**CS559 Machine learning final project**

**Income salary predict application: based on Weka and JSP MVC**

Xing Gao

Data Set for Training and Testing

Adult Data set: From UCI machine learning repository:

Probability of label “>50K” and “<=50K”

Project Overview

The object of this project is using machine learning technology to generate a classifier which can based on user information,

eg.age,occupation etc. , **to predict** **if that user** **can earn salary greater 50k US dollar or not.**

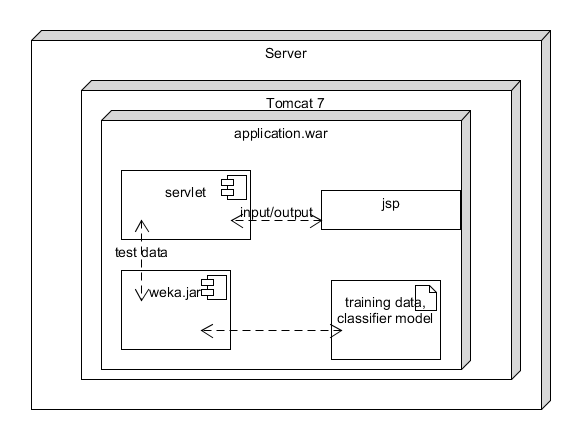
Using Weka to choose different classifier and corresponding optimal parameter.

* Naive Bayes,
* KNN
* C45
* Random forest.
* Bagging
* Boosting

Result page

* Classifier performance Demo(Correct Rate)
* User input one instance make prediction

Software Architecture



Data Description

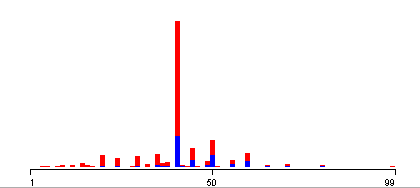
* **Attribute 1:** age: continuous.
* **Attribute 2:** workclass: Private, Self-emp-not-inc….
* **Attribute 3:** fnlwgt: continuous.
* **Attribute 4:** education: Bachelors, Some-college, 11th, …
* **Attribute 5:** education-num: continuous.
* **Attribute 6:** marital-status: Married-civ-spouse, Divorced, Never-married…
* **Attribute 7:** occupation: Tech-support, Craft-repair,
* **Attribute 8:** relationship: Wife, Own-child, Husband,
* **Attribute 9:** race: White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.
* **Attribute 10:** sex: Female, Male.
* **Attribute 11:** capital-gain: continuous.
* **Attribute 12:** capital-loss: continuous.
* **Attribute 13:** hours-per-week: continuous.
* **Attribute 14**: native-country: United-States, Cambodia, England..
* **Attribute 15(Class Label):** >50K, <=50K…

Remove Problematic Attribute

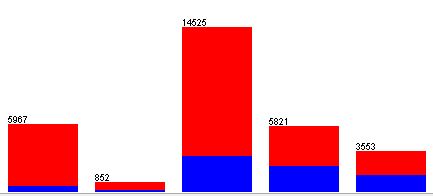
Discretize Data

Many attribute is continues numerical data, we put them in category

For example: Hours-Per-Week



After been discretize



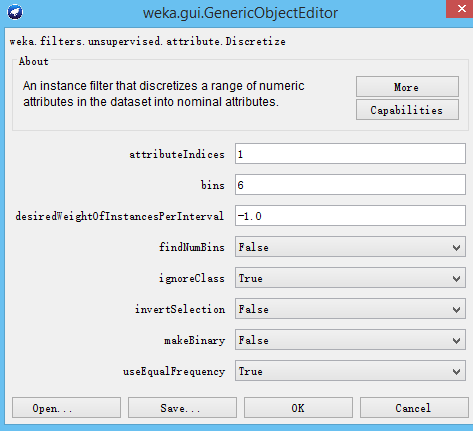
Attribute 5 and 4 are both representing education, we want keep relation between educations. eg.**Doctor > bachelor**.

**Remove attribute 4: education.**

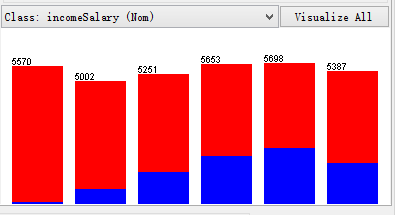
Attribute 3 is not relevant to income, also need been removed

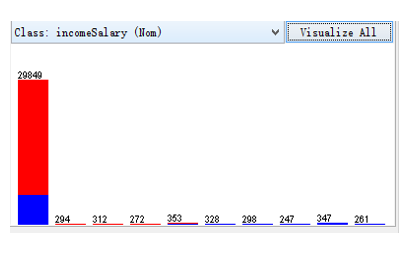
Discretize Data-Cont.

