

Smartphone-Based Recognition of Human Activities and Postural Transitions

Bruno M. Dobrovolski¹, Eduardo A. Schmoller¹

¹Departamento Acadêmico de Informática – Universidade Tecnológica Federal do Paraná
Pato Branco – PR – Brasil

{brunod,schmoller}@alunos.utfpr.edu.br

1. Introdução

Este trabalho apresentará resultados da aplicação de métodos de aprendizado de máquinas na predição de movimentos humanos. Foram implementados os seguintes métodos de aprendizado: KNN, Perceptron e SVM.

Os dados para base para treinamento e teste foram obtidos de [Dheeru and Karra Taniskidou 2017]. Essas informações são referente a sensores de um dispositivo móvel Galaxy Samsung S2. O *smartphone* foi mantido junto ao corpo dos voluntários em atividades de: Levantar, sentar, deitar, caminhar, subir escadas e descer escadas. Os testes foram executados com 30 voluntários e os dados coletados através dos sensores do smartphone, acelerômetro de 3 eixos e giroscópio de 3 eixos.

Os dados obtidos dos sensores foram pré-processados com a aplicação de filtros para remoção de ruídos, amostrados em intervalos constantes de tempo e normalizados. Também foram segmentados separando cada um dos 3 eixos e calculando várias informações estatísticas (totalmente descritas em citeDua:2017), como média, desvio padrão, máximo, mínimo, entre outras, totalizando 561 características. Cada conjunto de dados que representa os movimentos recebeu uma identificação. O conjunto de características e de identificações (*label*) são entradas para os algoritmos de classificação.

Como forma de validação do processo de aprendizado, o conjunto de dados de entrada foi dividido em dois subconjuntos denominados conjunto de treino e conjunto de testes. O conjunto de treino é utilizado no processo de treinamento, após o processo de treinamento o conjunto de testes é classificado e os resultados são comparados com os resultados esperados.

Ainda, os algoritmos foram alterados através da redução de dimensionalidade com o uso do método PCA. Como esse método não entrega um resultado ótimo, foi necessário variar o número de componentes para encontrar a melhor combinação.

Como métrica para a determinação da qualidade do processo de aprendizados foram utilizadas a acurácia da predição e utilização do *kappa score*. O *kappa score* funciona como uma validação do resultado apresentado pela acurácia, visto que ele considera em seu calculo os resultados falso-positivos e falso-negativos, colocando peso em cada uma das características.

2. Revisão

Esta seção apresenta os métodos aplicados para a classificação dos movimentos.

2.1. Perceptron

Separador linear através de uma função de *threshold*, reta esta que é definida através de pesos. O treinamento de um perceptron consiste na entrada de um conjunto de casos e calcular a saída para cada caso, após cada entrada os pesos são ajustados de forma a minimizar o erro na saída, o erro é definido como a diferença entre a saída e a saída esperada, não existe erro caso a saída e a saída esperada sejam iguais.

Perceptron de uma única camada apenas separam problemas lineares, para a classificação de classes que não sejam separáveis linearmente é necessário um perceptron de múltiplas camadas.

2.2. KNN

KNN algoritmo simples que realiza o aprendizado de todas as entradas e classifica as novas entradas com base na similaridade dos casos já aprendidos. Ao realizar a classificação de uma nova entrada é retornado k vizinhos mais próximos, classificação se dá pela maioria dos vizinhos.

2.3. SVM

SVM é um classificador definindo a separação entre as classes através de um hiperplano, a entrada são dados com *labels* e a saída é um hiperplano ótimo para categorização de novas entradas. Em duas dimensões o hiperplano é representado através de uma reta que divide o plano em duas partes.

2.4. PCA

Redutor de dimensionalidade selecionando as principais componentes, entre as formas de encontrar as componentes principais é buscar as projeções que maximizem a variância.

2.5. Testes Estatísticos

Testes estatísticos são utilizados para validação de hipóteses usando dados amostrais para decidir sobre a validade da hipótese.

Cenários típicos de uso de testes estatísticos em métodos de aprendizado: adição de etapas de processamento, comparar diversos algoritmos, validação de capacidade de generalização de um classificador. O princípio básico é a formulação de hipóteses e decidir se os dados mostram alguma evidencia para rejeição da hipótese.

2.5.1. Validação Cruzada

Validação cruzada é uma técnica que consiste em reservar uma parte do conjunto de dados para serem utilizados para testes após a etapa de treinamento.

Exemplos de métodos para validação cruzada: *HoldOut* (dividir em treino e teste), *K-fold* (k sub conjuntos de treino e teste), *Stratified K-Fold* (cada conjunto contem aproximadamente a mesma porcentagem de amostras de cada classe que o conjunto completo), *Leave-P-Out* (em um conjunto de n casos então n-p serão utilizados para treino e p casos para teste).

