.1 RGB

O modelo RGB descreve a cor como a composição de três cores primárias, o vermelho (*Red*), o verde (*Green*) e o azul (*Blue*). Por ser um modelo aditivo, a partir da mistura das diferentes intensidades de cada canal é possível formar outras cores. Nesse modelo, os valores para cada componente variam entre 0 e 255.

A representação tri-dimensional do seu espaço é feita em um cubo unitário, onde a origem (0,0,0) representa o preto e o valor (1,1,1) o branco, conforme a Figura 2.9. Como a diagonal do sólido é constituída por contribuições equivalentes das três cores primárias, ela compõem os tons de cinza [Oliveira et al., 2010].

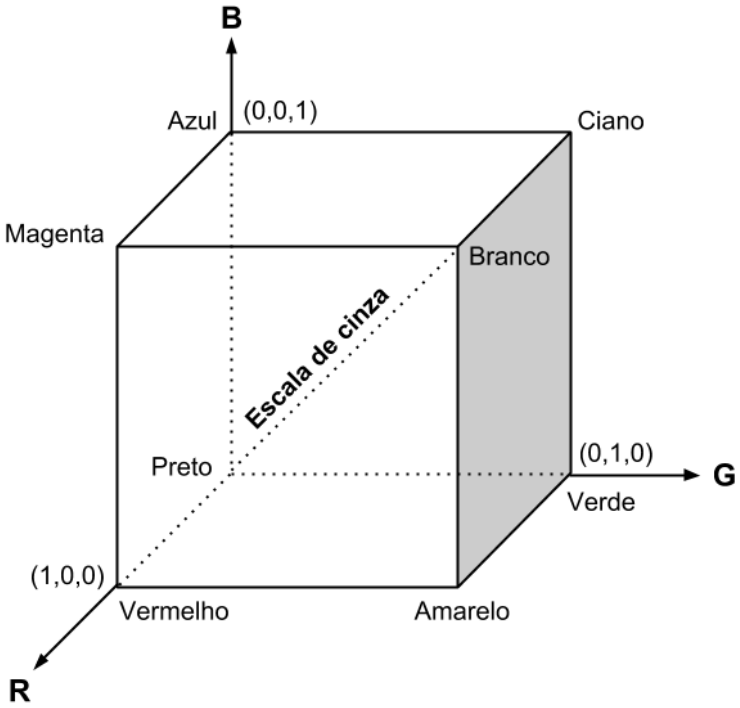

Figura 2.9: Representação do espaço de cores RGB.
Fonte: o autor.

2.6.2 HSV

O sistema de cores HSV é composto pelas componentes matiz (*Hue*), saturação (*Saturation*) e brilho (*Value*). Esses componentes configuram as coordenadas de cada pixel de uma imagem no espaço tri-dimensional de cor. Esse espaço de cor HSV é definido como um cone invertido, conforme a Figura 2.10, onde o ângulo em sua base define uma cor, a distância do centro até a sua borda define a saturação e a intensidade é definida pela profundidade do sólido geométrico [Oliveira et al., 2010].

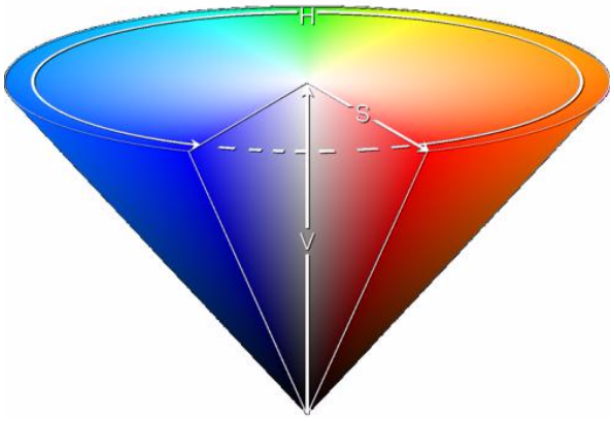

Figura 2.10: Representação do espaço de cores HSV.
Fonte: [White, 2016]

Capítulo 3

Trabalhos Relacionados

Um analisador de urina é um dispositivo utilizado no cenário clínico para realizar testes de urina de modo automatizado. Esses equipamentos podem detectar e quantificar um número de analitos incluindo bilirrubina, proteína, glicose e sangue. Muitos modelos realizam a análise a partir de leitores de tiras de urina, um tipo de fotômetro de reflectância que pode processar várias centenas de tiras por hora. A reflectância se dá na relação da luminosidade refletida pela fita e o fluxo luminoso que incide sobre ela, fornecendo as informações necessárias para obtenção dos resultados de cada analito.

Atualmente, existem no mercado diversos equipamentos que permitem a análise dos *dipsticks*. Todos tem como objetivo otimizar o fluxo de trabalho, gerenciando as análises de forma eficiente e fornecendo resultados confiáveis para detectar estágios iniciais de muitas doenças, como infecções do trato urinário, doenças renais ou diabetes. Além disso, o uso do dispositivo elimina a subjetividade da avaliação visual da tira e minimiza os riscos associados à transcrição manual dos resultados. É o caso da *Uriscan Pro II* Figura 3.1

Figura 3.1: Equipamento *Uriscan Pro II*.
Fonte: [YD Diagnostics, 2016].

O analisador *Uriscan Pro II* pode ser utilizado em locais onde a demanda é média, com capacidade de análise de 720 tiras de teste por hora. O equipamento é suportado por um leitor de código de barras e uma impressora térmica. Além da interface com código de barras, o equipamento também possui interface com o computador, mas que é lenta e não prática, segundo informações do laboratório que foi desenvolvido o presente trabalho [YD Diagnostics, 2016]. As tiras da *Uriscan* têm um sistema de aviso de ácido ascórbico, que indica concentração de vitamina C na urina e contribui para evitar falsos resultados para os itens glicose e sangue.

Outro equipamento é o *Cobas u* 411 da marca Roche Figura 3.2.

Figura 3.2: Equipamento *Cobas u* 411.
Fonte: [Cobas, 2016].

O analisador *Cobas u* 411 é descrito como a escolha certa para o gerenciamento da área de trabalho de uroanálise em locais com volume médio de testes, com capacidade de análise de 600 tiras de teste por hora. O equipamento suporta um leitor de código de barras opcional, assim como o *Uriscan Pro II*, e um terminal de sedimento, para a realização da etapa de sedimentoscopia. A análise é feita de forma seletiva, por determinação de múltiplos parâmetros definidos pelo usuário para sinalização de resultados anormais e posterior teste de acompanhamento microscópico. A detecção automática das tiras por dois sensores reduz o tempo de análise, quando comparada a leitura manual, e garante um reconhecimento confiável sobre o diagnóstico das tiras. Além disso, o *Cobas u* 411 possui porta USB e *memory stick* que permitem arquivamento conveniente de dados e fácil atualização do software do instrumento [Cobas, 2016].

Um analisador compacto, foi produzido pela Roche, o *Urisys* 1100, que foi projetado para laboratórios pequenos e escritórios médicos Figura 3.3.

Figura 3.3: Equipamento *Urisys* 1100.
Fonte: [Roche, 2016]

Assim como nos demais equipamentos, as operações são simples e as opções do software flexíveis. Apesar de realizar a análise de apenas 100 tiras de teste por hora, uma quantidade menor comparada a quantidade dos demais equipamentos, o *Urisys 1100* elimina a documentação manual através da exportação de dados por meio da transferência via *host* [Roche, 2016].

Apesar das informações obtidas, a descrição das técnicas utilizadas para avaliação dos padrões bem como as faixas de valores avaliados pela reflectância para a distinção dos padrões de cada item não foram encontradas. Além dos equipamentos, uma outra proposta para a análise das urinas foi feita por intermédio de um aplicativo para *smartphone*, que realiza a análise das fitas reagentes através das imagens obtidas pela câmera do próprio celular. A imagem obtida é comparada a um mapa codificado por cores e, em pouco tempo, relata os resultados [Medical News Today, 2016].

Capítulo 4

Materiais e Métodos

Este trabalho propõe-se a desenvolver uma metodologia para a análise de fitas reagentes, com a finalidade de dispensar equipamentos sofisticados e transformar o trabalho manual nos laboratórios que não possuem condições de ter a análise automatizada das fitas.

Por não existir uma base de imagens de fitas reagentes, foi necessário primeiramente montar uma base de imagens, adquirida via *scanner*, e desenvolver e avaliar o desempenho das técnicas propostas para a análise de fitas reagentes.

4.1 Materiais

Para a aquisição das imagens, conforme a proposta do presente trabalho, utilizou-se o scanner *HP Scanjet* 3800 *Photo*. Com o intuito de facilitar a detecção da localização da fita, foi desenvolvida uma grade de alumínio de tamanho 30 x 22 cm contendo espaço para o posicionamento de 10 fitas. Cada um desses espaços possui tamanho $0,55$ x $12,55$ cm e espaçamento de 2 cm entre as mesmas.

Figura 4.1: Grade posicionada no scanner.
Fonte: o autor.

Conforme a Figura 4.1, colocou-se um adesivo no canto superior direito da grade para que fosse sempre utilizado o mesmo lado, e dois papéis adesivos para facilitar a remoção da grade do *scanner*.

As 400 urinas utilizadas para os testes foram as que estavam conservadas em geladeira no laboratório. As mesmas ficam armazenadas por uma semana e descartadas após esse período.

O primeiro passo para que as análises pudessem ser feitas foi realizar as alíquotas das urinas que passariam pelo processo. Para tal, transferiu-se uma quantidade pequena para tubos de ensaio de fundo cônico, conforme Figura 4.2.

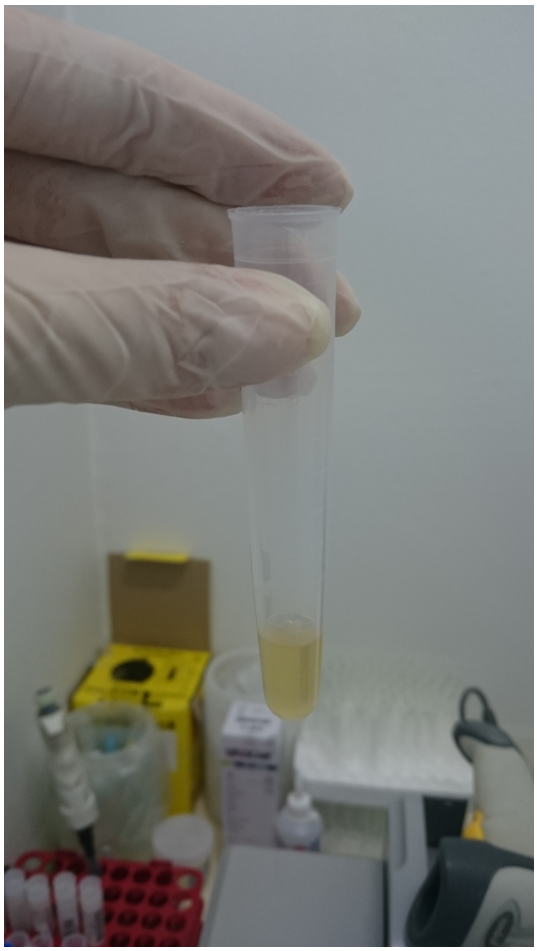

Figura 4.2: Alíquota de urina.
Fonte: o autor.

Após as urinas serem alíquotadas e registradas na máquina *Uriscan Pro II* ™, para cada urina uma fita foi submergida e posicionada no equipamento. O segundo passo foi mergulhar novas fitas nas mesmas urinas, e mais importante, na mesma ordem do processo anterior, e posicioná-las no *scanner*. No momento da aquisição da imagem, a resolução selecionada foi de 300 *dpi* e o formato png. As configurações de brilho e contraste foram mantidas as de *default* da tela de digitalização.

Apesar da grade desenvolvida ter espaços para dez fitas reagentes, foram feitas imagens de 5 em 5 *dipsticks*, conforme a Figura 4.3, para que o tempo das primeiras fitas posicionadas não extrapolassem o tempo ideal de leitura, que fica entre um e dois minutos, segundo a bula da fita. Além disso, para que todas as fitas estivessem no seu tempo ótimo de análise, aguardou-se um minuto após o posicionamento da última fita no *scanner*. Esse tempo foi utilizado pois permitiu que a última fita pudesse atingir seu tempo ideal de leitura e fez com que a primeira fita atingisse um tempo médio de um minuto e meio, o qual não excede o tempo ótimo para análise.

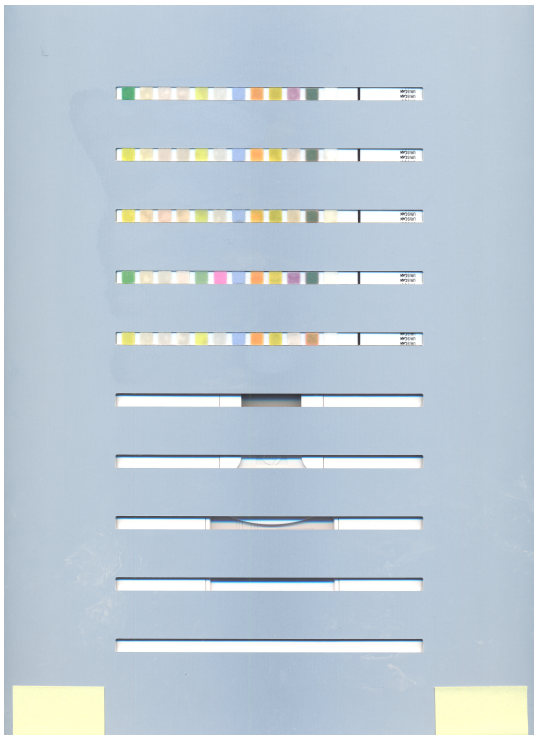

Figura 4.3: Exemplo de imagem adquirida.
Fonte: o autor.