In weka, the way make numerical data is apply “weka.filters.unsupervised.attribute.Discretize”

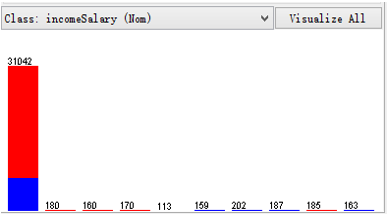


Attribute 1 age: 6 bins

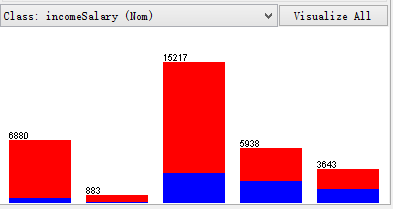




Attribute: Capital-gain: 10 Bins



Attribute: capital-loss: 10 Bins



Attribute: hours-per-week: 5 bins

Process Missing Data

For some reason, some sample missing data

* workclass has 1836 missing data,
* occupation: has 1843 missing data.
* native-country: has 583 missing data.

Two reason:

* data is missing when collection
* Don’t want provide by data provider’s intent.

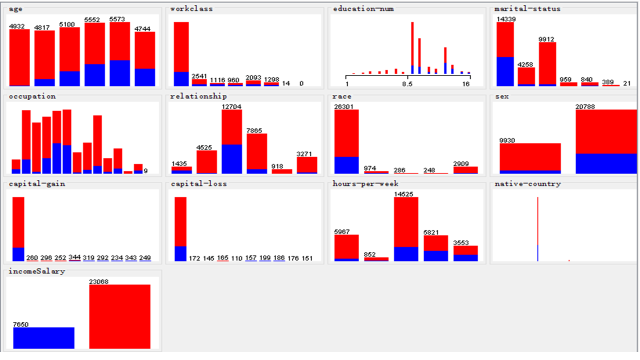
Two solutions:

Option 1: delete whole sample if some attribute data is missing.

Option 2: replace data with average of that attribute.

Solution to our project:

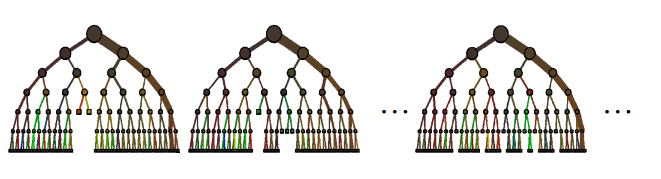
* Since **work class** and **occupation** has great influence on person’s salary, so if work class or occupation is missing, we **delete whole instance.**
* After remove all instance missing work class and occupation, we found out still have 556 instances missing native-country.
* In the rest 30718 instance, we have 27504 instances are from United-state, that 89.53%. So for people missing native country attribute, we will set them to United-state.



Data visualize

Advance Classifier-Random Forest

A forest is an ensemble of trees. The trees are all slightly different from one another



Experiment will test different number of trees.

Number of trees = 10 , 20, 30

When we choose 20 we got best result.

Simple Classifier-KNN-Cont.

-Ccc

We will test k = 1, 5, 9, 15, 21, 41, 51(only odd number)

Each classifier will test 10 times and get average error rate.

Based our best result. Choose K=51

Simple Classifier-Naive Bayes

In Naive Bayes, we assume that each of feature is continually independent of every other feature.

A.k.a. the "independent feature model”

p(ω1 |x1 , x 2 ,…)=αp(x1 | ω1 ) p(x2 | ω1)… p(ω1)

We compare posterior for each class, choose class ω which has biggest posterior.

