INSY-5339-002- Principles of business data mining.

Project Report

House Sales Price Prediction

Group – 8

Amy Rettig

Chaitali Bonke

Swati kohli

Salman Mohammed

Tapan Patel

TABLE OF CONTENTS

Abstract

This report describes a model for predicting house prices. We have used historical data, data describing sale of individual residential properties in Ames, Iowa from 2006 to 2010. The final data set that was used to develop the model contains 1461 instances having 65 Attributes. Our class attribute was a continuous numeric variable which was bifurcated into 5 separate bins to make it compatible with classification algorithms. Taking into consideration an exhaustive list of explanatory variables describing (almost) every aspect of residential homes in Ames, Iowa, we aim analyze this data and develop a model that can predict the price range of a given house.

Introduction

All of us have or will face this question in our lives How much should I pay for this house? Or What is my house worth? It is a tough question to answer the buyer always wants to pay less and the seller always wants a high price. House prices are very sensitive to extraneous factors and thus they vary. Every house in its own sense is unique. Every house has its own pros and cons. There are many factors that affect the pricing of the house. To understand this let’s see an example - Which house has a higher value, a very large home located on a farm or a small-medium sized apartment located in Manhattan. Or how about this one, two apartments in the same building on the same floor in Manhattan. One is facing east having a view of the city and park and other is facing west a beautiful view of the pier and the sea. It is tough to decide the prices. To answer this question, we need to weigh the various factors and decide based on people’s preferences. So now the question is what factors should be considered while buying or selling a house. Generally, we think of factors such as Sq. feet area or How old or what is the quality of materials, what are the various amenities, how Accessible the property is or what locality. While these factors do affect the price but they are not the only ones so we here have tried to include as many factors as we can in our model to predict house price Taking into consideration 70 explanatory variables describing (almost) every aspect of residential homes in Ames, Iowa, we aim to analyze this data and predict the price range of a given house.

1. Data Background

1.1 Source of Dataset.

The Data-Set was obtained from “Kaggle”. The Ames Housing dataset was compiled by Dean De Cock for use in data science education. The dataset describes the sale of individual residential property in Ames, Iowa from 2006 to 2010.

1.2 Description of the dataset

The original data set contained 80 variables that are directly related to property sales, a brief description about the attributes is given later on in the document, in general the 80 variables describe the qualitative and quantitative factors that have varying effect on the final sales price of the property. Many of these variables contain very general information that a home buyer would focus on while assessing a property for e.g. When was it built? How big is the lot? How many feet of living space is in the dwelling? Is the basement finished? How many bathrooms are there? The data was originally used to create a regression model that could predict exact numeric value of the house.

1.3 Description of Attributes.

In general, the 20 continuous variables relate to various area dimensions for each observation. In addition to the typical lot size and total dwelling square footage found on most common home listings, other more specific variables are quantified in the data set. Area measurements on the basement, main living area, and even porches are broken down into individual categories based on quality and type. The large number of continuous variables in this data set should give students many opportunities to differentiate themselves as they consider various methods of using and combining the variables.

The 14 discrete variables typically quantify the number of items occurring within the house. Most are specifically focused on the number of kitchens, bedrooms, and bathrooms (full and half) located in the basement and above grade (ground) living areas of the home. Additionally, the garage capacity and construction/remodeling dates are also recorded.

There are a large number of categorical variables (23 nominal, 23 ordinal) associated with this data set. They range from 2 to 28 classes with the smallest being *STREET* (gravel or paved) and the largest being *NEIGHBORHOOD* (areas within the Ames city limits). The nominal variables typically identify various types of dwellings, garages, materials, and environmental conditions while the ordinal variables typically rate various items within the property. The coding within the original data typically utilized an eight-character name that was relevant to the classification but some of the original class levels were difficult to interpret. For ease of use many class levels were recoded into slightly more usable forms (see the documentation file

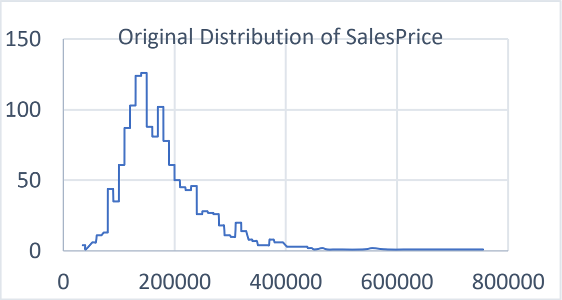
The data has 82 columns which include 23 nominal, 23 ordinal, 14 discrete, and 20 continuous variables (and 2 additional observation identifiers).