4.2 Desenvolvimento

A implementação do algoritmo para a detecção das fitas e das suas respectivas áreas reagentes foi feita utilizando-se a biblioteca OpenCV e da linguagem *Python*.

- Python é uma linguagem de programação de alto nível, interpretada, orientada a objetos e escrita com o intuito de ser de fácil aprendizado. A mesma contém diversos módulos convenientes, o que faz dela uma opção que oferece alta produtividade tratando-se do desenvolvimento de aplicações. Isso se deve ao fato da priorização da legibilidade do código sobre a velocidade, ou seja, a enfatização da importância do esforço do programador sobre o esforço computacional [Van Rossum et al., 2007].

- A biblioteca multiplataforma OpenCV (*Open Source Computer Vision Library*) foi desenvolvida pela Intel e contém mais de 500 algoritmos. Sua criação foi motivada pelo objetivo de tornar a visão computacional acessível a usuários e programadores de áreas como robótica e interação humano-computador em tempo real [Marengoni and Stringhini, 2009].

O primeiro passo do algoritmo realiza a segmentação das fitas e de seus respectivos itens para análise. Conforme discutido anteriormente, a grade desenvolvida para o posicionamento das fitas teve como objetivo facilitar a detecção das coordenadas das mesmas.

As coordenadas X e Y, inferior e superior, de cada fita da grade foram obtidas manualmente. Esses valores são representados na Tabela 4.1, onde X_1 e X_2 são as coordenadas inicial e final de X, que são as mesmas para todas as fitas, e Y_1 e Y_2 as coordenadas inicial e final de Y.

Fita reagente	X_1	Y_1	X_2	Y_2
1	570	403	1519	453
2	570	699	1519	749
3	570	993	1519	1043
4	570	1287	1519	1337
5	570	1583	1519	1633
6	570	1877	1519	1927
7	570	2173	1519	2223
8	570	2469	1519	2519
9	570	2763	1519	2813
10	570	3057	1519	3107

Tabela 4.1: Coordenadas X e Y inicial e final de cada fita reagente.

Cada fita segmentada foi salva no diretório apropriado, e o segundo passo foi percorrer esse diretório segmentando as áreas reagentes de cada uma das fitas. Do mesmo modo como descrito anteriormente, as coordenadas de cada área foi obtida manualmente e seus valores utilizados no algoritmo. Na Tabela 4.2 seguem as coordenadas X e Y iniciais e finais de cada área reagente da fita. No caso da segmentação das áreas, as coordenadas inicial e final de Y, (Y_1 e Y_2) é a mesma para todas e X_1 e X_2 representam as coordenas inicial e final de X.

Área reagente	X_1	Y_1	X_2	Y_2
Ascórbico	891	0	941	50
Bilirrubina	94	0	144	50
Sangue	6	0	56	50
Cetonas	271	0	321	50
Densidade	716	0	766	50
Glicose	539	0	589	50
Leucócitos	805	0	855	50
Nitrito	447	0	497	50
pH	626	0	676	50
Proteínas	360	0	410	50
Urobilinogênio	182	0	232	50

Tabela 4.2: Coordenadas X e Y inicial e final de cada área reagente da fita.

As áreas reagentes foram segmentadas de modo a obter-se uma imagem de 50x50 *pixels* para montagem da base de dados e posterior extração de informações. As mesmas foram armazenadas em pastas denominadas $Strip(i)$, onde i varia de 1 a 10, que corresponde a cada uma das fitas digitalizadas.

4.3 Montagem das Bases de Dados

As bases de dados foram formadas a partir das imagens segmentadas conforme descrito anteriormente.

4.3.1 Base *Patterns_1*

Para a montagem da primeira base de dados, cada padrão ou área reagente, de cada fita, obtido pelas imagens escaneadas e segmentadas pelo algoritmo, foram salvas na base com os valores impressos pela máquina. Logo, as imagens foram nomeadas de acordo com o padrão que a mesma representava. Essas imagens foram organizadas em onze pastas, onde cada pasta representa um item analisado pela fita. Em cada uma delas encontram-se 300 imagens de tamanho 50X50 *pixels*.

As Tabelas A.1 até A.11 no Apêndice A apresentam a quantidade de imagens referentes a cada padrão de cada analito. Ao analisá-las é possível perceber que itens com padrão negativo ou normal são os que possuem maior quantidade de imagens. Isso deve-se ao fato que para esses itens uma urina saudável apresenta resposta negativa ou normal.

No caso do pH a maior concentração de imagens deu-se entre os valores 5 e 6, que, conforme descrito anteriormente, é o valor normal para o pH da urina. Para os padrões relativos a densidade, observou-se urinas com densidades mais elevadas, concentradas entre os valores 1020 e 1030.

4.3.2 Base *Patterns_2*

Para a coleta da primeira base de dados, as fitas reagentes da máquina e do *scanner* foram molhadas em tempos diferentes. Essa metodologia pode acarretar alterações em alguns valores dos analitos, tendo em vista que a urina possui uma variação biológica própria. Tal fato pode explicar o motivo dos rótulos das imagens, salvas pelo *scanner*, não condizerem com os mesmos valores obtidos pela máquina.

Por isso, para a validação da base *Patterns_1*, optou-se por realizar uma nova coleta, em que as duas fitas reagentes foram molhadas ao mesmo tempo, evitando uma possível variação biológica. Esta nova base é composta por 100 imagens, e a ideia é que, se a urina realmente sofreu variação, esta nova base terá um percentual proporcional melhor de classificação.

4.3.3 Extração de informações

A extração de informações das imagens organizadas nas bases de dados *Patterns_1* e *Patterns_2* seguiu os respectivos passos:

- Conversão da imagem para o modelo de cores HSV;
- Para cada analito buscou-se os valores máximo e mínimo de cada canal do modelo HSV;
- Para cada imagem foi obtida a média dos valores dos canais H, S e V;
- As médias dos canais de cada imagem foram normalizadas linearmente com o propósito de ajustar as escalas de valores dos atributos para o mesmo intervalo [0,1]. A normalização deu-se através dos valores máximos e mínimos obtidos e conforme a equação 4.1:

$$f(x) = \frac{X - Min}{Max - Min} \tag{4.1}$$

Além da conversão para o modelo HSV, também foram feitos testes com o modelo RGB, utilizando-se a mesma metodologia para extração de características descrita acima. Com o intuito de se ter mais informações na utilização do modelo HSV, optou-se por utilizar mais seis valores como atributos, além das médias normalizadas. Esses valores foram obtidos a partir da normalização linear do máximo e mínimo dos canais H, S e V de cada imagem.

Os valores adquiridos foram organizados no arquivo ARFF para que os mesmos fossem utilizados pela ferramenta *Weka* no processo dos testes. Como as imagens foram nomeadas de acordo com os resultados obtidos através da máquina, o atributo classe incluído no arquivo ARFF foi apenas o nome da própria imagem.

4.3.4 Testes no *Weka*

A ferramenta *Weka* provê algumas opções para testes. Uma das opções propõe usar o conjunto de dados de treino para teste, o que não é uma boa alternativa. O problema está no fato de que aprender sobre um conjunto de dados e utilizar a função de predição, obtida no aprendizado, para testá-la nos mesmos dados é um erro metodológico. Isso porque um modelo utilizado para prever classes de dados que ele já viu teria um resultado perfeito, mas falharia na previsão para os dados ainda não vistos. Essa situação é o que chama-se de *overfitting*, já discutida no presente trabalho.

Para evitar tal erro, é comum utilizar um outro conjunto de dados como o conjunto de teste. Porém, quando o algoritmo permite a manipulação de hiperparâmetros, como o parâmetro C no algoritmo SVM, ainda há o risco de se chegar na situação de *overfitting*. Isso pode acontecer porque esses parâmetros poderão ser modificados até que a performance do algoritmo seja ótima. Para resolver esse problema, o *Weka* também dá a opção de particionar o conjunto de dados em treinamento e teste. Mas, a partição dos dados reduz o número de exemplos que poderiam ser utilizados no aprendizado.

Logo, a solução utilizada é o procedimento de *cross-validation* denominado *k-fold*. Nesse procedimento o conjunto de dados é dividido em k conjuntos menores (*folds*) e para cada um desses k conjuntos um modelo é treinado utilizando os outros *k-1* conjuntos como base de treinamento. No fim do processo, a taxa de acerto obtida pelo classificador é dada pela média dos valores provenientes de cada modelo obtido. No presente trabalho, optou-se por utilizar 10 *folds*.

Selecionada a opção de como os testes seriam executados, o próximo passo foi avaliar os parâmetros dos classificadores a serem manipulados. Conforme descrito anteriormente, os algoritmos selecionados para a realização dos testes foram o *k-NN*, *MLP* e o *SVM*.

No caso do classificador *k-NN*, os testes foram realizados apenas com a variação do parâmetro k, que representa o número de vizinhos a serem utilizados. Esse parâmetro foi testado com os seguintes valores: $k \in \{1; 3; 5; 7\}$. O cálculo do grau de similaridade foi feito utilizando-se a distância euclidiana, função *default* no *Weka*.

Para o classificador *MLP*, foram manipulados os seguintes parâmetros: taxa de aprendizado, momento, quantidade de camadas escondidas e de neurônios nas camadas e os ciclos de treinamento. A primeira etapa foi variar o número de neurônios nas camadas escondidas. O padrão trazido pelo *Weka* é de uma camada escondida representado pela letra **a**, que significa o chão da média aritmética entre a quantidade de atributos e a quantidade de classes. A quantidade de atributos representa o número de neurônios da

camada de entrada e o número de classes a quantidade de neurônios presentes na camada de saída.

Para as *MLPs* do presente trabalho, utilizou-se o valor **a** padrão e outros dois valores que definem a rede com duas e três camadas escondidas. Para a rede com duas camadas escondidas, a definição da quantidade de neurônios de cada camada deu-se através da combinação da quantidade de atributos e da quantidade de classes. Enquanto que no caso da rede de três camadas escondidas, a quantidade de neurônios foi definida a partir da combinação dos valores descritos anteriormente mais a soma da quantidade de atributos com a quantidade de classes. Com o intuito de facilitar na descrição dos resultados, as abordagens descritas foram denominadas de **2 Camadas** e **3 Camadas**.

Como cada item avaliado da fita reagente possui número de classes diferentes, na Tabela 4.3 seguem os valores utilizados para o número de neurônios de cada camada.

	Atributos	Classes	Atributos + Classes	a
Ascórbico	3	4	7	3
Bilirrubina	3	3	6	3
Sangue	3	5	8	4
Cetonas	3	5	8	4
Densidade	3	6	9	4
Glicose	3	6	9	4
Leucócitos	3	5	8	4
Nitrito	3	2	5	2
pH	3	8	11	5
Proteínas	3	6	9	4
Urobilinogênio	3	4	7	3

Tabela 4.3: Quantidade de neurônios de cada camada.

Utilizando-se os melhores percentuais obtidos através da variação do número de camadas e suas respectivas quantidades de neurônios, o passo seguinte foi variar a taxa de aprendizado. Com a obtenção dos melhores resultados, as taxas de aprendizado foram mantidas e o parâmetro momento passou a ser manipulado. E por fim, o último parâmetro a ser avaliado foi a quantidade de ciclos de treinamento. Os testes foram realizados com os seguintes valores:

Taxa de aprendizado $\in \{0,1; 0,3; 0,5; 0,7; 0,9; 1,0\}$

Momento $\in \{0,1; 0,2; 0,3; 0,5; 0,7; 0,9\}$

Ciclos $\in \{250; 500; 1000; 2000\}$

Para os testes com o algoritmo *SVM*, optou-se por utilizar os três tipos de *kernel* descritos anteriormente: linear, RBF e o polinomial. Para cada um dos *kernels*, variou-se seus parâmetros específicos como gama, C e o grau do polinômio com os valores descritos abaixo:

Gama $\in \{0,0; 0,1; 1,0; 10,0; 100,0; 1000,0\}$

C $\in \{1,0; 10,0; 100,0; 1000,0; 10000,0\}$

Grau do polinômio $\in \{2; 3; 4; 5\}$

Para os *kernels* RBF e linear o primeiro passo foi variar o parâmetro gama. Visando obter a melhor combinação de parâmetros que expressasse da melhor maneira a divisão do problema, optou-se por realizar um refinamento desse parâmetro. Tal refinamento deu-se através de uma busca aleatória de valores próximos do conjunto pré-definido apresentado

acima. A partir dos valores de gama refinados é que partiu-se então para a variação do parâmetro C.

O único diferencial para o *kernel* polinomial é que o primeiro parâmetro a ser variado foi o grau do polinômio, para então seguir com os testes na mesma metodologia descrita para os demais parâmetros.