Simple Classifier-KNN

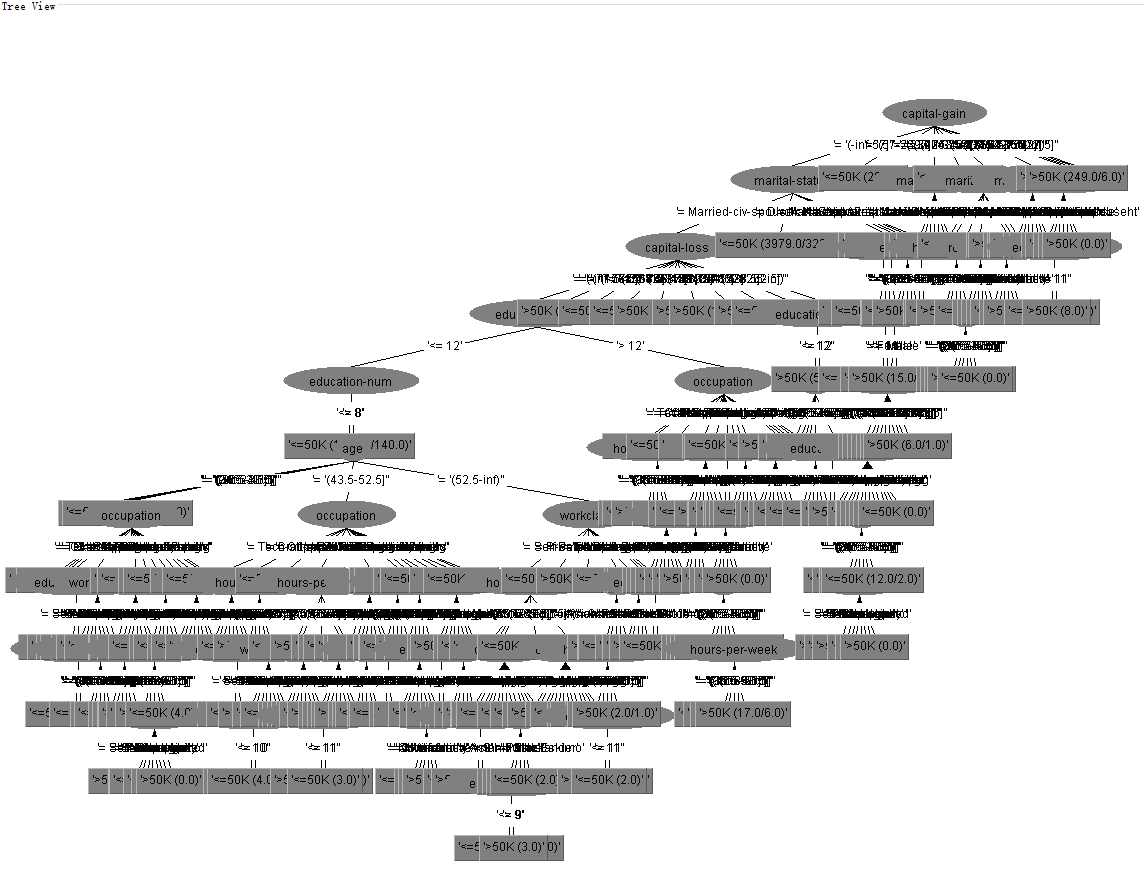
KNN: k-nearest neighbors algorithm

In *k-NN classification*, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its *k* nearest neighbors

If *k* = 1, then the object is simply assigned to the class of that single nearest neighbor.

Improving Result: Ensemble Method-Bagging

* Bootstrap aggregating, given a dataset S, sampling it to small size Si ( i is the number of classifier), since we have different training set, we will have different classifier, Ci.
* When we classifying a new instance, we run every classifier, and vote based on calls predication.
* Advantage:
* Improving result, when there has noise in Data set



Advance Classifier-C4.5

In Weka it’s call J48.

Basic idea of C4.5 is generate a decision tree based on data, the parameter we need test is size of the tree.

Conclusions

For sample belongs to “>50K”, Naive Bayes has best correct rate 80%, comparing average correct rate for other classifier only have 56-64%.

For Sample belongs to “<=50K”, C4.5 has best correct rate 93.352%, but naive bayes only have 82%.

Correct Rate table

Bagging

Bootstrap aggregating, given a dataset S, sampling it to small size Si ( i is the number of classifier), since we have different training set, we will have different classifier, Ci.