1) Order (Discrete): Observation number.
2) PID (Nominal): Parcel identification number - can be used with city web site for parcel review.
3) MS SubClass (Nominal): Identifies the type of dwelling involved in the sale.
4) MS Zoning (Nominal): Identifies the general zoning classification of the sale.
5) Lot Frontage (Continuous): Linear feet of street connected to property
6) Lot Area (Continuous): Lot size in square feet
7) Street (Nominal): Type of road access to property
8) Alley (Nominal): Type of alley access to property
9) Lot Shape (Ordinal): General shape of property
10) Land Contour (Nominal): Flatness of the property
11) Utilities (Ordinal): Type of utilities available
12) Lot Config (Nominal): Lot configuration
13) Land Slope (Ordinal): Slope of property
14) Neighborhood (Nominal): Physical locations within Ames city limits (map available)
15) Condition 1 (Nominal): Proximity to various conditions
16) Condition 2 (Nominal): Proximity to various conditions (if more than one is present)
17) Bldg Type (Nominal): Type of dwelling
18) House Style (Nominal): Style of dwelling
19) Overall Qual (Ordinal): Rates the overall material and finish of the house
20) Overall Cond (Ordinal): Rates the overall condition of the house
21) Year Built (Discrete): Original construction date
22) Year Remod/Add (Discrete): Remodel date (same as construction date if no remodeling or additions)
23) Roof Style (Nominal): Type of roof
24) Roof Matl (Nominal): Roof material
25) Exterior 1 (Nominal): Exterior covering on house
26) Exterior 2 (Nominal): Exterior covering on house (if more than one material)
27) Mas Vnr Type (Nominal): Masonry veneer type
28) Mas Vnr Area (Continuous): Masonry veneer area in square feet
29) ExterQual (Ordinal): Evaluates the quality of the material on the exterior
30) Exter Cond (Ordinal): Evaluates the present condition of the material on the exterior
31) Foundation (Nominal): Type of foundation
32) BsmtQual (Ordinal): Evaluates the height of the basement
33) Bsmt Cond (Ordinal): Evaluates the general condition of the basement
34) Bsmt Exposure (Ordinal): Refers to walkout or garden level walls
35) BsmtFin Type 1(Ordinal): Rating of basement finished area
36) BsmtFin SF 1 (Continuous): Type 1 finished square feet
37) BsmtFinType 2 (Ordinal): Rating of basement finished area (if multiple types)
38) BsmtFin SF 2 (Continuous): Type 2 finished square feet
39) BsmtUnf SF (Continuous): Unfinished square feet of basement area
40) Total Bsmt SF (Continuous): Total square feet of basement area
41) Heating (Nominal): Type of heating
42) HeatingQC (Ordinal): Heating quality and condition

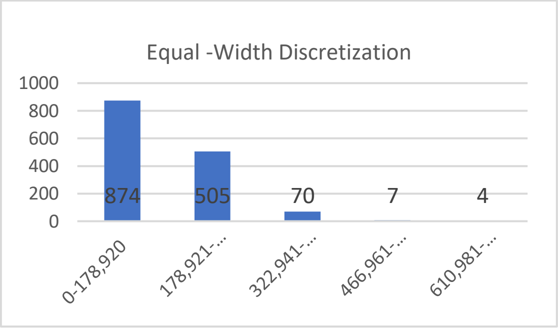
43) Central Air (Nominal): Central air conditioning
44) Electrical (Ordinal): Electrical system
45) 1st Flr SF (Continuous): First Floor square feet
46) 2nd Flr SF (Continuous): Second floor square feet
47) Low Qual Fin SF (Continuous): Low quality finished square feet (all floors)
48) Gr Liv Area (Continuous): Above grade (ground) living area square feet
49) Bsmt Full Bath (Discrete): Basement full bathrooms
50) Bsmt Half Bath (Discrete): Basement half bathrooms
51) Full Bath (Discrete): Full bathrooms above grade
52) Half Bath (Discrete): Half baths above grade
53) Bedroom (Discrete): Bedrooms above grade (does NOT include basement bedrooms)
54) Kitchen (Discrete): Kitchens above grade
55) KitchenQual (Ordinal): Kitchen quality
56) TotRmsAbvGrd (Discrete): Total rooms above grade (does not include bathrooms)
57) Functional (Ordinal): Home functionality (Assume typical unless deductions are warranted)
58) Fireplaces (Discrete): Number of fireplaces
59) FireplaceQu (Ordinal): Fireplace quality
60) Garage Type (Nominal): Garage location
61) Garage YrBlt (Discrete): Year garage was built
62) Garage Finish (Ordinal): Interior finish of the garage
63) Garage Cars (Discrete): Size of garage in car capacity
64) Garage Area (Continuous): Size of garage in square feet
65) Garage Qual (Ordinal): Garage quality
66) Garage Cond (Ordinal): Garage condition
67) Paved Drive (Ordinal): Paved driveway
68) Wood Deck SF (Continuous): Wood deck area in square feet
69) Open Porch SF (Continuous): Open porch area in square feet
70) Enclosed Porch (Continuous): Enclosed porch area in square feet
71) 3-Ssn Porch (Continuous): Three season porch area in square feet
72) Screen Porch (Continuous): Screen porch area in square feet
73) Pool Area (Continuous): Pool area in square feet
74) Pool QC (Ordinal): Pool quality
75) Fence (Ordinal): Fence quality
76) Misc Feature (Nominal): Miscellaneous feature not covered in other categories
77) Misc Val (Continuous): $Value of miscellaneous feature
78) Mo Sold (Discrete): Month Sold (MM)
79) Yr Sold (Discrete): Year Sold (YYYY)
80) Sale Type (Nominal): Type of sale
81) Sale Condition (Nominal): Condition of sale
82) SalePrice (Continuous): Sale price $$

1.4 Class Attribute

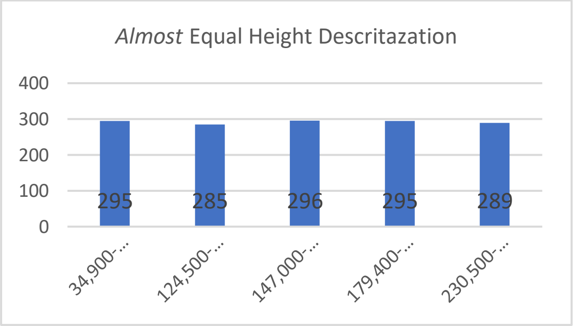
Our class attribute, SalePrice, represents the final sales price of homes sold in Ames, Iowa between 2006 and 2010. In the original dataset this was presented as a continuous numeric variable ranging from $34,900 to $755,000 which was not compatible with classification algorithms. In order to approach the prediction of final sale price as a classification problem we needed to discretize our class variable into bins of either equal width or equal height.



The highly skewed of housing prices immediately presented a problem in using equal width discretization. The range of sale prices in the original dataset divided into equal widths would yield five bins covering ranges of $144020. Of the 1460 records, 874 (59.9%) were in the lowest range and only 4 (0.2%) in the highest range. While this would give even the simplest algorithm (ZeroR) a 60% accuracy rate, any model would be left with little basis for classification in the higher ranges.