Capítulo 5

Resultados e Discussão

Os resultados apresentados a seguir foram obtidos através da conversão das imagens da base *Patterns_1* para o modelo de cores HSV e dos atributos sendo representados pelas médias normalizadas de cada canal.

Conforme apresentado na metodologia, optou-se por utilizar também o modelo de cores RGB. No caso do modelo HSV, foram utilizados os valores de máximo e mínimo de cada canal da imagem como atributo, além das médias normalizados. Porém, para os testes realizados com essas duas outras abordagens descritas, não foram obtidos resultados significativos comparados com os obtidos a seguir e, por isso, não serão apresentados nesta seção.

5.1 *k - Nearest Neighbors*

No caso do classificador *k-NN*, variou-se apenas o parâmetro para o número de vizinhos mais próximos. Os resultados obtidos são apresentados na Tabela 5.1.

k	1	3	5	7
Ascórbico	$91,33\%$	**92%**	$91,33\%$	$91,33\%$
Bilirrubina	**99, 67%**	99%	99%	99%
Cetonas	$96,33\%$	**97%**	95%	$95,33\%$
Densidade	$69,33\%$	$71,67\%$	$71,33\%$	**73, 67%**
Glicose	**100%**	$99,33\%$	$99,33\%$	$96,33\%$
Leucócitos	$89,33\%$	89%	**89, 67%**	89%
Nitrito	**99, 33%**	$98,67\%$	**99, 33%**	**99, 33%**
pH	$63,67\%$	**69, 33%**	69%	$68,67\%$
Proteínas	**86, 33%**	85%	86%	$85,67\%$
Sangue	**95%**	$93,33\%$	$93,33\%$	92%
Urobilinogênio	**98, 67%**	$97,33\%$	$97,67\%$	$96,33\%$

Tabela 5.1: Taxa de acerto para os padrões com variação do número de vizinhos.

A partir da análise dos dados obtidos na Tabela 5.1, é possível perceber que, na maioria dos casos, o maior percentual obtido foi através do valor $k = 1$. Para tal valor de k os itens bilirrubina, sangue, glicose, proteínas e urobilinogênio obtiveram as melhores taxas de acerto, atingindo $99,67\%$, 95%, 100%, $86,33\%$ e $98,67\%$, respectivamente. Para os itens ascórbico, cetonas e pH as melhores taxas de acerto deram-se utilizando $k = 3$, atingindo 92%, 97% e $69,33\%$ respectivamente.

No caso dos leucócitos, $k = 5$ foi o valor que proveu o melhor percentual, $89,67\%$, enquanto que para a densidade, a melhor taxa, $73,67\%$, foi obtida através de $k = 7$. Já o analito nitrito apresentou o maior percentual para os valores de $k = 1$, 5 e 7, atingindo $99,33\%$.

5.2 *Multilayer Perceptron*

O primeiro passo dos testes foi avaliar as taxas de acerto das abordagens para o número de camadas escondidas e a quantidade de neurônios em cada camada.

Conforme descrito na metodologia, utilizou-se o valor **a** e as abordagens propostas: **2 Camadas** e **3 Camadas** escondidas. Para todos eles mantiveram-se valores *default* dos parâmetros do *Weka*. O valor *default* para a taxa de aprendizado, momento e ciclos de treinamento são $(0,3)$, $(0,2)$ e (500) respectivamente. Os resultados obtidos são apresentados na Tabela 5.2.

	a	**2 Camadas**	**3 Camadas**
Ascórbico	$88,33\%$	**90, 33%**	$88,67\%$
Bilirrubina	**99%**	**99%**	$97,67\%$
Cetonas	**95, 33%**	92%	92%
Densidade	**72%**	$70,33\%$	$67,33\%$
Glicose	**98, 67%**	96%	$94,67\%$
Leucócitos	$87,33\%$	$87,33\%$	**87, 67%**
Nitrito	99%	99%	**99, 33%**
pH	**72, 33%**	$69,33\%$	$67,67\%$
Proteínas	86%	**86, 33%**	$85,67\%$
Sangue	$94,33\%$	$92,67\%$	**95, 33%**
Urobilinogênio	**98, 33%**	98%	94%

Tabela 5.2: Taxas de acerto para os padrões com variação do número de camadas.

A partir da avaliação das abordagens para a quantidade de camadas e neurônios, pode-se observar que todas as propostas obtiveram pelo menos um item com a maior taxa de acerto. A utilização do valor **a** *default* do *Weka*, que representa o chão da média aritmética entre a quantidade de atributos e a quantidade de classes, foi o que obteve o maior número de itens com os maiores percentuais, que são eles: cetonas $(95,33\%)$, densidade (72%), glicose $(98,67\%)$, pH $(72,33\%)$, urobilinogênio $(98,33\%)$ e a bilirrubina (99%), que também apresentou o mesmo percentual para a abordagem de **2 Camadas**.

Os itens sangue, leucócitos e nitrito obtiveram maiores percentuais com a abordagem de **3 Camadas**, obtendo $95,33\%$, $87,67\%$ e $99,33\%$ respectivamente. Enquanto que para a abordagem de **2 Camadas** os itens ascórbico e proteínas obtiveram $90,33\%$ e $86,33\%$.

Utilizando-se o valor **a** para os itens bilirrubina, cetonas, densidade, glicose, pH e urobilinogênio, a abordagem de **2 Camadas** para as proteínas e a abordagem de **3 Camadas** para os leucócitos, nitrito e sangue, o próximo passo foi manipular a taxa de aprendizado. A Tabela 5.3 apresenta os resultados desses testes. Os percentuais que representam o valor $0,3$ para a taxa de aprendizado são relativos as melhores taxas obtidas na Tabela 5.2.

Taxa de aprendizado	0,1	0,3	0,5	0,7	0,9	1,0
Ascórbico	87,67%	**90,33%**	89%	**90,33%**	88,33%	89%
Bilirrubina	**99%**	**99%**	**99%**	**99%**	**99%**	**99%**
Cetonas	94,33%	**95,33%**	95%	95%	95%	**95,33%**
Densidade	**74%**	72%	72%	71,67%	70,67%	71,33%
Glicose	95,67%	98,67%	99,33%	**99,67%**	99,33%	**99,67%**
Leucócitos	83%	87,67%	**88,67%**	87,33%	86%	**88,67%**
Nitrito	92%	**99,33%**	99%	99%	99%	99%
pH	31,33%	**72,33%**	64,67%	62,67%	63%	56,33%
Proteínas	84,67%	**86,33%**	84,67%	85,33%	85,33%	83,67%
Sangue	84,33%	**95,33%**	93%	92,67%	93,33%	93%
Urobilinogênio	98%	**98,33%**	**98,33%**	98%	**98,33%**	**98,33%**

Tabela 5.3: Taxas de acerto para variação da taxa de aprendizado.

Com a variação da taxa de aprendizado, é possível observar que em alguns casos as taxas de acerto obtiveram uma pequena melhora, quando em comparação ao valor padrão $(0,3)$. No caso da densidade a melhora foi de 2%, atingindo 74% com valor de aprendizado $0,1$. A glicose e os leucócitos apresentaram melhoria de 1%, atingindo $99,67\%$ e $88,67\%$ respectivamente. Esses percentuais foram alcançados com os valores de taxa $0,7$ e $1,0$ para a glicose e $0,5$ e $1,0$ para os leucócitos.

Para o sangue $(95,33\%)$, nitrito $(99,33\%)$, pH $(72,33\%)$ e proteínas $(86,33\%)$, as melhores taxas de acerto foram obtidas através do valor *default* de aprendizado $(0,3)$. O analito ascórbico apresentou $90,33\%$ para aprendizado igual a $(0,3)$ e $(0,7)$, enquanto que para as cetonas, com $95,33\%$, as taxas de aprendizado foram $0,3$ e $1,0$. O urobilinogênio apresentou a mesma taxa de acerto para quase todos os valores de aprendizado, $98,33\%$ para $(0,3)$, $(0,5)$, $(0,9)$ e $(1,0)$. Já no caso da bilirrubina (99%), o percentual foi o mesmo para todas as variações da taxa de aprendizado.

Tendo em vista que em alguns casos o percentual foi o mesmo para mais de uma taxa de aprendizado, a seguir seguem os itens com as taxas utilizadas para os próximos passos dos testes.

Ascórbico $= 0,3$

Bilirrubina $= 0,3$

Cetonas $= 0,3$

Densidade $= 0,1$

Glicose $= 0,7$

Leucócitos $= 0,5$

Nitrito $= 0,3$

pH $= 0,3$

Proteínas $= 0,3$

Sangue $= 0,3$

Urobilinogênio $= 0,3$

A partir das taxas de aprendizado definidas para cada analito, o próximo passo foi a variação do parâmetro momento. A Tabela 5.4 apresenta os resultados dessa etapa de teste. Os percentuais apresentados para momento igual a $0,2$ são relativos as melhores taxas obtidas através dos valores de aprendizado definidos na descrição acima.

Momento	0,1	0,2	0,3	0,5	0,7	0,9
Ascórbico	88,33%	**90,33%**	89,67%	89,67%	89,33%	87,33%
Bilirrubina	99%	99%	99%	99%	99%	**99,33%**
Cetonas	**95,33%**	**95,33%**	**95,33%**	95%	95%	**95,33%**
Densidade	73,33%	**74%**	**74%**	**74%**	73%	72%
Glicose	99,67%	99,67%	99,33%	99,67%	**100%**	**100%**
Leucócitos	**89%**	88,67%	88,67%	88,33%	87,33%	84,33%
Nitrito	95,67%	**99,33%**	**99,33%**	99%	99%	99%
pH	**73,33%**	72,33%	72,67%	**73,33%**	**73,33%**	64,67%
Proteínas	**86,33%**	**86,33%**	85,33%	84,33%	85%	84,67%
Sangue	93,33%	**95,33%**	93,33%	94%	92,33%	89%
Urobilinogênio	**98,33%**	**98,33%**	**98,33%**	**98,33%**	98%	97,67%

Tabela 5.4: Taxas de acerto para variação do parâmetro momento.

Com a manipulação do parâmetro momento, é possível observar que em apenas quatro casos as taxas de acerto obtiveram uma pequena melhora, quando em comparação ao valor padrão $(0,2)$. É o caso dos padrões bilirrubina, glicose, leucócitos e pH. Os itens bilirrubina, glicose e leucócitos obtiveram uma melhora de $0,33\%$, atingindo $99,33\%$, 100% e 89% respectivamente. Para a bilirrubina o valor do momento foi de $(0,9)$, para a glicose o percentual foi atingido com os parâmetros $0,7$ e $0,9$ e para os leucócitos o momento foi de $0,1$. Para o pH, o momento igual a $(0,1)$, $(0,5)$ e $(0,7)$ proporcionou uma melhora de 1%, atingindo uma taxa de acerto de $73,33\%$.

Nos demais casos, não houve melhoras nos percentuais resultantes quando comparado ao percentual obtido quando o parâmetro momento não foi manipulado. É o caso do ascórbico $(90,33\%)$ e sangue $(95,33\%)$. Apesar disso, algumas variações do valor do parâmetro obteve a mesma taxa de acerto obtida pelo valor *default* $(0,2)$. É o caso dos itens cetonas $(95,33\%)$, densidade (74%), nitrito $(99,33\%)$, proteínas $(86,33\%)$ e urobilinogênio $(98,33\%)$. Além do valor padrão $(0,2)$, os percentuais foram atingidos com o momento igual a $(0,1)$, $(0,3)$, e $(0,9)$ no caso das cetonas, para a densidade os valores foram $(0,3)$ e $(0,5)$, para o nitrito $(0,3)$, para as proteínas $0,1$ e para o urobilinogênio $(0,1)$, $(0,3)$ e $(0,5)$.

Tendo em vista que em alguns casos o percentual obtido foi o mesmo para mais de um valor de momento, a seguir seguem os itens com os valores utilizados para os próximos passos dos testes.

Ascórbico $= 0,2$

Bilirrubina $= 0,9$

Cetonas $= 0,2$

Densidade $= 0,2$

Glicose $= 0,7$

Leucócitos $= 0,1$

Nitrito $= 0,2$

pH $= 0,1$

Proteínas $= 0,2$

Sangue $= 0,2$

Urobilinogênio $= 0,2$

O último passo dos testes com o algoritmo *MLP* foi variar o parâmetro *epoch* ou ciclos de treinamento. Foram utilizados a quantidade de camadas, neurônios, taxa de aprendizado e o momento, definidos nos passos anteriores para cada um dos analitos. Os resultados obtidos seguem na Tabela 5.5. Os percentuais apresentados anteriormente nos testes são referentes a 500 ciclos de treinamento.

Ciclos	250	500	1000	2000
Ascórbico	88,33%	**90,33%**	89,67%	88,67
Bilirrubina	99%	99,33%	**99,67%**	**99,67%**
Cetonas	95%	**95,33%**	94,67%	**95,33%**
Densidade	72,33%	74%	**75%**	73,67%
Glicose	**100%**	**100%**	**100%**	**100%**
Leucócitos	**89,33%**	89%	88%	88%
Nitrito	92%	**99,33%**	99%	99%
pH	71%	73,33%	**74%**	73,33%
Proteínas	85%	**86,33%**	86%	84,67%
Sangue	90,33%	**95,33%**	93,33%	94%
Urobilinogênio	98%	98,33%	**98,67%**	98%

Tabela 5.5: Taxas de acerto para variação do parâmetro ciclos de treinamento.