When we classifying a new instance, we run every classifier, and vote based on calls predication.

|  |  |  |  |
| --- | --- | --- | --- |
| Name of classifier | Overall correct rate | correct rate for “>50K” class | correct rate for “<=50K” class |
| KNN | 84.2773 % | 56.417% | 93.244% |
| Naive Bayes | 82.2266% | 80.548% | 82.766% |
| Random Forest | 84.668 % | 63.337% | 91.523% |
| C4.5 | 85.6445 % | 61.698% | 93.352% |
| Bagging | 86.0189 % | 62.834% | 93.481% |
| Boosting | 84.4076 % | 64.171% | 90.921% |

Real world implementation

**If we have cost table:**

If C1 is greater than C2, we better choose higher correct rate classifier among “>50K”, which is “Naive Bayes”

If C2 is greater than C1, we better choose higher correct rate classifier among “<=50K”, which is “C4.5”.

|  |  |  |
| --- | --- | --- |
| **Real class**  **Assumption class** | **>50K** | **<=50K** |
| **>50K** | 0 | C1 |
| **<=50K** | C2 | 0 |

**If we don’t have cost table:**

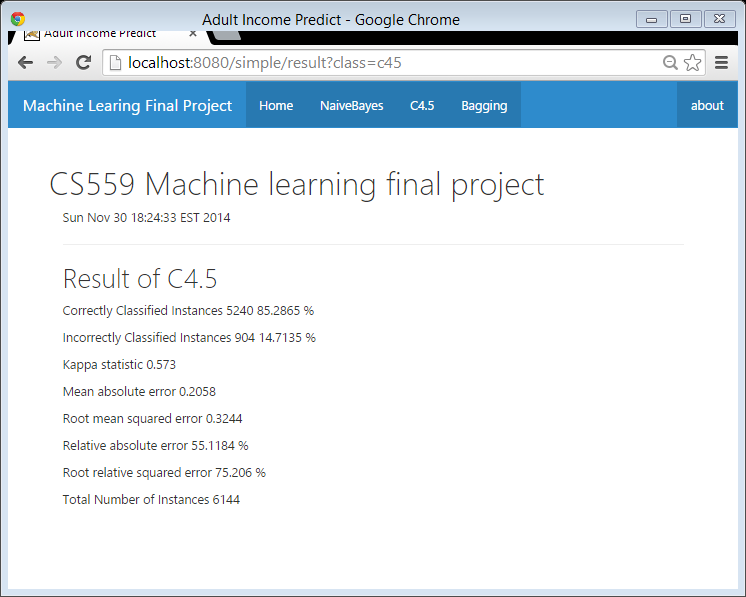
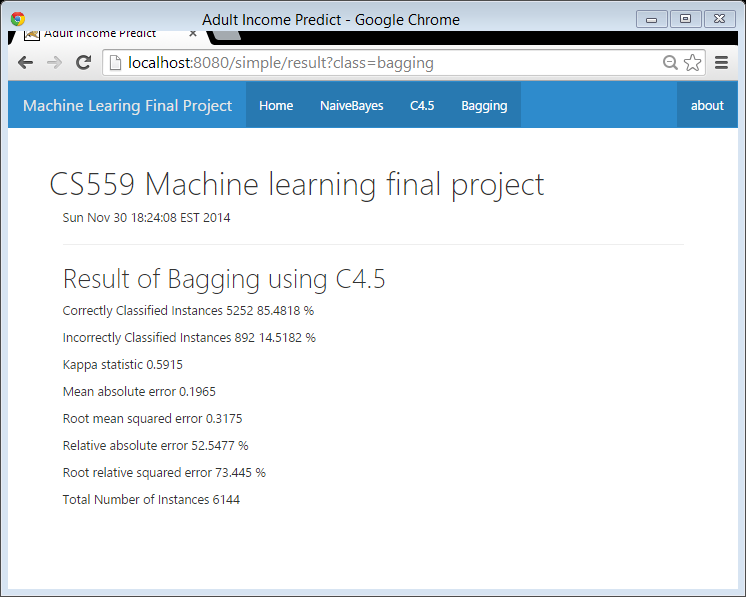
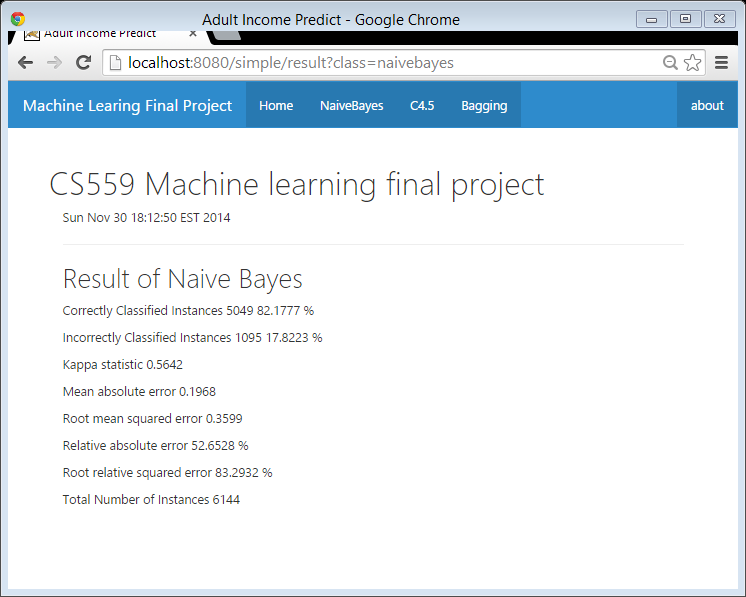
To get probability of person is “>50K”:

Probability of “>50K” calculated by Naive Bayes **X** correct rate of Naive Bayes in class “>50K”;

To get probability of person is “<=50K”:

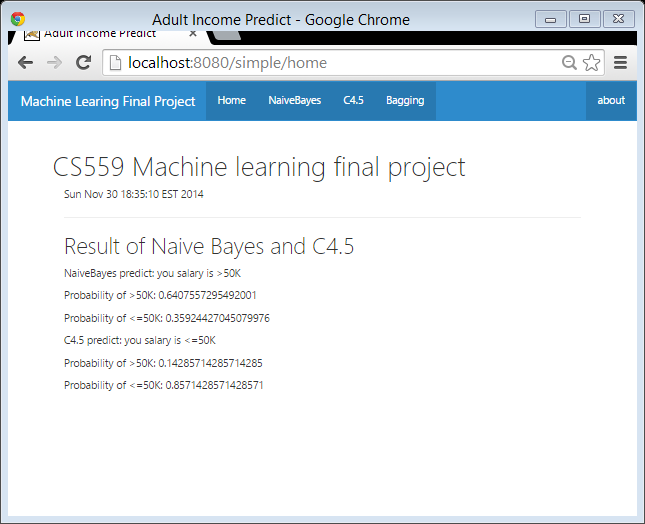
Probability of “<=50K” calculated by C4.5 **X** correct rate of C4.5 in class “<=50K”

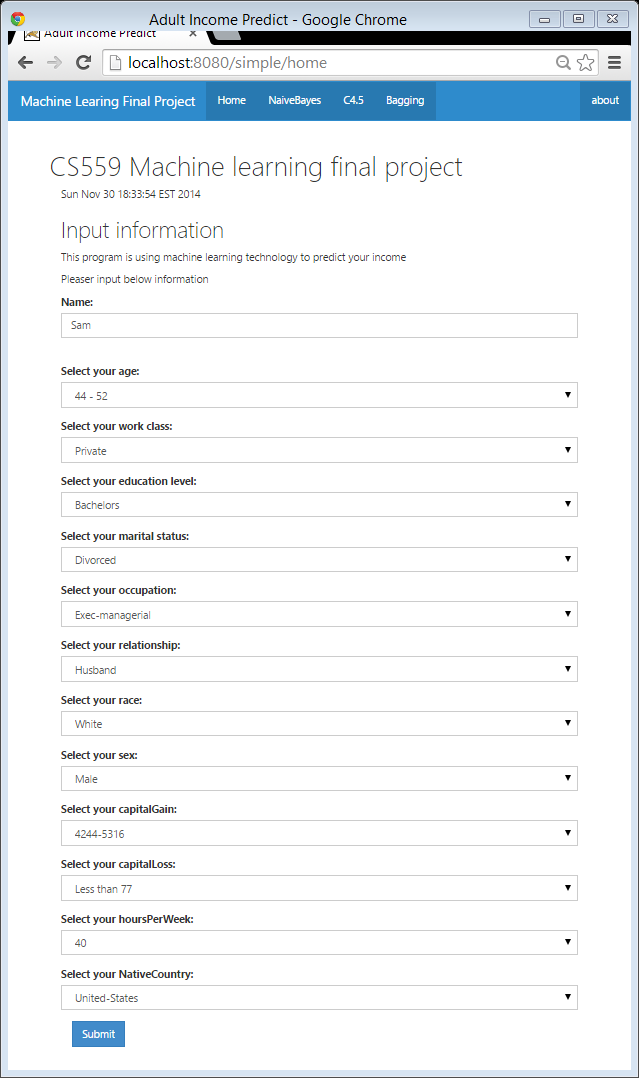
Compare two probabilities, choose the class which has better probability.



Single Instance Predict

This page let user input data for each attribute and getting prediction from Naive Bayes and C4.5





Classifier performance demo