We chose to use equal height discretization which proved to be much better suited to our data. Precisely equal distribution was prevented by clumping, but 1460 records were sorted into five bins with an average of 292 records each.



2. DATA CLEANING PROCESS

2.1 MISSING VALUES

Many of the attributes in our dataset describe value adding features of house such as fireplaces or garage. Because houses can have many combinations of features, not every attribute applied to every record. Houses with fewer features tend to be in the lower price range, meaning the absence of feature can generally be as strong a predictor of sales price as its description.

Attribute describing features not applicable to a record were marked “NA” in the original dataset which presented a problem in the numeric attributes MasVnrArea, LotFrontage and GarageYrBlt where it forced Weka to treat the attributes as nominal. We addressed this issue by replacing “NA” with “?” when it was found in these attributes so that they would be interpreted as missing values in a numeric attribute without interfering with the interpretation of values in other instances.

2.2 DERIVED ATTRIBUTES

By looking over the data collected in the original dataset, it becomes very clear that square footage is expected to be an important factor in the sale price of a house. 15 of the 80 attributes describe the size of various parts of the house, such as the second floor, basement and garage, in square feet. However, while the total living area is featured prominently in any description of a house for sale, that information is conspicuously absent in the data. To address this, attributes describing the square footage of the basement and the above ground living area were added to derive the total living area of the house.

The original data described one feature, the porch of the house, using four attributes representing the four possible styles and containing the square footage as a numeric value. Because nearly all houses have, at most one style porch and many lack that feature entirely, over 80% of the values contained in these four attributes are “0”. We chose to condense these into two attributes, one as a nominal value describing the type of porch and the other a numeric value for its size.

2.3 IRRELEVANT ATTRIBUTES

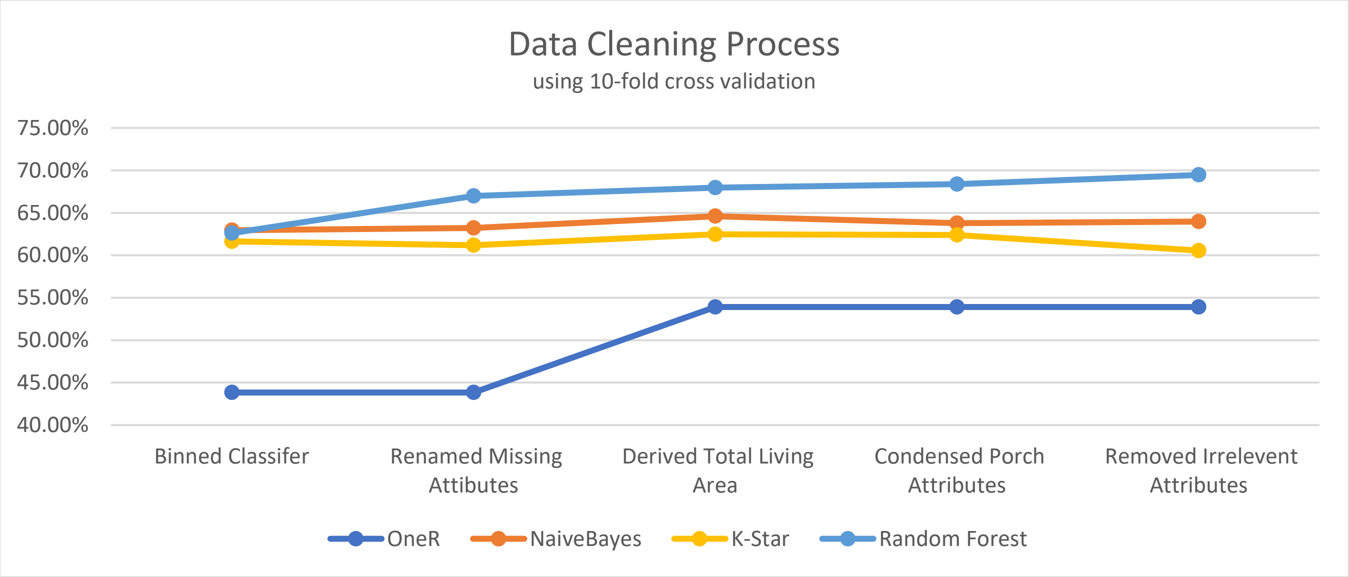
The original dataset contained many attributes that were dominated by a single value. These are listed in the following table. Given the lack of variation in these attributes, the weak predictive influence that they might contribute was outweighed by the effect of the complexity they added to the models. These attributes were deleted from our data.

Attribute	Dominant Value
PoolQC	1453 values = NA
Pool Area	1453 values = 0
LowQualFinSF	1434 values = 0
MiscVal	1408 values = 0
MiscFeature	1406 values = NA
BsmtHalfBath	1378 values = 0
Alley	1369 values = 0
BsmtFinSF2	1293 values = 0
Condition2	1445 values = Norm
Utilities	1459 values = AllPub
RoofMatl	1434 values = CompShg
Street	1454 values = Pave
LandSlope	1382 values = Gtl
Funtional	1360 values = Typ

Additionally, original dataset assigned each record an ID number as a primary key. These values had no relationship with the record they were assigned to and so were deleted from our data.

2.4 EFFECTS OF DATA CLEANING

Each step in the data cleaning process was tested against five algorithms using 10-fold cross validation. The performance of OneR improved most noticeably with the addition of the total living area implying that the derived is a stronger predictor that the attributes it was derived from. Other algorithms show small but consistent improvements over the process. The average performance across all algorithms increased from 57.96% to 61.89%.



3 EXPERIMENT DESIGN

3.1 CLASSIFIER SELECTION:

Once the task is decided and goals are codified, a concrete method (or set of methods) needs to be chosen for searching patterns in the data. Depending on the choice of techniques, parameter optimization may or may not be required. The selection of classifiers is based on two main reasons:

i) High prediction accuracy

ii) Low variance and high stability of the model with prediction accuracy being high or reasonable.