A partir dos percentuais obtidos, é possível perceber ganhos sutis para alguns dos itens analisados, quando comparados as taxa obtidas pelo valor padrão (500). É o caso da bilirrubina, densidade, leucócitos, pH e do urobilinogênio, atingindo 99,67%, 75%, 89,33%, 74% e 98,67%, respectivamente. Nos casos da densidade, pH e do urobilinogênio essas taxas foram atingidas com o valor de ciclos igual a 1000. Para a bilirrubina, a melhor taxa foi obtida através do valor 1000 e 2000, enquanto que para os leucócitos foi o valor 250.

Para os demais casos não houve ganho nos percentuais comparado com os percentuais obtidos quando o parâmetro ciclos de treinamento não foi variado.

A Tabela 5.6 a seguir apresenta as melhores configurações da rede neural para cada analito.

	# Camadas	Taxa de Aprendizado	Momento	Ciclos	Percentual
Ascórbico	**2 Camadas**	0,3	0,2	500	90,33%
Bilirrubina	**a**	0,3	0,9	1000	99,67%
Cetonas	**a**	0,3	0,2	500	95,33%
Densidade	**a**	0,1	0,2	1000	75%
Glicose	**a**	0,7	0,7	500	100%
Leucócitos	**3 Camadas**	0,5	0,1	250	89,33%
Nitrito	**3 Camadas**	0,3	0,2	500	99,33%
pH	**a**	0,3	0,1	1000	74%
Proteínas	**2 Camadas**	0,3	0,2	500	86,33%
Sangue	**3 Camadas**	0,3	0,2	500	95,33%
Urobilinogênio	**a**	0,3	0,2	1000	98,67%

Tabela 5.6: Parâmetros do *MLP* que obtiveram melhores taxas de acerto para cada analito.

A partir dos valores obtidos para cada um dos parâmetros manipulados é possível perceber que em sua maioria foram mantidos os valores padrões definidos pela ferramenta *Weka*. No caso do número de camadas e de neurônios a maioria dos analitos obtiveram melhores resultados com o valor **a**, que apresenta apenas uma camada e com o número de neurônios igual ao chão da média aritmética entre a quantidade de atributos e a quantidade de classes. No caso da taxa de aprendizado, oito dos onze itens apresentaram melhores percentuais com o valor 0,3.

Apesar de algumas variações, o momento também mostrou-se melhor com o valor *default* $(0,2)$. Para o parâmetro ciclos de aprendizado seis dos onze analitos atingiram os melhores percentuais por meio do valor padrão (500), quatro dos itens obtiveram os melhores percentuais com ciclos igual a 1000 e apenas um com o valor 1000.

5.3 *SVM*

5.3.1 *Kernel*

O primeiro passo dos testes foi avaliar as taxas de acerto de cada um dos *kernels* selecionados, que foram: linear, RBF e polinomial. Para todos eles mantiveram-se valores *default* dos parâmetros do *Weka*. O valor *default* para o parâmetro gama e o parâmetro C são $0,0$ e $1,0$ respectivamente. No caso do *kernel* poloninomial, o grau padrão é 3.

Na Tabela 5.7 é possível observar os resultados obtidos com a execução do classificador sem otimização de seus parâmetros.

Kernel	RBF	Linear	Polinomial
Ascórbico	$75,67\%$	**81**%	$70,33\%$
Bilirrubina	**97,67**%	**97,67**%	**97,67**%
Cetonas	$93,33\%$	**94**%	$91,33\%$
Densidade	$64,67\%$	**69,33**%	$42,33\%$
Glicose	**95,67**%	**95,67**%	$93,67\%$
Leucócitos	83%	**84**%	$80,67\%$
Nitrito	**99,33**%	**99,33**%	98%
pH	$44,33\%$	**46**%	$32,67\%$
Proteínas	81%	**82**%	$74,67\%$
Sangue	$85,33\%$	**88**%	$80,67\%$
Urobilinogênio	**94**%	**94**%	**94**%

Tabela 5.7: Taxas de acerto para os padrões com variação de *kernel*.

Alguns dos itens analisados obtiveram uma taxa de acerto relativamente boa, levando-se em consideração que não houve otimização dos parâmetros. É o exemplo da bilirrubina, que apresentou taxa de $97,67\%$ para todos os *kernels*, ou o analito nitrito que apresentou um percentual de $99,33\%$ para os *kernels* RBF e linear.

A seguir, cada *kernel* terá seus parâmetros específicos otimizados.

5.3.2 *Kernel* - RBF

Utilizando-se o *kernel* RBF, o primeiro parâmetro a ser manipulado foi o gama. Na Tabela 5.8 seguem os resutados obtidos. Os percentuais obtidos para o valor de gama $0,0$ são relativos as taxas de acerto atingidas sem a manipulação desse parâmetro.

Gama	0,0	0,1	1,0	10,0	100,0	1000,0
Ascórbico	75,67%	72,33%	83,67%	**94,33%**	92,33%	82%
Bilirrubina	97,67%	97,67%	97,67%	99%	**99,33%**	99%
Cetonas	93,33%	91,33%	94,33%	95,67%	**96,67%**	93,67%
Densidade	64,67%	35,67%	72,33%	**74%**	**74%**	**74%**
Glicose	95,67%	93,67%	95,67%	99%	**99,33%**	97%
Leucócitos	83%	83%	86,33%	89%	**89,67%**	78,67%
Nitrito	**99,33%**	**99,33%**	**99,33%**	**99,33%**	97%	93%
pH	44,33%	37%	48,33%	66,33%	**71,33%**	61,67%
Proteínas	81%	74,67%	82,33%	**87%**	86,67%	77,67%
Sangue	85,33%	80,33%	88,67%	91,33%	**92%**	77,67%
Urobilinogênio	94%	94%	94,67%	96,67%	**98,33%**	96%

Tabela 5.8: Taxas de acerto para os padrões com *kernel* RBF e variação do valor gama.

A partir dos percentuais obtidos, é possível observar que as taxas de acerto obtiveram uma melhora quando comparadas as taxas resultantes para o valor padrão de gama $(0,0)$. Apenas o item nitrito manteve o mesmo percentual de $99,33\%$ com ou sem o refinamento do parâmetro.

Dentre os onze analitos, sete deles obtiveram melhor resultado com gama igual a $100,0$, sendo eles: bilirrubina $(99,33\%)$ com melhora de $1,66\%$, sangue (92%) e leucócitos $(89,67\%)$ apresentando melhora de $6,67\%$, cetonas $(96,67\%)$ com acréscimo de $3,34\%$, glicose $(99,33\%)$ com $3,66\%$, urobilinogênio $(98,33\%)$ com $4,33\%$ e pH $(71,33\%)$ com o maior percentual de melhora, 27%.

No caso do padrão ascórbico e proteínas, gama igual a $10,0$ foi o que teve melhor percentual, com $94,33\%$ e 87% e obtendo $18,66\%$ e 6% de melhora, respectivamente. Para a densidade o melhor percentual (74%) foi obtido para os valores de gama igual a $(10,0)$, $(100,0)$ e $(1000,0)$, representando uma melhora de $9,33\%$.

A partir dos melhores valores de gama, optou-se por fazer um ajuste mais fino. A ideia foi realizar uma busca aleatória, com refinamentos sucessivos de valores próximos do parâmetro gama onde houve melhor taxa de acerto, com o intuito de obter-se, talvez, melhores percentuais. A seguir, a Tabela 5.9 apresenta os valores encontrados através dos refinamentos e seus respectivos percentuais.

	Gama	Taxa de acerto
Ascórbico	$25,0$	$94,67\%$
Bilirrubina	$100,0$	$99,33\%$
Cetonas	$100,0$	$96,67\%$
Densidade	$10,0$	74%
Glicose	$100,0$	$99,33\%$
Leucócitos	$100,0$	$89,67\%$
Nitrito	$1,0$	$99,33\%$
pH	$320,0$	$72,33\%$
Proteínas	$40,0$	$87,67\%$
Sangue	$127,0$	$92,67\%$
Urobilinogênio	$100,0$	$98,33\%$

Tabela 5.9: Taxas de acerto para os padrões com *kernel* RBF e melhores valores encontrados de gama.

É possível observar que para a maioria dos itens, o valor de gama manteve-se o mesmo. Mas para alguns casos foi possível obter sutis melhoras nos percentuais de acerto. É o caso do ascórbico, que antes apresentava percentual de $94,33\%$ com valor de gama igual a $10,0$ e passou a ter $94,67\%$ com gama igual a $25,0$. No caso do sangue e das proteínas, a melhora foi de $0,87\%$, para gama igual a $127,0$ e $40,0$ respectivamente. O analito pH foi o que obteve a maior melhora, com o aumento de 1% para o valor de gama igual a $320,0$.

Utilizando-se os valores de gama da Tabela 5.9 obtidos através da busca aleatória, o próximo passo foi ajustar o parâmetro C. Os resultados obtidos estão representados na Tabela 5.10. Os percentuais apresentados para o valor C igual a $1,0$ são relativos as melhores taxas obtidas através dos valores de gama da Tabela 5.9.

C	1,0	10,0	100,0	1000,0	10000,0
Ascórbico	**94, 67%**	94%	94, 33%	92, 67%	90, 67%
Bilirrubina	**99, 33%**	**99, 33%**	**99, 33%**	**99, 33%**	**99, 33%**
Cetonas	96, 67%	**97%**	96, 67%	96%	96%
Densidade	74%	74, 33%	**75, 33%**	74, 67%	73%
Glicose	**99, 33%**	**99, 33%**	**99, 33%**	**99, 33%**	**99, 33%**
Leucócitos	**89, 67%**	88, 67%	86, 67%	86, 33%	86%
Nitrito	**99, 33%**	**99, 33%**	99%	98, 67%	98, 33%
pH	**72, 33%**	69, 33%	69%	64, 33%	64, 33%
Proteínas	**87, 67%**	84, 67%	87%	87, 33%	86, 67%
Sangue	**92, 67%**	92, 33%	92, 33%	92, 33%	92, 33%
Urobilinogênio	98, 33%	**99, 33%**	**99, 33%**	**99, 33%**	**99, 33%**

Tabela 5.10: Taxas de acerto para os padrões com *kernel* RBF e variação do valor C.

A partir da análise da variação do parâmetro C, é possível perceber que para alguns dos itens a taxa de acerto melhorou, para outros caiu e para os demais não houve diferença comparado aos percentuais obtidos para o valor *default* de C $(1,0)$. Para bilirrubina, glicose e nitrito não houve diferença entre valor *default* e as demais variações de C.

No caso dos analitos ascórbico, sangue e proteínas a queda no percentual foi sutil, de apenas $0,34\%$, enquanto que para os leucócitos a queda foi de 1%. O pH foi o que teve maior queda (3%), apresentando $69,33\%$ com C igual a 10, enquanto que com C igual a $1,0\%$ a taxa de acerto foi de $72,33\%$.

Os itens que obtiveram uma pequena melhora foram as cetonas (97%), apresentando melhora de $0,33\%$ com C igual a $(10,0)$, a densidade $(75,33\%)$ com acréscimo de $1,33\%$ para C igual a $100,0$ e o urobilinogênio, com maior taxa de melhora (1%), apresentando $99,33\%$ para todos os valores testados de C.

A Tabela 5.11 a seguir apresenta as melhores configurações do *SVM* com o *kernel* RBF para cada analito.

	Gama	C	Percentual
Ascórbico	25, 0	1, 0	94, 67%
Bilirrubina	100, 0	1, 0	99, 33%
Cetonas	100, 0	10, 0	97%
Densidade	10, 0	100, 0	75, 33%
Glicose	100, 0	1, 0	99, 33%
Leucócitos	100, 0	1, 0	89, 67%
Nitrito	1, 0	1, 0	99, 33%
pH	320, 0	1, 0	72, 33%
Proteínas	40, 0	1, 0	87, 67%
Sangue	127, 0	1, 0	92, 67%
Urobilinogênio	100, 0	10, 0	99, 33%

Tabela 5.11: Parâmetros do *SVM* com o *kernel* RBF que obtiveram melhores taxas de acerto para cada analito.

A partir dos valores obtidos é possível perceber que para o parâmetro gama nenhum analito manteve o valor padrão definido pelo *Weka* $(0, 0)$, tendo em vista os refinamentos sucessivos realizado sobre essa variável. No caso do parâmetro C, oito dos onze analitos obtiveram os melhores percentuais com o valor *default* $(1, 0)$, dois deles com o valor $10, 0$ e apenas um analito com C igual a $100, 0$.

5.3.3 *Kernel* - Linear

Para o *kernel* linear, apenas o parâmetro C foi ajustado. Os valores obtidos seguem na Tabela 5.12.

