FUNDAÇÃO GETULIO VARGAS

ESCOLA DE MATEMÁTICA APLICADA

MESTRADO 2015.3

APRENDIZAGEM POR MÁQUINAS

Prof Eduardo Mendes

Projeto Final

KIZZY TERRA

LUCAS MACHADO

RIO DE JANEIRO

DEZEMBRO/2015

1 Introdução

Este relatório visa descrever o processo de implementação das soluções submetidas na Competição de Reconhecimento de Dígitos do Kaggle (https://www.kaggle.com/c/digit-recognizer). O desempenho destas soluçõeos nesta competição será utilizado na disciplina de Aprendizado de Máquina, correspondendo ao seu projeto final, a fim de avaliar a habilidade dos alunos de utilizar técnicas de machine learning para solucionar um problema real.

2 A Competição de Reconhecimento de Dígitos

O objetivo da competição é reconhecer dígitos em imagens que contém dígitos escritos a mão. Os dados para a compentição foram retirados do MNIST dataset. O MNIST ("Modified National Institute of Standards and Technology") dataset é um conjunto de dados clássico dentro da comunidade de Aprendizado de Máquina e tem sido largamente estudado (mais detalhes em http://yann.lecun.com/exdb/mnist/index.html).

3 Escolha de *features*

A escolha das *features* a serem utilizadas em cada algoritmo é uma etapa importante na proposta de uma solução adequada. Quando lidamos com imagens consideramos cada pixel como uma dimensão diferente; é fácil perceber que para uma imagem não muito grande o número de dimensões já se torna grande e a correlação entre elas também. Para contornar este problema, utilizam-se técnicas como análise de componente principais (PCA) e *data* whitening.

A análise de componentes principais (PCA) é um método para reduzir as dimensões de um conjunto de dados. Isto é feito, explorando-se correlações entre algumas das dimensões. Por essa razão, a utilização deste tipo de método é bastante conveniente em contextos como o de reconhecimento de imagens.

A análise de componentes principais foi feita utilizando-se a classe *decomposition* do pacote *Scikit Learn.* Uma vez feita a decomposição foi possível gerar um gráfico com a relação entre fração de variância e o número de componentes geradas. A informação obtida com este gráfico acarretou a conlusão de que utilizar 35 componentes poderia levar a um resultado satisfatório.

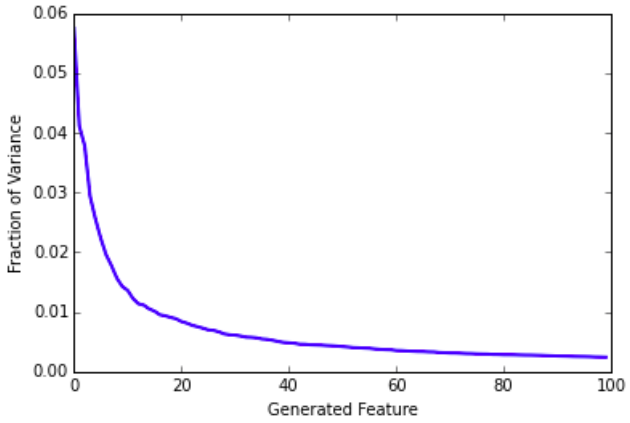

Figura 1: Variância x Número de *features* geradas

O *data* whitening é utilizado para duas principais finalidades: diminuir a correlação entre as features existentes e forçar todas as *features* a terem a mesma variância. Esta método é bastante utilizad no contexto de classificação de imagens devido a alta redundância de informação presente em pixels adjacentes. No caso de imagens, pixels próximos fornecem informações altamente correlacionadas e o objetivo do *whitening* é diminuir a quantidade de informação redundante.

4 Seleção de Modelos

A seleção de modelos adequados a serem utilizados na classificação de um determinado conjunto de dados deve considerar variáveis como por exemplo a natureza destes dados - categóricos ou numéricos? - , a quantidade de dados disponível, a complexidade da classificação, a acurácia na classifiação e a velocidade de classificação.

No contexto de uma competição do Kaggle a variável de maior prioriadade é a acurácia na classificação uma vez que submete-se o resultado da classificação sobre o conjunto de teste e não o algorimto implementado em si.

Evidentemente, a acurácia de um classificador baseia-se no conjunto de dados que se está analisando, consequentemente esta escolha exige experimentação e por essa razão a etapa de seleção de modelos consiste de elencar alguns modelos candidatos cuja performance deverá ser comparada para a classificação do conjunto de dados do MNIST. A escolha destes candidatos por sua vez, leva em conta as características e vantagens dos principais modelos conhecidos. Os classificadores escolhidos estão descritos na seção 5 a seguir.

5 Classificadores Utilizados

Nesta seção descreve-se os modelos selecionados os quais foram utilizados para a classificação do conjunto de dados do MNIST. A descrição destes algoritmos compreende uma breve análise teórica

- incluindo comentários sobre a seleção de parâmetros - e uma análise da respectiva performance obtida.

5.1 Linear

O primeiro modelo utilizado foi um modelo básico linear....

5.2 k-NN

O k-NN té uma generalização do algoritmo *Nearest Neighbors* o qual consiste de considerar a informação dos pontos vizinhos mais próximos como uma aproximação para um determindado ponto que se quer classificar. Este é um algoritmo simples cuja vantagem é a não exigência de tempo de treinamento, entretanto os requisitos de memória e tempo de classificação são relativamente grandes.

O parâmetro k controla o trade off entre a aproximação e a generalização. Escolhendo$k$ muito pequeno obtém-se uma regra de decisão complexa, o que causa overfitting nos dados; Por outro lado, um$k$ muito garnde conduz a underfitting. Embora $k = 3$ funcione bem quando E_{out} é pequeno, em situações mais gerais precisamos escolher entre os diferentes k: cada valor de k, por sua vez, corrsponde será um modelo diferente. Para cada caso, portanto, deve-se o melhor valor de k.

Para a classificação dos dígitos utilizou-se k=?? a acurácia obtida foi ???

5.3 SVM

Outro modelo escolhido para experimentação de classificação do conjudo de dados da competição foi o SVM. Um modelo SVM é uma representação dos exemplos presentes no conjunto de treinamento como pontos no espaço, mapeados de maneira que os exemplos de cada classe sejam divididos por hiperplanos de separação. Em outra palavras, dado um conjunto de dados de treinamento já classificado (aprendizado supervisionado) o algoritmo gera um hiperplano ótimo que categoriza os novos exemplos presentes no conjunto de teste.

Combinado com este algoritmo utilizou-se um algoritmo de análise de componentes principais a fim de selecionar as melhores features a serem utilzadas na classificação. Como mencionado na seção 3 foram selecionadas as 35 principais componentes a partir do conjunto de treinamento. O subconjunto dos dados obtido a partir da decomposição dos dados foi utilizado como entrada para o classificador SVM.

Para utilizar o SVM foi necessário também escolher seus parâmetros de forma cuidadosa....

inicialmente utilizou-se o kernel gaussiano e a acuracia obtida foi..

em seguida resolveu-se utilizar o kernel rbf

Este algoritmo foi o que resultou na melhor acurácia de 98.283%..

Resultados, erros in-sample e out–of-sample

6 Conclusão