We selected below classification models for our experiment design:

1. **One-R**: We have used One-R as our benchmark for this project i.e. the other classifiers should have accuracy greater than that of One-R.It learns a one-level decision tree, i.e. generates a set of rules that test one particular attribute. Basic version (**assuming nominal attributes**):

   - One branch for each of the attribute’s values
   - Each branch assigns most frequent class
   - Error rate: proportion of instances that don't belong to the majority class of their corresponding branch
   - Choose attribute with lowest error rate.

2. **Naïve Bayes (bayes):** The Naive Bayes algorithm is a classification algorithm based on Bayes rule and a set of conditional independence assumptions. For some types of probability models, Naïve Bayes classifiers can be trained very efficiently in a supervised learning setting. An advantage of Naïve Bayes is that it only requires a small number of training data to estimate the parameters necessary for classification. Naive Bayes learners and classifiers can be extremely fast compared to more sophisticated methods.

3. **K-Star:** K* is an instance-based classifier, that is the class of a test instance is based upon the class of those training instances similar to it, as determined by some similarity function. It differs from other instance-based learners in that it uses an entropy-based distance function. The use of entropy as a distance measure has several benefits. Amongst other things it provides a consistent approach to handling of symbolic attributes, real valued attributes and missing values. It is a lazy learning method i.e. generalization beyond the training data is delayed until a query is made to the system, as opposed to in eager learning, where the system tries to generalize the training data before receiving queries.

4. **Random Forest:** Random forests or random decision forests[1][2] are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the

classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set.

3.2 ATTRIBUTE SELECTION

The attribute selection process is a part of the experiment design as one of the two factors in our experiment design. For our experiment design we have used two datasets. The datasets are the Full Set and the Top 20 Set. We used Weka's OneRAttributeEval to determine the Top 20 Attributes.

OneRAttributeEval : Evaluates the worth of an attribute by using the OneR classifier.

The table below describes the options available for OneRAttributeEval.

Option	Description
evalUsingTrainingData	Use the training data to evaluate attributes rather than cross validation.
Folds	Set the number of folds for cross validation.
minimumBucketSize	The minimum number of objects in a bucket (passed to OneR).
Seed	Set the seed for use in cross validation.

The table below describes the capabilites of OneRAttributeEval.

Capability	Supported
Class	Binary class, Missing class values, Nominal class
Attributes	Date attributes, Nominal attributes, Empty nominal attributes, Missing values, Unary attributes, Numeric attributes, Binary attributes
Min # of instances	1

We have below attributes (Top 20) with their ranking:

average merit			average rank				Attribute
53.052	+-	0.915	1.2	+-	0.4	39	totalLivingArea
52.534	+-	0.321	1.8	+-	0.4	12	OverallQual
47.633	+-	0.587	3	+-	0	8	Neighborhood
44.193	+-	0.679	4.2	+-	0.4	53	GarageArea
43.364	+-	0.298	5	+-	0.45	52	GarageCars
42.07	+-	1.084	6.1	+-	0.83	38	GrLivArea
40.982	+-	0.725	7.1	+-	0.7	14	YearBuilt
40.046	+-	0.881	8.7	+-	1.49	31	TotalBsmtSF
39.711	+-	0.599	8.9	+-	1.14	24	BsmtQual
38.63	+-	0.323	10.8	+-	0.75	45	KitchenQual
38.531	+-	1.267	11.2	+-	2.44	50	GarageYrBlt
38.683	+-	0.653	11.4	+-	1.2	15	YearRemodAdd
38.356	+-	0.337	12.2	+-	1.25	51	GarageFinish
37.565	+-	0.752	14.2	+-	1.66	46	TotRmsAbvGrd
36.986	+-	0.156	15	+-	0.77	21	ExterQual
36.233	+-	0.138	16.9	+-	0.83	41	FullBath
36.043	+-	0.647	17.6	+-	2.37	48	FireplaceQu
36.096	+-	1.113	17.8	+-	2.79	36	1stFlrSF
35.746	+-	0.255	18.9	+-	1.37	49	GarageType
35.647	+-	0.268	19.4	+-	0.92	23	Foundation
34.597	+-	0.178	22.2	+-	1.33	47	Fireplaces
34.635	+-	0.813	22.5	+-	2.06	3	LotFrontage
34.581	+-	0.934	22.6	+-	3.1	1	MSSubClass
34.087	+-	0.825	23.8	+-	2.23	28	BsmtFinSF1
33.775	+-	0.713	25.2	+-	1.72	37	2ndFlrSF
33.531	+-	0.694	25.3	+-	1.68	27	BsmtFinType1
33.447	+-	0.571	25.5	+-	1.5	33	HeatingQC
32.131	+-	0.936	28.5	+-	1.91	4	LotArea
31.918	+-	0.498	29.1	+-	0.7	17	Exterior1st
31.476	+-	0.492	30.5	+-	1.02	20	MasVnrArea
31.385	+-	0.72	30.9	+-	1.37	18	Exterior2nd
31.073	+-	0.869	31.6	+-	1.85	58	TypeOfPorch
30.769	+-	0.766	32.8	+-	1.99	57	WoodDeckSF
30.236	+-	0.591	33.8	+-	0.6	19	MasVnrType
29.323	+-	0.457	35.5	+-	1.5	13	OverallCond
29.163	+-	0.878	36.7	+-	1.95	30	BsmtUnfSF
28.919	+-	0.59	37.3	+-	2.28	43	BedroomAbvGr