C	1,0	10,0	100,0	1000,0	10000,0
Ascórbico	81%	84, 33%	86%	**86, 67%**	**86, 67%**
Bilirrubina	97, 67%	99%	**99, 33%**	**99, 33%**	**99, 33%**
Cetonas	94%	95, 67%	94, 67%	**96, 33%**	**96, 33%**
Densidade	69, 33%	75, 33%	75%	**76%**	75, 33%
Glicose	95, 67%	97, 33%	99, 33%	**100%**	**100%**
Leucócitos	84%	87%	87, 33%	88%	**88, 33%**
Nitrito	**99, 33%**	99%	99%	99%	99%
pH	46%	63%	73%	**75, 33%**	73, 67%
Proteínas	82%	87%	88, 33%	**89, 33%**	89%
Sangue	88%	89, 67%	**93%**	**93%**	92, 67%
Urobilinogênio	94%	97%	98%	**98, 67%**	**98, 67%**

Tabela 5.12: Taxas de acerto para os padrões com *kernel* linear e variação do valor C.

Com a utilização do *kernel* linear e do refinamento do parâmetro C é possível perceber que para todos os parâmetros, menos para o analito nitrito, houve uma melhora nas taxas de acerto, quando comparadas as taxas obtidas através do valor padrão de C $(1, 0)$.

O nitrito que antes apresentou taxa de $99, 33\%$ quando analisado com o valor C igual a $(1, 0)$, obteve uma pequena queda, passando a ter resultado de 99% para todos os valores analisados de C. Para os demais itens foram obtidas notáveis melhoras, como por

exemplo para o pH, que para o valor de C igual a $1000,0$ apresentou uma taxa de $75,33\%$ de acerto, o que representa uma melhora de $29,33\%$ quando comparado com o valor obtido com o *kernel* linear sem nenhum refinamento do parâmetro C. Um caso importante é o da glicose, que antes apresentava $95,67\%$ de acerto e que alcançou 100% para valores de C igual a $1000,0$ e $10000,0$. Com esses mesmos valores de C, os itens ascórbico, cetonas e urobilinogênio obtiveram melhoras. No caso do ascórbico o percentual melhorou $5,67\%$, atingindo $86,67\%$ de acerto. Para as cetonas a taxa que antes era de 94% passou a ser de $96,33\%$, e para o urobilinogênio a melhoria foi de $4,67\%$, atingindo $98,67\%$. Já os itens densidade e proteínas obtiveram melhores resultados com C igual a $1000,0$, apresentando 76% com melhora de $6,67\%$ e $89,33\%$ com aumento de $7,33\%$ respectivamente.

A bilirrubina apresentou uma pequena melhora, $1,66\%$, atingindo $99,33\%$ para C igual a $(100,0)$, $(1000,0)$ e $(10000,0)$. No caso do sangue a melhor taxa foi obtida para C igual a $(100,0)$ e $(1000,0)$, atingindo 93%, que representa uma melhora de 5% quando comparada a taxa obtida para o valor *default* de C. Para os leucócitos a melhoria foi de $4,33\%$, atingindo o percentual de $88,33\%$.

5.3.4 *Kernel* - Polinomial

No *kernel* polinomial, o primeiro parâmetro que foi ajustado foi o grau do polinômio. Como o valor *default* do grau era 3, utilizou-se os graus 2, 4 e 5 para teste, conforme descrito na metodologia dos testes. Na Tabela 5.13 estão descritas as taxas de acerto obtidas de cada item.

Grau	2	3	4	5
Ascórbico	**72,67%**	70,33%	69,33%	69,33%
Bilirrubina	**97,67%**	**97,67%**	**97,67%**	**97,67%**
Cetonas	**91,33%**	**91,33%**	87,33%	87,33%
Densidade	**45,33%**	42,33%	29,67%	28,67%
Glicose	**93,67%**	**93,67%**	89%	89%
Leucócitos	**83%**	80,67%	73,67%	73,67%
Nitrito	**99,33%**	98%	97%	94,67%
pH	**37,33%**	32,67%	32%	32%
Proteínas	**74,67%**	**74,67%**	**74,67%**	**74,67%**
Sangue	80,33%	**80,67%**	72%	70%
Urobilinogênio	**94%**	**94%**	**94%**	**94%**

Tabela 5.13: Taxas de acerto para os padrões com *kernel* polinomial e variação de grau.

A partir da variação do grau, é possível concluir que para a maioria dos casos, menos para o sangue, o grau 2 foi o que obteve melhores resultados. No caso do sangue, o grau 3 (valor *default*) foi o que obteve o melhor percentual, atingindo $80,67\%$. No caso da bilirrubina ($97,67\%$), proteínas ($74,67\%$) e do urobilinogênio (94%) o percentual foi o mesmo para todas as variações do parâmetro grau. As cetonas e a glicose foram as que obtiveram o mesmo percentual para os valores de grau 2 e 3, alcançando $91,33\%$ e $93,67\%$ respectivamente.

Utilizando-se o valor de grau 3 para o analito sangue e de grau 2 para os demais itens, o próximo passo foi variar o parâmetro gama. Os resultados obtidos dessa etapa de teste estão representados na Tabela 5.14. Os percentuais obtidos para o valor gama igual

a $0,0$ são relativos as taxas de acerto atingidas com a manipulação apenas do parâmetro grau, descrito anteriormente.

Gama	0,0	0,1	1,0	10,0	100,0	1000,0
Ascórbico	$72,67\%$	$69,33\%$	77%	$91,33\%$	**$91,67\%$**	**$91,67\%$**
Bilirrubina	$97,67\%$	$97,67\%$	$97,67\%$	**$99,67\%$**	**$99,67\%$**	**$99,67\%$**
Cetonas	$91,33\%$	$87,33\%$	$94,33\%$	$94,33\%$	**$97,33\%$**	97%
Densidade	$45,33\%$	$28,67\%$	$69,33\%$	75%	**76%**	$75,67\%$
Glicose	$93,67\%$	89%	$95,67\%$	**100%**	**100%**	**100%**
Leucócitos	83%	$73,67\%$	$86,67\%$	**$88,67\%$**	88%	$86,33\%$
Nitrito	$99,33\%$	92%	$99,33\%$	$99,33\%$	**$99,67\%$**	$99,33\%$
pH	$37,33\%$	32%	$49,33\%$	**$74,67\%$**	$74,33\%$	$70,67\%$
Proteínas	$74,67\%$	$74,67\%$	$82,67\%$	**88%**	$87,67\%$	$75,67\%$
Sangue	$80,67\%$	70%	$88,33\%$	**$91,67\%$**	**$91,67\%$**	90%
Urobilinogênio	94%	94%	95%	$98,33\%$	**99%**	**99%**

Tabela 5.14: Taxas de acerto para os padrões com *kernel* polinomial e variação do valor gama.

A partir dos percentuais obtidos, é possível observar que as taxas de acerto obtiveram uma melhora para todos os itens, quando comparadas as taxa obtidas a partir do valor *default* $(0,0)$. Dentre os onze analitos, três deles obtiveram melhor resultado com gama igual a $100,0$: cetonas ($97,33\%$), densidade (76%) e nitrito ($99,67\%$). Para esses 3 itens as melhoras foram de 6%, $30,67\%$ e $0,34\%$ respectivamente. Para o valor de gama igual a $10,0$, os melhores resultados foram obtidos pelos leucócitos ($88,67\%$) com melhora de $5,67\%$, proteínas (88%) com ganho de $13,33\%$ e pH ($74,67\%$), obtendo a maior melhora de $37,34\%$.

No caso do analito ascórbico a melhoria foi de 19% para valores gama igual a 100 e 1000 atingindo $91,67\%$, enquanto que para a bilirrubina foi de apenas 2%, atingindo $99,67\%$ com gama igual a 10, 100 e 1000. Para o sangue o melhor percentual ($91,67\%$) foi obtido para os valores de gama igual a $(10,0)$ e $(100,0)$, representando uma melhora de 11%, enquanto que no caso do urobilinogênio (99%) o percentual de melhora foi de 5% para gama igual a $100,0$ e $1000,0$. Para o item glicose, o percentual atingido foi de 100% para gama igual a $(10,0)$, $(100,0)$ e $(1000,0)$.

Assim como no *kernel* RBF, a partir dos melhores valores de gama, optou-se por fazer um ajuste mais fino, variando-se os valores próximos do gama onde houve melhor percentual de acerto. A seguir, a Tabela 5.15 apresenta os valores encontrados e seus respectivos percentuais.

	Gama	Taxa de acerto
Ascórbico	103, 0	92%
Bilirrubina	10, 0	99, 67%
Cetonas	100, 0	97, 33%
Densidade	120, 0	76, 33%
Glicose	10, 0	100%
Leucócitos	40, 0	89%
Nitrito	100, 0	99, 67%
pH	10, 0	74, 67%
Proteínas	20, 0	88, 33%
Sangue	20, 0	92, 33%
Urobilinogênio	100, 0	99%

Tabela 5.15: Taxas de acerto para os padrões com *kernel* polinomial e melhores valores encontrados de gama.

Alguns dos valores de gama mantiveram-se os mesmos da tabela anterior, mas para alguns casos foi possível obter pequenas melhoras nos percentuais de acerto.

É possível observar que para sete dos onze itens, o valor de gama manteve-se o mesmo, mas para alguns casos foi possível obter sutis melhoras nos percentuais de acerto. Para os itens ascórbico, densidade, leucócitos e proteínas obteve-se 0, 33% de melhora com gama igual a (103, 0), (120, 0), (40, 0) e (20, 0) respectivamente. Para o sangue, o percentual de melhora foi minimamente melhor, de apenas 0, 66% com valor de gama igual a 20.

Com os valores de gama obtidos, o último passo foi variar o valor do parâmetro C. Os resultados dessa última etapa estão representados na Tabela 5.16. Os percentuais apresentados para o valor C igual a 1, 0 são relativos as melhores taxas obtidas através da combinação dos valores de grau e gama definidos nos passos anteriores desse teste.

C	1,0	10,0	100,0	1000,0	10000,0
Ascórbico	**92%**	91, 67%	91, 67%	**92%**	91%
Bilirrubina	**99, 67%**	**99, 67%**	**99, 67%**	**99, 67%**	**99, 67%**
Cetonas	**97, 33%**	97%	97%	95, 33%	91, 67%
Densidade	76, 33%	**76, 67%**	76%	**76, 67%**	72, 67%
Glicose	**100%**	**100%**	**100%**	**100%**	**100%**
Leucócitos	**89%**	88%	86, 67%	86%	88%
Nitrito	**99, 67%**	99, 33%	99, 33%	99, 33%	99, 33%
pH	74, 67%	73%	74, 33%	72%	**75%**
Proteínas	**88, 33%**	87, 67%	88%	87, 33%	85%
Sangue	**92, 33%**	91, 67%	91, 67%	91%	91, 33%
Urobilinogênio	**99%**	**99%**	**99%**	**99%**	**99%**

Tabela 5.16: Taxas de acerto para os padrões com *kernel* polinomial e variação do valor C.

Com a utilização do *kernel* polinomial, da definição do grau do polinômio, do refinamento dos parâmetros gama e C, é possível perceber que para alguns dos itens a taxa de acerto melhorou, para outros caiu e para os demais não houve diferença, quando comparadas as taxas obtidas através do valor padrão de C (1, 0). Para o item ascórbico,

bilirrubina, glicose e urobilinogênio não houve diferença entre as taxas de acerto obtidas com ou sem refinamento do parâmetro C.

No caso dos analitos cetonas e proteínas a queda no percentual foi sutil, de apenas $0,33\%$, comparada com as taxas obtidas através do valor *default* $(1,0)$. Para o nitrito a queda foi de $0,34\%$, para o sangue $0,66\%$ e para os leucócitos 1%.

Os dois itens que obtiveram uma pequena melhora foi a densidade $(76,67\%)$, apresentando melhora de $0,34\%$ com C igual a $10,0$ e $100,0$ e o pH (75%) com acréscimo de $0,33\%$ para C igual a $10000,0$.

A Tabela 5.17 a seguir apresenta as melhores configurações do *SVM* com o *kernel* polinomial para cada analito.

	Grau do polinômio	Gama	C	Percentual
Ascórbico	2	$103,0$	$1,0$	92%
Bilirrubina	2	$10,0$	$1,0$	$99,67\%$
Cetonas	2	$100,0$	$1,0$	$97,33\%$
Densidade	2	$120,0$	$10,0$	$76,67\%$
Glicose	2	$10,0$	$1,0$	100%
Leucócitos	2	$40,0$	$1,0$	89%
Nitrito	2	$100,0$	$1,0$	$99,67\%$
pH	2	$10,0$	$10000,0$	75%
Proteínas	2	$20,0$	$1,0$	$88,33\%$
Sangue	3	$20,0$	$1,0$	$92,33\%$
Urobilinogênio	2	$100,0$	$1,0$	99%

Tabela 5.17: Parâmetros do *SVM* com o *kernel* polinomial que obtiveram melhores taxas de acerto para cada analito.

A partir dos valores obtidos é possível perceber que para o parâmetro gama, assim como no *kernel* RBF, nenhum analito manteve o valor padrão definido pelo *Weka* $(0,0)$. Do mesmo modo, o grau do polinômio de apenas um dos analitos manteve o valor padrão (3) e todos os demais atingiram os melhores percentuais com grau igual a 2. No caso do parâmetro C, nove dos onze analitos obtiveram os melhores taxas de acerto com o valor *default* $(1,0)$, um com o valor $10,0$ e outro com C igual a $10000,0$.