2.5.2. Matriz de Confusão

Matriz de confusão contém informações sobre o estado real e o estado previsto por um classificador, a performance do sistema é avaliada usando os dados da matriz, em um sistema com duas classes os valores que podem ser calculados são: acurácia (proporção de predições corretas), recall (proporção de casos positivos classificados corretamente), falsos positivos (proporção de casos negativos classificados incorretamente), negativos (proporção de negativos classificados corretamente), falsos negativos (proporção de positivos classificados incorretamente), precisão (proporção de casos positivos classificados corretamente).

2.5.3. Kappa Score

Kappa Score mede a confiabilidade de uma classificação entre múltiplos classificadores para um mesmo conjunto de dados, mede a porcentagem de valores de dados na diagonal principal da tabela e, em seguida, ajusta esses valores para a quantidade de concordância que poderia ser esperada em uma classificação aleatória.

Possível interpretação do resultado do Kappa Score:

Pouca concordância = Menor que 0.20

Razoável concordância = 0.20 até 0.40

Moderada Concordância = 0.40 até 0.60

Boa concordância = 0.60 até 0.80

Concordância muito boa = 0.80 até 1.00

3. Resultados

Com a aplicação dos métodos citados anteriormente foi possível apresentar a Figura 1. Esse gráfico representa a busca pelo melhor agrupamento de características utilizando PCA. Como é possível observar, cada um dos métodos tem um comportamento diferente. O Perceptron tem melhores resultados quanto maior for a quantidade de características; Já o KNN não apresenta muitas melhoras; Por fim o SVM tem uma redução da acurácia para maior quantidade de características. É importante notar que apesar da variação nos resultados, as perdas são relativamente pequenas, cerca de 5%. O código executado partiu de 10 grupos a 260 grupos, com incremento de 10

Porém, é sabido que os bons resultados da acurácia podem não ser reais. Para isso, foi aplicado o Kappa-Score, como explicado anteriormente ele pondera os falso-positivos e falso-negativos, quando comparado à acurácia a diferença entre esses dois valores informa a qualidade do método. A Figura 2, apresenta essa comparação em relação aos 3 métodos.

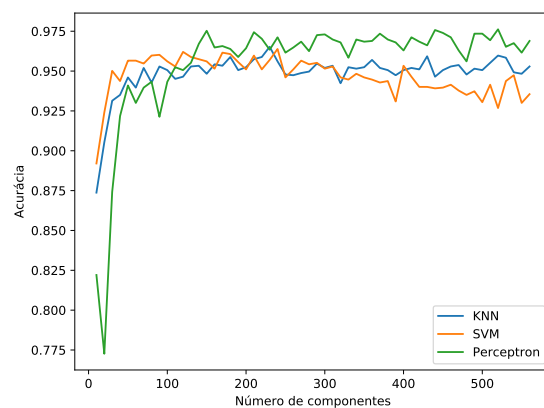
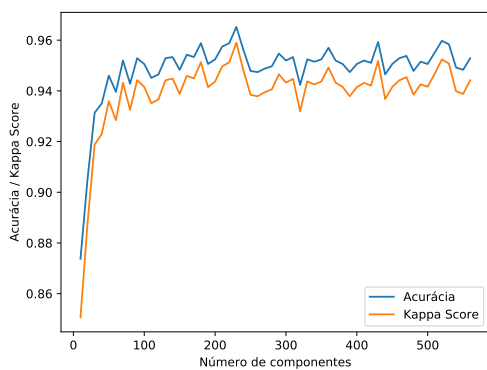
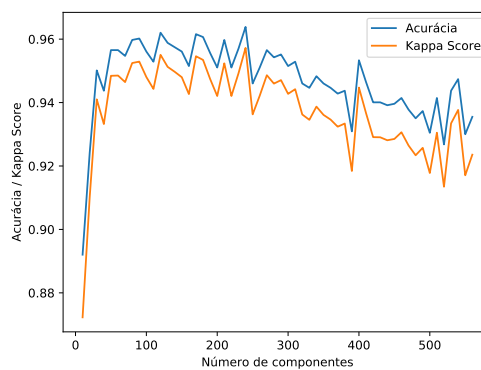


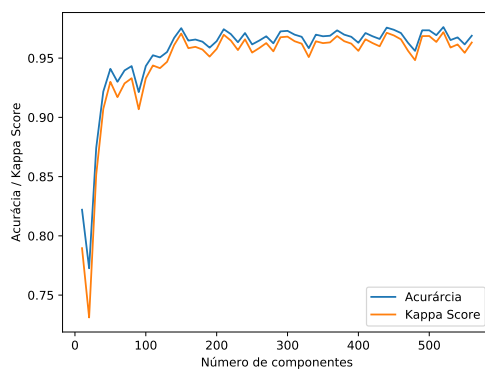
Figura 1. Acurácia - KNN, SVM, Perceptron



(a) Comparação - KNN



(b) Comparação - SVM



(c) Comparação - Perceptron

Figura 2. Comparação entre Acurácia e Kappa Score

Observando esses resultados, foi possível obter o melhor agrupamento para cada um dos métodos, essa escolha é apresentada na Tabela 1.