28.364	+-	0.416	39.1	+-	1.64	2	MSZoning
28.067	+-	0.659	39.8	+-	2.52	26	BsmtExposure
28.014	+-	0.545	40.4	+-	1.85	42	HalfBath
27.915	+-	1.134	40.8	+-	3.37	59	Porchsqft
27.915	+-	0.476	41.2	+-	1.83	5	LotShape
27.61	+-	0.859	41.7	+-	2.37	11	HouseStyle
27.443	+-	0.332	42.2	+-	1.78	40	BsmtFullBath
26.629	+-	0.269	44.7	+-	0.46	54	GarageQual
26.012	+-	0.206	46.1	+-	0.54	55	GarageCond
25.502	+-	0.451	47.3	+-	0.9	64	SaleCondition
25.114	+-	0.475	48.4	+-	1.43	16	RoofStyle
24.505	+-	0.321	50.9	+-	1.3	34	CentralAir
24.406	+-	0.275	51.1	+-	1.76	35	Electrical
24.216	+-	0.412	51.9	+-	2.43	63	SaleType
24.247	+-	0.574	51.9	+-	2.51	60	Fence
24.033	+-	0.518	52.7	+-	1.85	25	BsmtCond
24.041	+-	1.014	53	+-	4.12	61	MoSold
24.087	+-	0.244	53.2	+-	1.4	56	PavedDrive
23.615	+-	0.408	55.1	+-	1.37	10	BldgType
22.907	+-	0.264	57.6	+-	1.28	6	LandContour
22.686	+-	0.555	58.1	+-	1.04	9	Condition1
22.481	+-	0.535	58.3	+-	1.62	29	BsmtFinType2
22.253	+-	0.52	59.5	+-	1.02	22	ExterCond
21.263	+-	0.646	61.8	+-	1.08	7	LotConfig
21.355	+-	0.119	61.9	+-	0.83	32	Heating
21.309	+-	0.449	62.1	+-	0.83	44	KitchenAbvGr
18.28	+-	0.716	64	+-	0	62	YrSold

3.3 FOUR CELL EXPERIMENT DESIGN

Two Factor Design: Our experiment design contained of two factors:

i) **Factor 1 (F1): Attribute selection**
   1) Full Dataset: All attributes (dataset after cleaning)
   2) Top 20 attributes

ii) **Factor 2 (F2): Percentage Split**
   1) 80% / 20% Percentage Split
   2) 30% / 70% Percentage Split

Four Criteria of the Design: The two factors are to be divided up into 4 criteria by keeping one factor constant and varying the other factor between two values and vice versa.

The conditions can be summarized as follows:

C1: 80%/20% split on the Full Set.

C2: 80%/20% split on the Top 20 Set.

C3: 30%/70% split on the Full Set.

C4: 30%/70% split on the Top 20 Set.

This is illustrated more clearly in the table below.

	Full Dataset	**Top 20 Attributes**
80% / 20% Percentage Split	C1	C2
30% / 70% Percentage Split	C3	C4

We applied the classifiers on the data set for all 4 conditions. In order to make our training and test data truly representative, as the data might lose its properties due to sampling and while running the classifiers we are doing ten runs for each criteria, each classifier with a distinct increasing seed value.

4 EXPERIMENTAL RESULTS

4.1 Results For each classifier

4.1.1 One-R

One-R				
Seed	c1	c2	c3	c4
1	53.42%	53.42%	51.57%	51.57%
3	50.68%	50.68%	50.29%	50.29%
5	48.97%	48.97%	52.94%	52.94%
7	55.14%	55.14%	53.62%	53.62%
9	54.79%	54.79%	50%	50%
11	50.34%	50.34%	50.29%	50.29%
13	57.19%	57.19%	54.11%	54.11%
15	48.63%	48.63%	52.64%	52.64%
17	54.11%	54.11%	52.45%	52.45%
19	57.88%	57.88%	53.42%	53.42%
AVE	53.12%	53.12%	52.13%	52.13%
STdDev	3.30%	3.30%	1.51%	4.27%
Variance	0.11%	0.11%	0.02%	0.18%

4.1.2 Naïve Bayes

Naïve Bayes				
Seed	c1	c2	c3	c4
1	67.47%	65.41%	64.78%	63.99%
3	61.99%	58.56%	65.46%	64.97%
5	63.70%	59.59%	64.58%	65.17%
7	63.70%	61.99%	59.98%	58.22%
9	70.55%	64.04%	64%	63%
11	65.07%	59.93%	67.81%	64.38%
13	66.10%	58.90%	61.06%	60.47%
15	64.73%	59.93%	63.80%	60.67%
17	64.38%	65.75%	64.19%	63.89%
19	63.36%	61.64%	65.07%	62.82%
AVE	65.10%	61.58%	64.12%	62.77%
STdDev	2.44%	2.67%	2.20%	4.47%
Variance	0.06%	0.07%	0.05%	0.20%

4.1.3 K-star

K-star				
Seed	c1	c2	c3	c4
1	62.33%	66.78%	60.67%	59.00%
3	57.19%	58.90%	61.45%	59.39%
5	62.67%	59.93%	58.12%	60.27%
7	64.04%	62.67%	58.12%	60.27%
9	60.27%	61.64%	59%	59%
11	61.99%	64.38%	56.65%	59.88%
13	65.07%	66.44%	58.61%	59.59%
15	60.96%	66.78%	59.30%	60.27%
17	55.82%	60.62%	58.22%	57.34%
19	63.01%	64.38%	60.96%	59.88%
AVE	61.34%	63.25%	59.07%	59.45%
STdDev	2.91%	2.93%	1.51%	2.36%
Variance	0.08%	0.09%	0.02%	0.06%