5.4 Discussão

Além dos testes realizados acima, e conforme abordado no início dessa seção, optou-se por utilizar também o modelo de cores RGB e o modelo HSV com os valores normalizados de máximo e mínimo de cada canal como atributo, aliados as médias normalizadas. Porém, para os testes realizados com essas duas outras abordagens descritas não foram obtidos resultados significativos comparados com os apresentados anteriormente. Mesmo com o uso dessas duas propostas, não houveram mudanças em relação a separação dos dados. Isso porque as duas metodologias não forneceram mais informações que pudessem discriminar melhor as classes, obtendo uma projeção dos dados similar as obtidas com a utilização das médias normalizadas de cada canal H, S e V, apresentadas no Apêndice B.

A partir de uma análise dos resultados finais descritos na seção anterior e conforme a Tabela 5.18, é possível perceber que a maioria dos itens obtiveram as melhores taxas de acerto com o classificador *SVM*, atingindo a maior média de acerto $92,24\%$.

Classificador	*k-NN*	*MLP*	*SVM*
Ascórbico	92%	90,33%	**94,67%**
Bilirrubina	**99,67%**	**99,67%**	**99,67%**
Cetonas	97%	95,33%	**97,33%**
Densidade	73,67%	75%	**76,67%**
Glicose	**100%**	**100%**	**100%**
Leucócitos	**89,67%**	89,33%	**89,67%**
Nitrito	99,33%	99,33%	**99,67%**
pH	69,33%	74%	**75,33%**
Proteínas	86,33%	86,33%	**89,33%**
Sangue	95%	**95,33%**	93%
Urobilinogênio	98,67%	98,67%	**99,33%**
Média	90,67%	91,21%	**92,24%**

Tabela 5.18: Melhores taxas de acerto de cada classificador.

No caso dos itens bilirrubina (99,67%) e glicose (100%), as taxas de acerto foram as mesmas para todos os classificadores utilizados, enquanto que os leucócitos atingiram o mesmo percentual com o *k-NN* e o *SVM*.

O único analito que não obteve o melhor percentual com o *SVM* foi o sangue, que atingiu 95,33% com o *MLP*. Além disso, alguns dos itens analisados apresentaram uma taxa de acerto abaixo dos 90%. É o caso dos analitos leucócitos, proteínas, densidade e pH que atingiram respectivamente 89,67%, 89,33%, 76,67% e 75,33% com o classificador *SVM*.

Tais resultados podem ser justificados pelas projeções dos dados apresentados no Apêndice B. É possível perceber que nesses casos há uma nuvem de exemplos de diferentes classes condensados, tornando a separação das amostras mais difícil.

Outra justificativa poderia se dar pela variação biológica que a urina possui. O fato da diferença no momento de molhar as fitas nas urinas, descrita na metodologia, poderia resultar na alteração em alguns valores dos analitos. Por isso, optou-se pela utilização da base *Patterns_2*, com o intuito de, talvez, obter melhores taxas de acerto. Essa segunda base passou pela mesma metodologia de extração de características e testes, mas não apresentou melhores percentuais. Do mesmo modo como na base *Patterns_1*, os dados dos analitos com os menores percentuais apresentaram-se condensados, dificultando a classificação.

Capítulo 6

Conclusão

No presente trabalho é proposta uma metodologia para análise das fitas reagentes no exame de urina, através do uso de um *scanner* e de técnicas simples de processamento de imagens e de aprendizado de máquina. Nesta, primeiramente construiu-se uma base de dados com imagens, adquiridas via *scanner*, de cada um dos itens analisados pela fita reagente. A partir destas imagens, a ideia foi extrair informações que pudessem auxiliar no processo de classificação.

A abordagem da extração de informações das imagens organizadas nas bases de dados *Patterns_*1 e *Patterns_*2 deu-se por meio da conversão das imagens para o modelo de cores HSV e da obtenção dos valores normalizados de cada um dos canais. Além da conversão para o modelo HSV, também foram feitos testes com o modelo RGB. Com o intuito de se ter mais informações na utilização do modelo HSV, optou-se por testar a utilização de mais seis valores como atributos, além das médias normalizadas. Esses valores foram obtidos a partir da normalização linear do máximo e mínimo dos canais H, S e V de cada imagem. Os valores adquiridos foram organizados no arquivo ARFF para que os mesmos fossem utilizados pela ferramenta *Weka* no processo dos testes.

Para fins de teste, foram selecionados três classificadores, com a intenção de se avaliar a melhor abordagem para o problema da classificação das imagens das tiras reagentes. Foram utilizados nesse trabalho o algoritmo *k-NN (k-Nearest Neighbors)*, *MLP (Multi-layer Perceptron)* e o *SVM (Support Vector Machine)*.

Para cada classificador, foram manipulados seus parâmetros particulares. No caso do *k-NN*, o único parâmetro a ser variado foi o número de vizinhos mais próximos, que mostrou ter melhores resultados com $k = 1$. Para o *MLP*, foram variados o número de camadas e sua quantidade de neurônios, e posteriormente a taxa de aprendizado, momento e ciclos de treinamento. Os percentuais obtidos na primeira etapa de testes com esse algoritmo, que foi na variação do número de camadas e de seus neurônios, mostraram-se equivalentes aos percentuais obtidos após o refinamento dos demais parâmetros, apresentando melhorias sutis. Como no caso da densidade que obteve a maior melhoria, de 3%, atingindo 75%. No caso do classificador *SVM*, a manipulação do *kernel* e dos parâmetros gama, C e grau do polinômio, no caso do *kernel* polinomial, trouxe boas melhorias para os resultados.

Os testes também mostraram que a utilização do modelo RGB e do modelo HSV com os valores normalizados de máximo e mínimo de cada canal, não obtiveram resultados significativos. Mesmo com o uso desses valores, não ocorreram mudanças em relação a separação dos dados. Isso porque não foram obtidas informações adicionais que pudessem

discriminar melhor as classes, obtendo uma projeção dos dados similar as obtidas com a utilização das médias normalizadas de cada canal H, S e V, apresentadas no Apêndice B.

Além disso, os três classificadores mostraram-se eficientes na classificação dos dados e com médias de taxa de acerto próximas. O *k-NN* obteve uma média de $90,67\%$, a mais baixa dentre os algoritmos. O *MLP*, com a segunda melhor média obteve $91,21\%$ de taxa de acerto e o *SVM* mostrou-se o melhor classificador para a proposta, com percentual de acerto de $92,24\%$.

Dos onze itens analisados, os quatro que apresentaram os menores percentuais foram os leucócitos, proteínas, densidade e pH que atingiram respectivamente $89,67\%$, $89,33\%$, $76,67\%$ e $75,33\%$. Tais resultados podem ser justificados pelas projeções dos dados da densidade conforme a Figura 6.1 e do pH conforme a Figura 6.2.

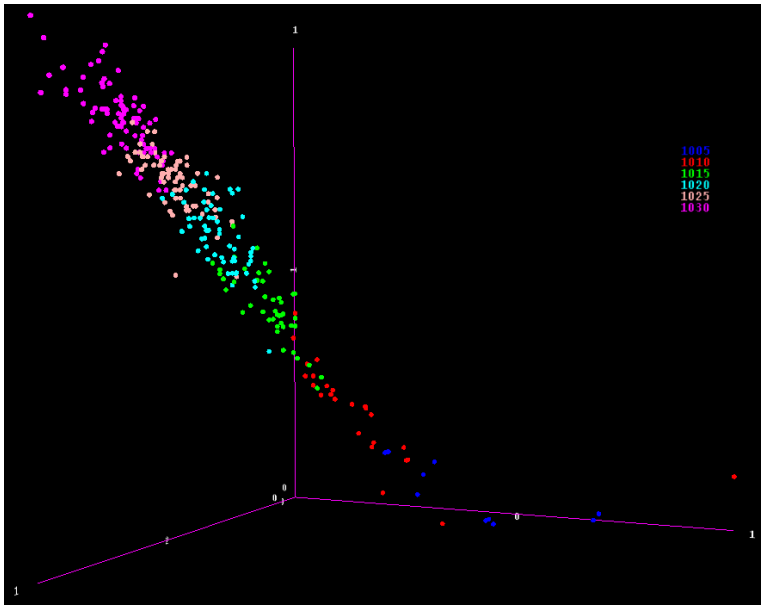

Figura 6.1: Projeção dos dados do padrão densidade.
Fonte: o autor.

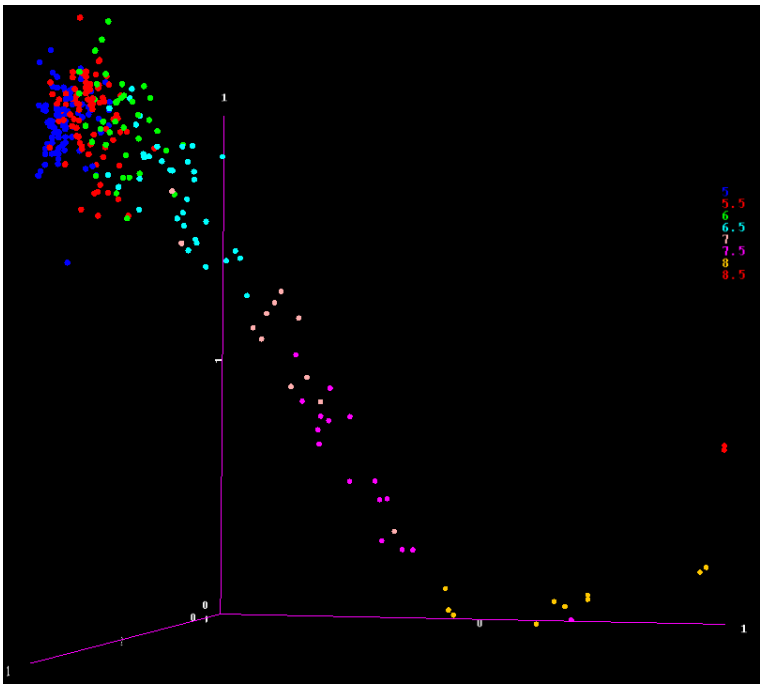

Figura 6.2: Projeção dos dados do padrão pH.
Fonte: o autor.

É possível perceber que nos casos da densidade e do pH há uma nuvem de exemplos de diferentes classes condensados, tornando a separação das amostras mais difícil. No caso da densidade, as classes 1015, 1020, 1025 e 1030 são as que se encontram mais embaraçadas, no canto superior esquerdo do gráfico, enquanto que para o analito pH, os valores 5, $5,5$ e 6 são os que apresentam maior proximidade.

Apesar dos resultados promissores, é necessário o aprimoramento dos métodos de extração e combinação de características, otimizar a manipulação dos parâmetros dos algoritmos e obter mais amostras dos casos extremos de cada padrão, que pudessem auxiliar na melhor definição das funções dos classificadores, podem ser abordados em trabalhos futuros. Além disso, um estudo e avaliação das fitas reagentes faz-se necessária, medindo a variação das respostas entre fitas coletadas a partir de uma mesma amostra.

Referências Bibliográficas

[ABNT, 2005] ABNT (2005). *Associação Brasileira de Normas Técnicas. Laboratório clínico. Requisitos e recomendações para o exame da urina.*

[Amorim et al., 2009] Amorim, A. E., Pacheco, J. B. P., Fernandes, T. T., and Biomedicina, F.-A. (2009). Exame de urina tipo i: freqüência percentual de amostras que sugerem infecção urinária. *Anuário da Produção de Iniciação Científica Discente. 2008; 24 (12): 57*, 68.

[Backer, 1995] Backer, E. (1995). *Computer-assisted Reasoning in Cluster Analysis*. Prentice Hall International (UK) Ltd., Hertfordshire, UK, UK.

[Bolodeoku and Donaldson, 1996] Bolodeoku, J. and Donaldson, D. (1996). Urinalysis in clinical diagnosis. *Journal of clinical pathology*, 49(8):623.

[Chang and Lin, 2016] Chang, C.-C. and Lin, C.-J. (2016). A library for support vector machines. `https://weka.wikispaces.com/LibSVM`. Acessado em 15/09/2016.

[Chien et al., 2007] Chien, T.-I., Kao, J.-T., Liu, H.-L., Lin, P.-C., Hong, J.-S., Hsieh, H.-P., and Chien, M.-J. (2007). Urine sediment examination: a comparison of automated urinalysis systems and manual microscopy. *Clinica Chimica Acta*, 384(1):28–34.

[Cobas, 2016] Cobas (2016). Cobas u 411 analyser. `http://www.cobas.com/home/product/urinalysis-testing/cobas-u-411-urine-analyzer.html`. Acessado em 15/08/2016.

[Cover and Hart, 1967] Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27.

[Damasceno, 2015] Damasceno, M. (2015). Introdução a mineração de dados utilizando o weka. *Disponível: http://connepi. ifal. edu. br/ocs/index. php/connepi/CONNEPI2010/paper/viewFile/258/207*. Acessado em 15/09/2016.

[Falcão et al., 2013] Falcão, H. S., Lovato, A. V., dos Santos, A. F., Lucas Santos de Oliveira, R., Guimarães, M., and Santana, M. (2013). Classificação de vagas de estacionamento com utilização de rede perceptron multicamadas. *Revista de Sistemas de Informação da FSMA*, pages 41–48.