Para o caso ótimo desses conjuntos foi construída a matriz de confusão, afim de validar as informações apresentadas pela acurácia e Kappa Score. As matrizes apresenta-

Tabela 1. Resultados			
Método	Acurácia	Kappa Score	Agrupamento
KNN	0.9652	0.9589	230
Perceptron	0.9762	0.9718	520
SVM	0.9663	0.9572	240

das na Figura 3

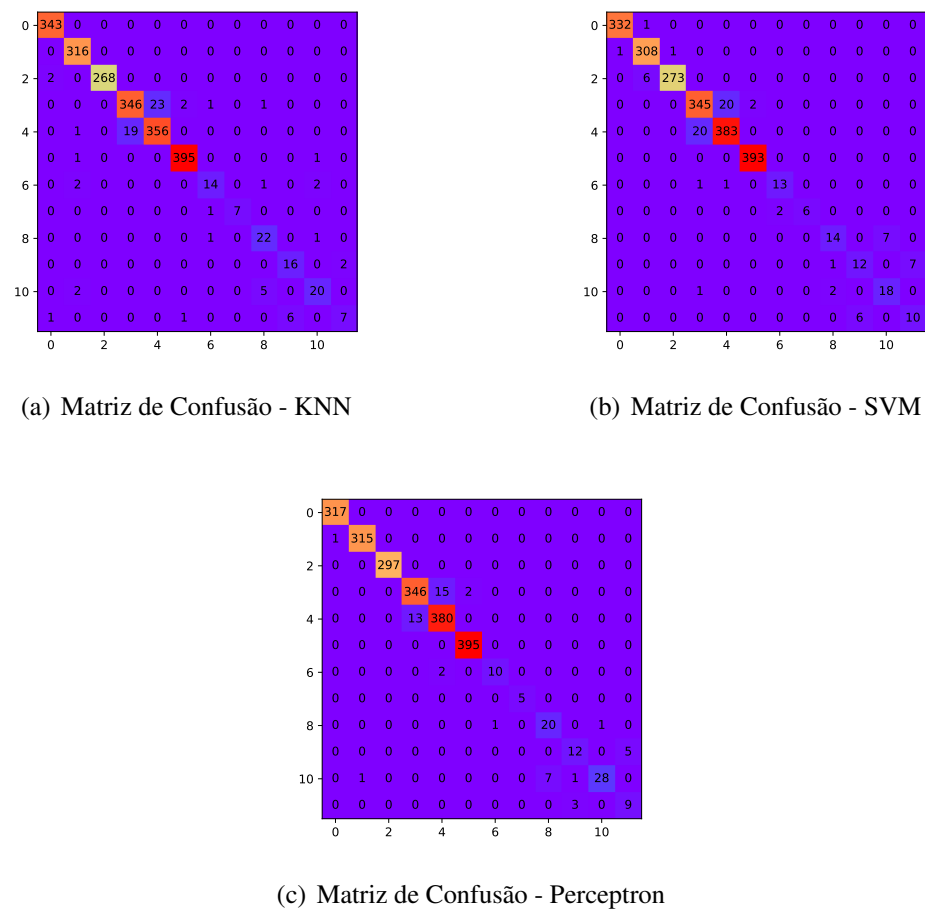


Figura 3. Comparação entre Matrizes de confusão

4. Conclusões

Este trabalho permitiu a compreensão de alguns métodos de aprendizagem. A escolha de cada método deve ser adequada ao tipo dos dados a serem aprendidos e, neste caso, a não uniformidade da quantidade de informações sobre cada característica gerou uma diferença considerável entre cada método. Também notou-se que a técnica PCA não necessariamente colabora com os resultados, por exemplo, no método Perceptron, quanto mais características, melhor foram os resultados. Outro fator notável, foi o ganho ao aumentar a quantidade de classes, isso gerou um ganho relativamente pequeno de acurácia e aumentou o tempo de computação consideravelmente. De forma conclusiva, não houve um método com destaque nos resultados obtidos, as diferenças foram sutis e todos foram considerados satisfatórios.

Referências

[Dheeru and Karra Taniskidou 2017] Dheeru, D. and Karra Taniskidou, E. (2017). {UCI} machine learning repository.