4.1.4 Random Forest

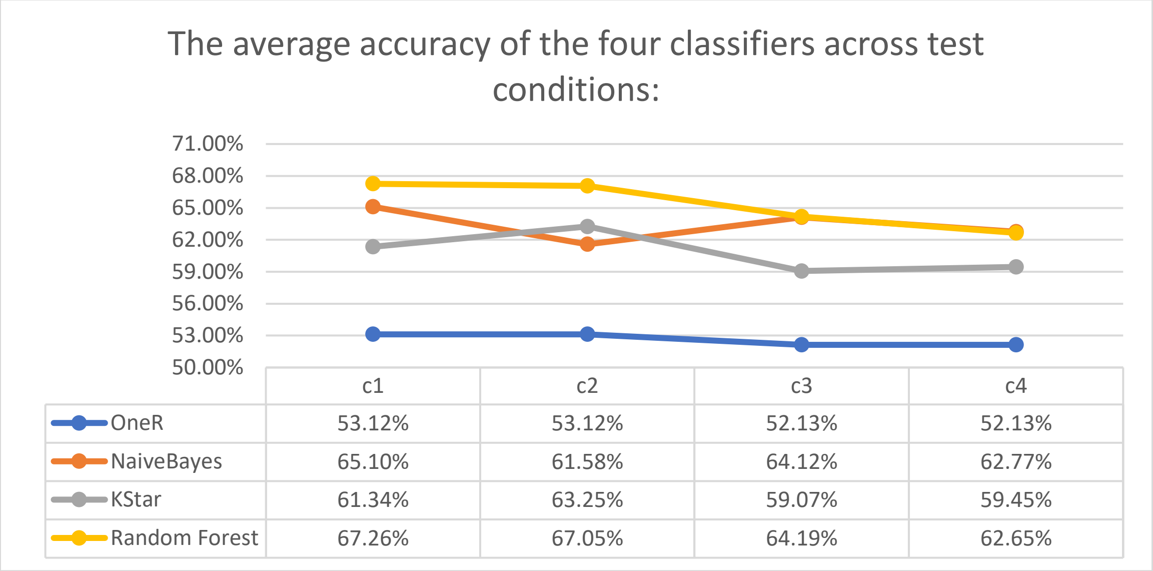
Random Forest				
Seed	c1	c2	c3	c4
1	69.18%	70.89%	63.80%	62.62%
3	66.44%	65.41%	65.07%	61.35%
5	69.18%	66.78%	62.92%	64.29%
7	69.52%	65.41%	60.27%	60.47%
9	69.52%	67.81%	65%	65%
11	66.78%	68.84%	63.70%	63.50%
13	68.15%	66.44%	64.19%	62.62%
15	61.30%	64.38%	67.22%	63.01%
17	64.04%	65.41%	63.99%	60.47%
19	68.49%	69.18%	65.26%	63.11%
AVE	67.26%	67.05%	64.19%	62.65%
STdDev	2.72%	2.07%	1.83%	1.52%
Variance	0.07%	0.04%	0.03%	0.02%

Total number of experiment runs = # of Criteria * # of Classifiers * # of Executions

$$4 * 4 * 10 = 160\ Runs$$

4.2 Summary of Results

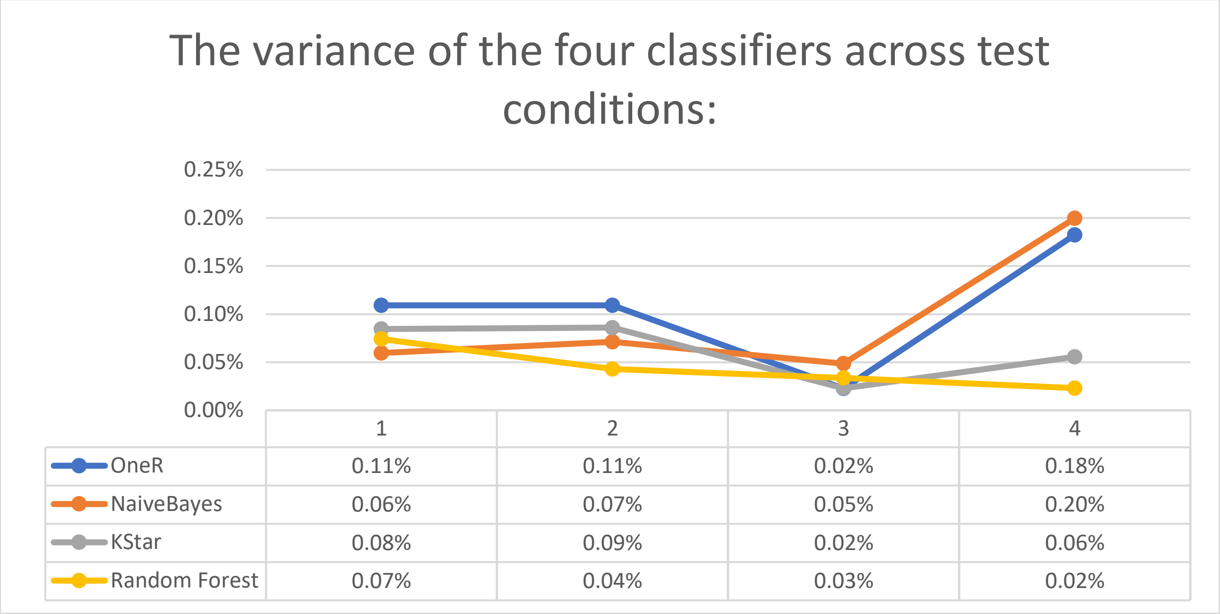
<u>The averages of the accuracy tested for the 3 Classifiers is as shown below:</u>



	c1	c2	c3	c4
OneR	53.12%	53.12%	52.13%	52.13%
NaiveBayes	65.10%	61.58%	64.12%	62.77%
KStar	61.34%	63.25%	59.07%	59.45%
Random Forest	67.26%	67.05%	64.19%	62.65%

The plot above gives the accuracy results where the 80%/20% splits (C1 & C2) are doing better than the 30%/70% splits (C3&C4). This was observed because C1 and C2 are using 80% to train and 20% to test and C3 & C4 are using 30% to train and 70% to test. When all the attributes (C1&C3) are being used to train and test, the results are better as compared to the selected attributes (C2&C4). Random Forest has the best accuracy under all conditions as compared to the other tests. One-R has the worst accuracy compared to the other three, followed by the K-Star which is a lazy learning algorithm, that does no learning in the training test and uses a distance metric in the test set. Naive Bayes is a Bayesian classifier that uses a probabilistic approach for classification. Random Forests are an algorithm of the Decision Trees and they correct the decision trees problem of overfitting to their training set, so its accuracy is the best in this case.