[Ferrero, 2009] Ferrero, C. A. (2009). *Algoritmo kNN para previsão de dados temporais: funções de previsão e critérios de seleção de vizinhos próximos aplicados a variáveis ambientais em limnologia*. PhD thesis, Universidade de São Paulo.

[Fiorin et al., 2011] Fiorin, D. V., Martins, F. R., Schuch, N. J., and Pereira, E. B. (2011). Aplicações de redes neurais e previsões de disponibilidade de recursos energéticos solares. *Revista Brasileira de Ensino de Fısica*, 33(1):1309.

[Friedman et al., 2001] Friedman, J., Hastie, T., and Tibshirani, R. (2001). *The elements of statistical learning*, volume 1. Springer series in statistics Springer, Berlin.

[Haykin, 2001] Haykin, S. S. (2001). *Redes neurais*. Bookman.

[Joachims, 1998] Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer.

[Joachims, 2002] Joachims, T. (2002). *Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA.

[Kramer, 2013] Kramer, O. (2013). K-nearest neighbors. In *Dimensionality Reduction with Unsupervised Nearest Neighbors*, pages 13–23. Springer.

[Lima et al., 1992] Lima, A. O., Soares, J. B., Greco, J., Galizzi, J., and Cançado, J. R. (1992). *Métodos de laboratório aplicados à clínica: técnica e interpretaçäo*. Guanabara Koogan.

[Lopes, 2004] Lopes, H. (2004). O laboratório clínico na avaliação da função renal. *Belo Horizonte: Gold Analisa Diagnóstico Ltda*, page 27.

[Marengoni and Stringhini, 2009] Marengoni, M. and Stringhini, S. (2009). Tutorial: Introdução à visão computacional usando opencv. *Revista de Informática Teórica e Aplicada*, 16(1):125–160.

[McCulloch and Pitts, 1943] McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.

[Medical News Today, 2016] Medical News Today, M. (2016). Innovative smartphone app tests your urine for medical issues. `http://www.medicalnewstoday.com/articles/256974.php`. Acessado em 15/08/2016.

[Monard and Baranauskas, 2003] Monard, M. C. and Baranauskas, J. A. (2003). Conceitos sobre aprendizado de máquina. *Sistemas Inteligentes-Fundamentos e Aplicações*, 1(1).

[Noble et al., 2004] Noble, W. S. et al. (2004). Support vector machine applications in computational biology. *Kernel methods in computational biology*, pages 71–92.

[Oliveira et al., 2010] Oliveira, L. F. d., Kist, D. M., Cavalheiro, G. G. H., Meneghello, G. E., and Tillmann, M. A. A. (2010). Segmentação de imagens com fundo azul utilizando a multiplicação dos canais hsv.

[Peterson, 2009] Peterson, L. E. (2009). K-nearest neighbor. *Scholarpedia*, 4(2):1883.

[Pontil and Verri, 1998] Pontil, M. and Verri, A. (1998). Support vector machines for 3D object recognition. *IEEE transactions on pattern analysis and machine intelligence*, 20(6):637–646.

[Ravel, 1995] Ravel, R. (1995). *Laboratório clínico: aplicações clínicas dos dados laboratoriais*. Guanabara Koogan.

[Reine and Langston, 2005] Reine, N. J. and Langston, C. E. (2005). Urinalysis interpretation: how to squeeze out the maximum information from a small sample. *Clinical techniques in small animal practice*, 20(1):2–10.

[REPRESENTANTES, 2004] REPRESENTANTES, E. (2004). Laboratório clínico–requisitos e recomendações para o exame da urina.

[Roche, 2016] Roche (2016). Urisys 1100 urine analyzer. `https://usdiagnostics.roche.com/en/instrument/urisys-1100.html`. Acessado em 15/08/2016.

[Rumelhart et al., 1985] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1985). Learning internal representations by error propagation. Technical report, DTIC Document.

[Silva, 2005] Silva, R. R. d. (2005). *Reconhecimento de Imagens Digitais Utilizando Redes Neurais Artificiais*. PhD thesis, Universidade Federal de Lavras.

[Simões, 2000] Simões, A. d. S. (2000). *Segmentação de imagens por classificação de cores: uma abordagem neural*. PhD thesis, Universidade de São Paulo.

[Sklearn, 2016] Sklearn, S. (2016). Support vector machines. `http://scikit-learn.org/stable/modules/svm.html`. Acessado em 30/09/2016.

[Souza, 1999] Souza, J. A. d. (1999). *Reconhecimento de padrões usando indexação recursiva*. PhD thesis, Universidade Federal de Santa Catarina, Centro Tecnológico.

[Strasinger, 2000] Strasinger, S. (2000). Uroanálise e fluidos biológicos.

[Strasinger and Torquettitoloi, 1996] Strasinger, S. K. and Torquettitoloi, M. R. (1996). *Uroanálise e fluidos biológicos*. Editorial Premier.

[University of Waikato, 2016] University of Waikato, M. L. G. (2016). Weka. `http://www.cs.waikato.ac.nz/ml/weka/`. Acessado em 15/09/2016.

[Van Rossum et al., 2007] Van Rossum, G. et al. (2007). Python programming language. In *USENIX Annual Technical Conference*, volume 41.

[Vapnik and Vapnik, 1998] Vapnik, V. N. and Vapnik, V. (1998). *Statistical learning theory*, volume 1. Wiley New York.

[White, 2016] White, W. (2016). Color models. `http://www.cs.cornell.edu/courses/cs1133/2014fa/assignments/assignment2/`. Acessado em 25/09/2016.

[YD Diagnostics, 2016] YD Diagnostics, Y. (2016). Urican pro ii analyzer. `http://www.yd-diagnostics.com/new/product/uriscan_3.php`. Acessado em 15/08/2016.

Apêndice A

Quantidade de imagens por padrão

As Tabelas A.1 até A.11 a seguir apresentam a quantidade de imagens referentes a cada padrão de cada analito da Base *Patterns*_1.

- **Ácido Ascórbico**

Padrão	10	25	50	neg
Quantidade	63	18	11	208

Tabela A.1: Quantidade de imagens para o padrão ácido ascórbico.

- **Bilirrubina**

Padrão	0.5	1	neg
Quantidade	2	5	293

Tabela A.2: Quantidade de imagens para o padrão bilirrubina.

- **Cetonas**

Padrão	5	10	50	100	neg
Quantidade	10	14	7	7	262

Tabela A.3: Quantidade de imagens para o padrão cetonas.

- **Densidade**

Padrão	1005	1010	1015	1020	1025	1030
Quantidade	11	25	46	61	71	86

Tabela A.4: Quantidade de imagens para o padrão densidade.

- **Glicose**

Padrão	100	250	500	1000	2000	neg
Quantidade	4	6	5	4	14	267

Tabela A.5: Quantidade de imagens para o padrão glicose.

- **Leucócitos**

Padrão	10	25	75	500	neg
Quantidade	14	20	17	28	221

Tabela A.6: Quantidade de imagens para o padrão leucócitos.

- **Nitrito**

Padrão	pos	neg
Quantidade	24	276

Tabela A.7: Quantidade de imagens para o padrão nitrito.

- **pH**

Padrão	5	5.5	6	6.5	7	7.5	8	8.5
Quantidade	96	90	38	36	12	16	10	2

Tabela A.8: Quantidade de imagens para o padrão pH.

- **Proteínas**

Padrão	10	30	100	300	1000	neg
Quantidade	18	18	29	6	5	224

Tabela A.9: Quantidade de imagens para o padrão proteínas.

- **Sangue**

Padrão	5	10	50	250	neg
Quantidade	17	14	27	32	210

Tabela A.10: Quantidade de imagens para o padrão sangue.

- **Urobilinogênio**

Padrão	1	4	8	norm
Quantidade	14	2	2	282

Tabela A.11: Quantidade de imagens para o padrão urobilinogênio.

Apêndice B

Projeção 3D dos dados

A projeção de cada um dos analitos foi feita a partir da conversão das imagens da base *Patterns_*1 para o modelo de cores HSV e dos atributos sendo representados pelas médias normalizadas de cada canal.

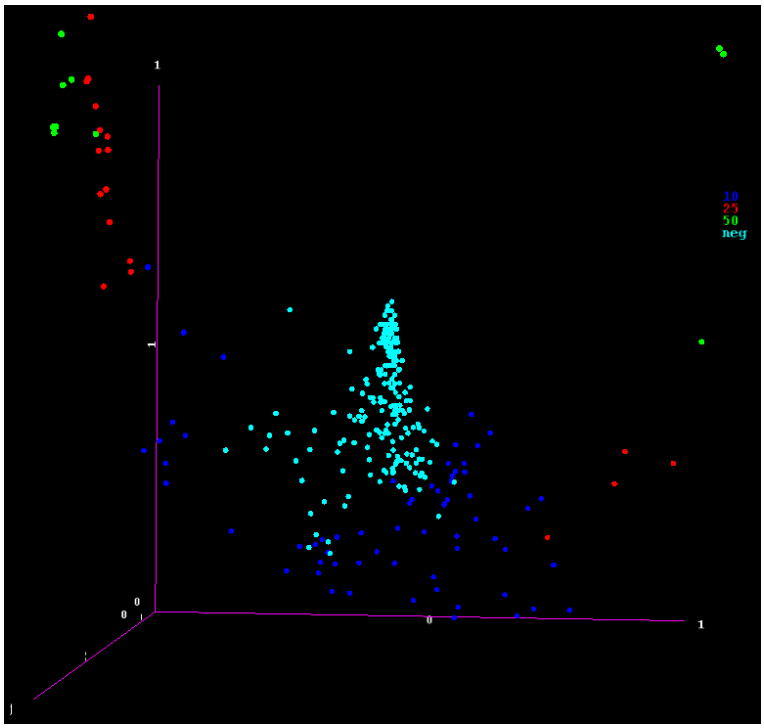

Figura B.1: Projeção dos dados do padrão ácido ascórbico.
Fonte: o autor.

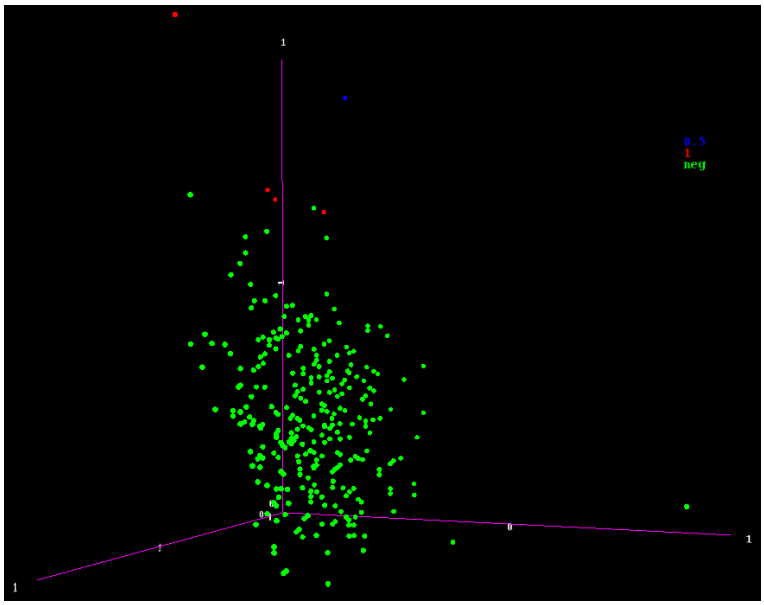

Figura B.2: Projeção dos dados do padrão bilirrubina.
Fonte: o autor.

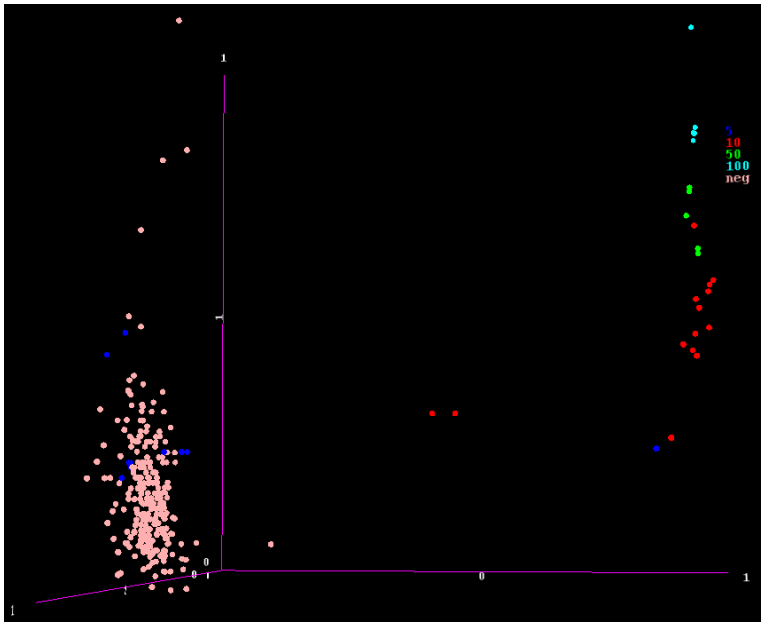

Figura B.3: Projeção dos dados do padrão cetonas.
Fonte: o autor.