The Variance tested from the three classifier are shown by the chart below :

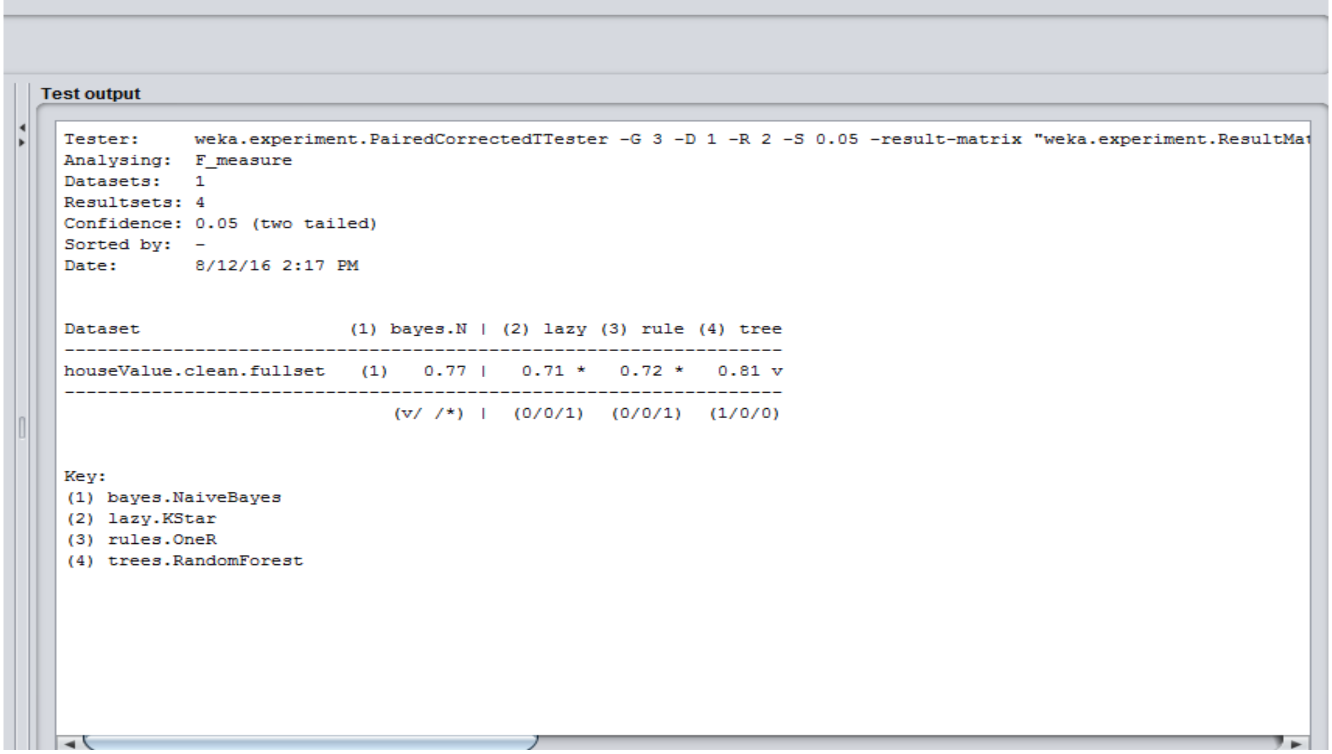


	1	2	3	4
OneR	0.11%	0.11%	0.02%	0.18%
NaiveBayes	0.06%	0.07%	0.05%	0.20%
KStar	0.08%	0.09%	0.02%	0.06%
Random Forest	0.07%	0.04%	0.03%	0.02%

From the above plot, OneR has the highest variance and this indicates that the OneR classifier is the least stable of the other classifiers and its Variance increases from C1 to C4. Naïve Bayes has the least Variance that indicates that it is the most stable. Random Forest has a low variance compared to K-star. K-star has slightly higher Variance and it varies drastically from C2 to C3 and from C3 to C4.

5 Analysis and Conclusion

5.1 Models Comparison



```
Tester:     weka.experiment.PairedCorrectedTTester -G 3 -D 1 -R 2 -S 0.05 -result-matrix "weka.experiment.ResultMat
Analysing:  F_measure
Datasets:   1
Resultsets: 4
Confidence: 0.05 (two tailed)
Sorted by:  -
Date:       8/12/16 2:17 PM


Dataset                    (1) bayes.N | (2) lazy (3) rule (4) tree
-------------------------------------------------------------------
houseValue.clean.fullset   (1)   0.77 |   0.71 *   0.72 *   0.81 v
-------------------------------------------------------------------
                                (v/ /*) |  (0/0/1)  (0/0/1)  (1/0/0)


Key:
(1) bayes.NaiveBayes
(2) lazy.KStar
(3) rules.OneR
(4) trees.RandomForest
```