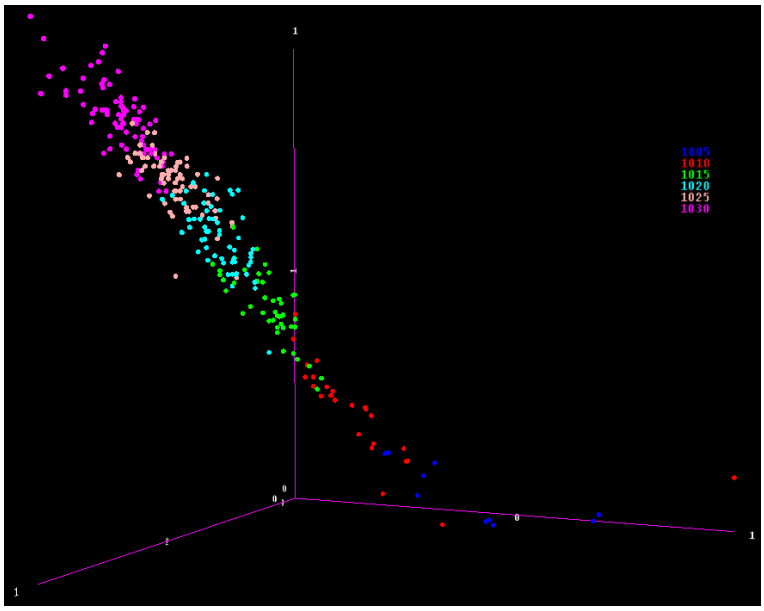

Figura B.4: Projeção dos dados do padrão densidade.
Fonte: o autor.

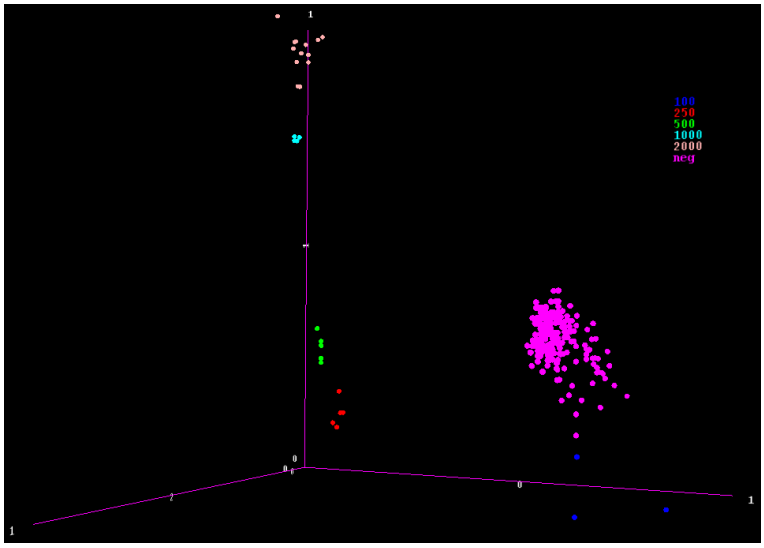

Figura B.5: Projeção dos dados do padrão glicose.
Fonte: o autor.

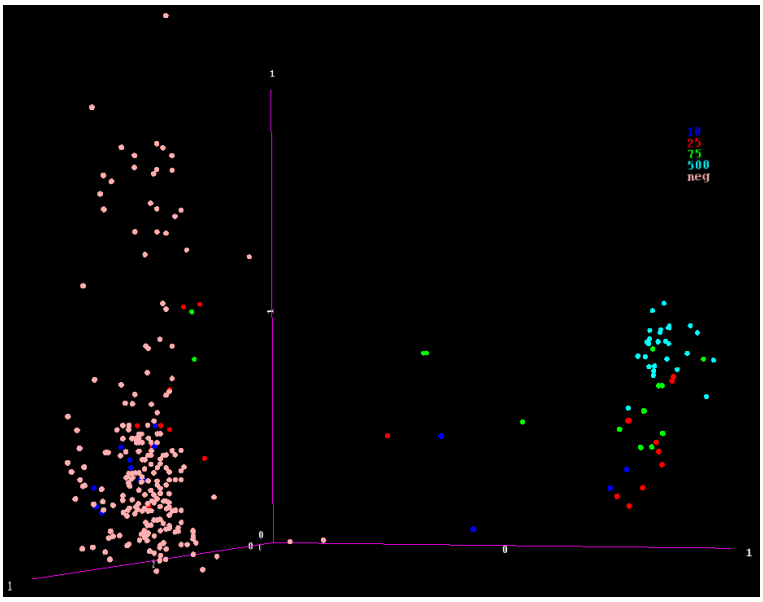

Figura B.6: Projeção dos dados do padrão leucócitos.
Fonte: o autor.

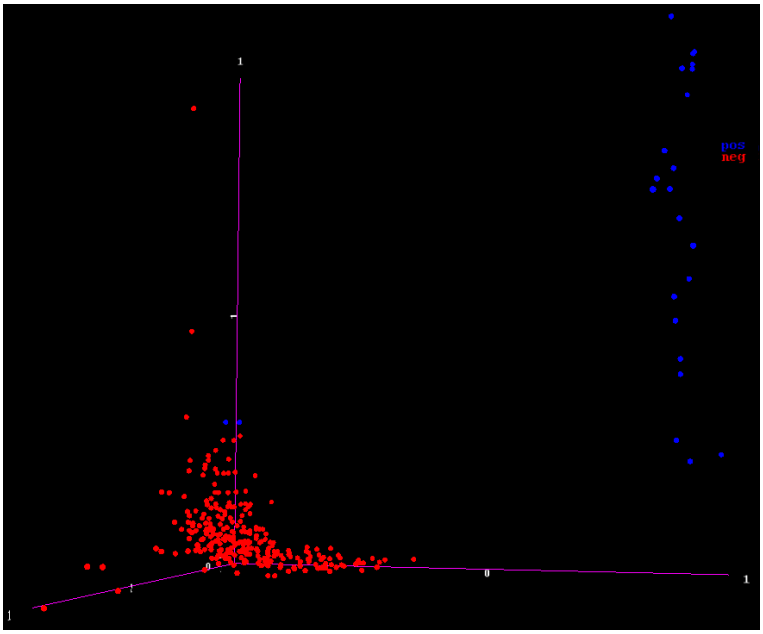

Figura B.7: Projeção dos dados do padrão nitrito.
Fonte: o autor.

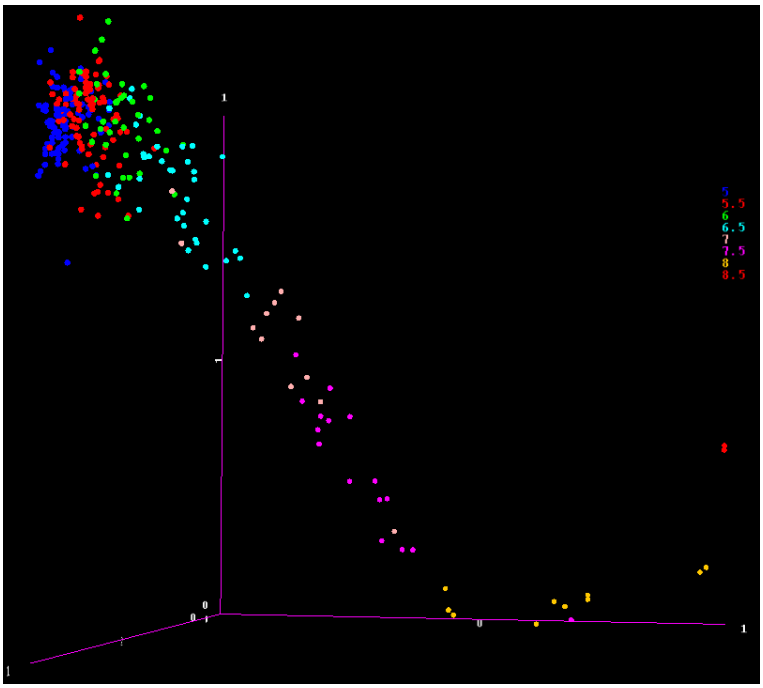

Figura B.8: Projeção dos dados do padrão pH.
Fonte: o autor.

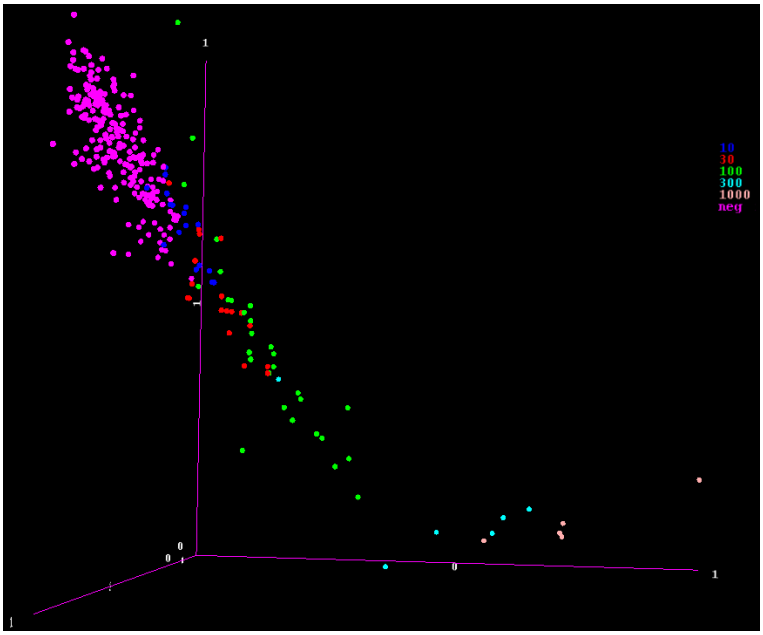

Figura B.9: Projeção dos dados do padrão proteínas.
Fonte: o autor.

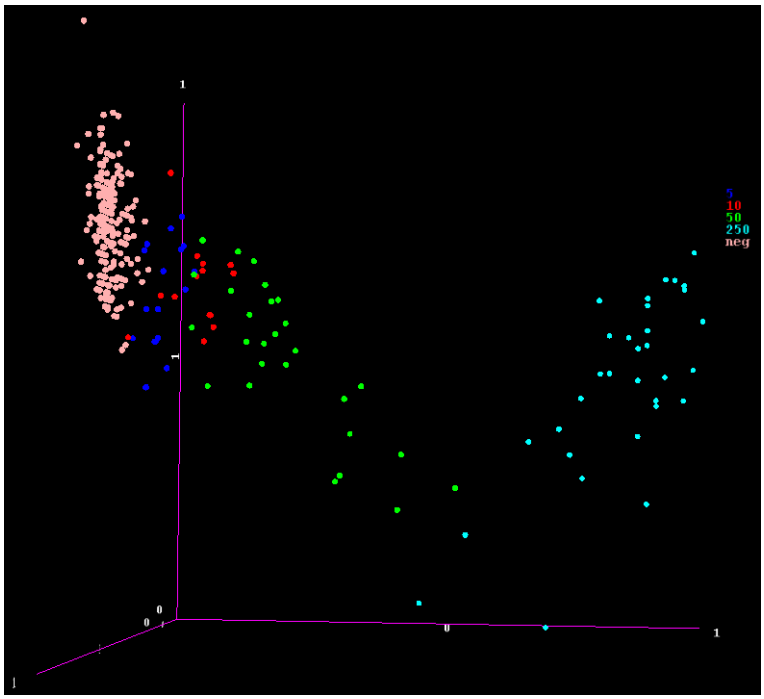

Figura B.10: Projeção dos dados do padrão sangue.
Fonte: o autor.

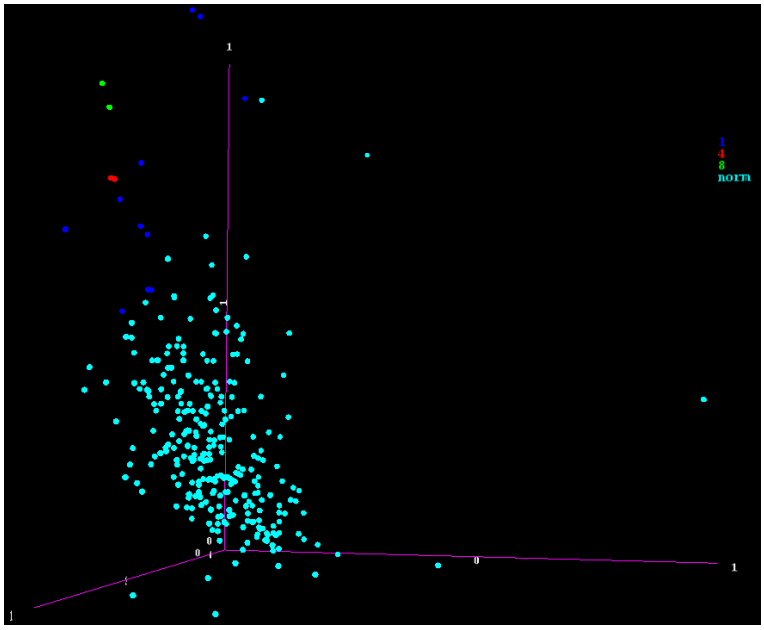

Figura B.11: Projeção dos dados do padrão urobilinogênio.
Fonte: o autor.