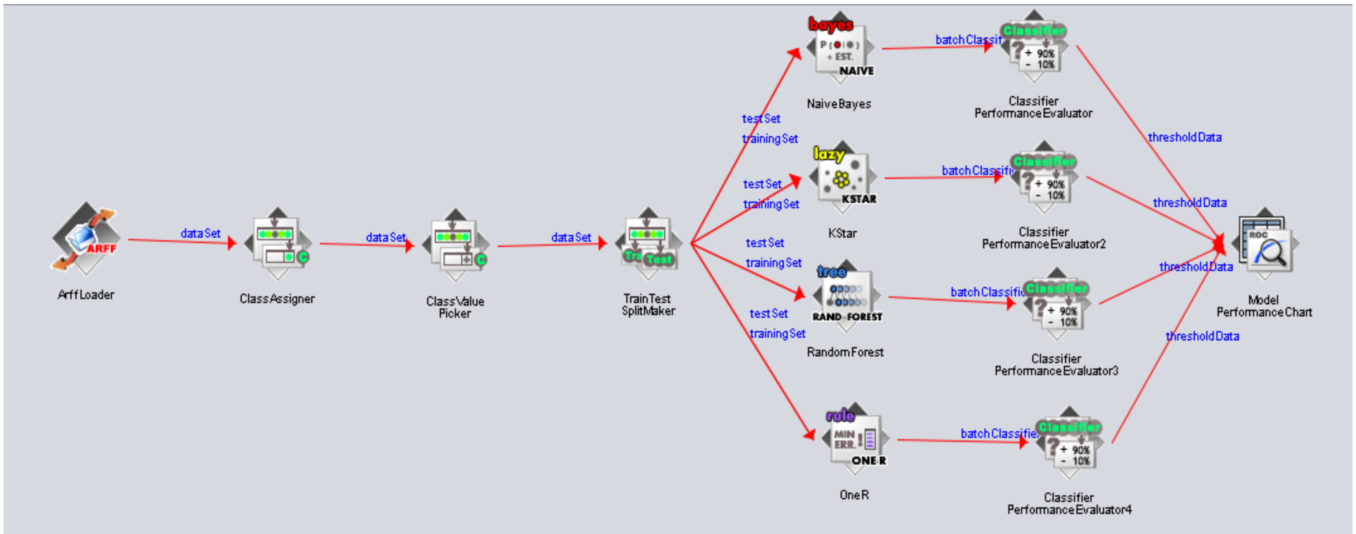
The ‘Experimenter’ feature in Weka is used to determine the performance of the algorithms chosen. The F-Scores of each of the 4 classifiers has been displayed. It is used to compare the performances of the algorithms with respect to the Naïve Bayes classifier algorithm. Observing the above results, the Random Forest performed the best out of the 4 classifiers.

5.2 ROC CURVE

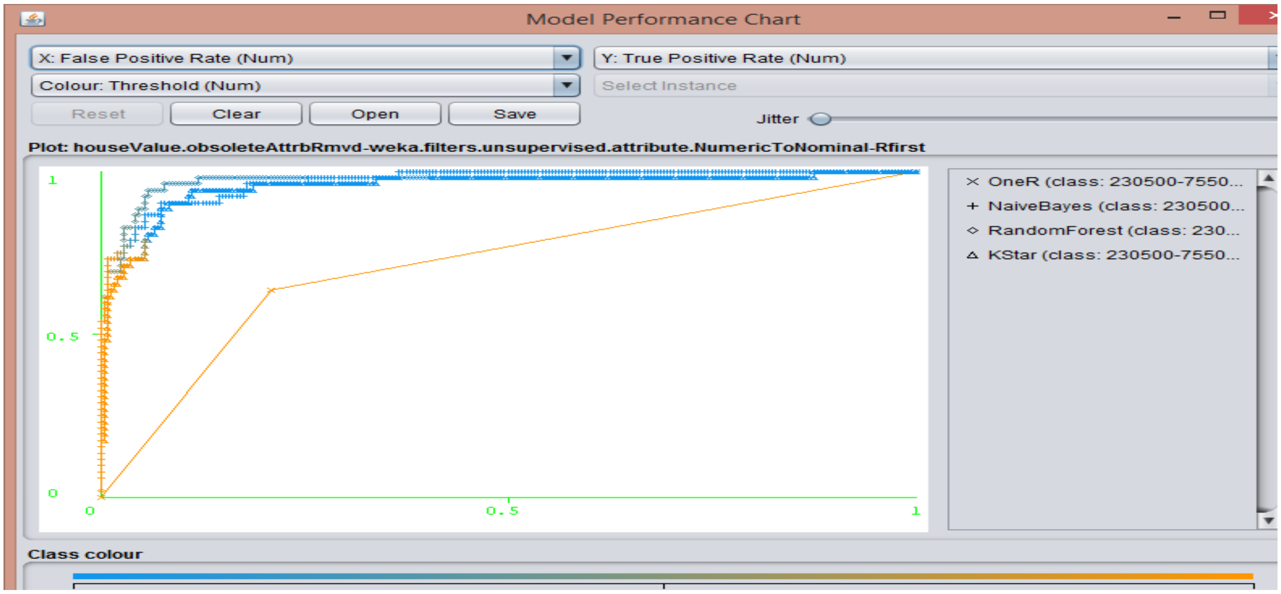
- **Definition**: The Receiver Operating Characteristic (ROC) is a graphical plot that explains the performance of a binary classifier system as its discrimination threshold is varied. It is generated by plotting the fraction of true positives out of the total actual positives (TPR = true positive rate) vs. the fraction of false positives out of the total actual negatives (FPR = false positive rate), at different threshold values.

  The ROC curve plots the accuracy of a classifier to predict the TPR (True Positive Rate) and FPR (False Positive Rate) on a curve. This results in finding out the accuracy with which our classifiers can predict the true positives and true negatives. This method of determining the classifier and factor overall efficiency is by 'how much area is covered under the ROC curve. Higher the area under the curve, better the model.

- **Generation of Multiple ROC Curves**: We designed a model using the 'Knowledge Flow' feature in Weka to plot multiple ROC curves for one factor comparison with another. The figure below shows the flow and design using the knowledge flow feature in Weka.



5.2.1 ROC Curve for the Full Set



Model Performance Chart

X: False Positive Rate (Num)

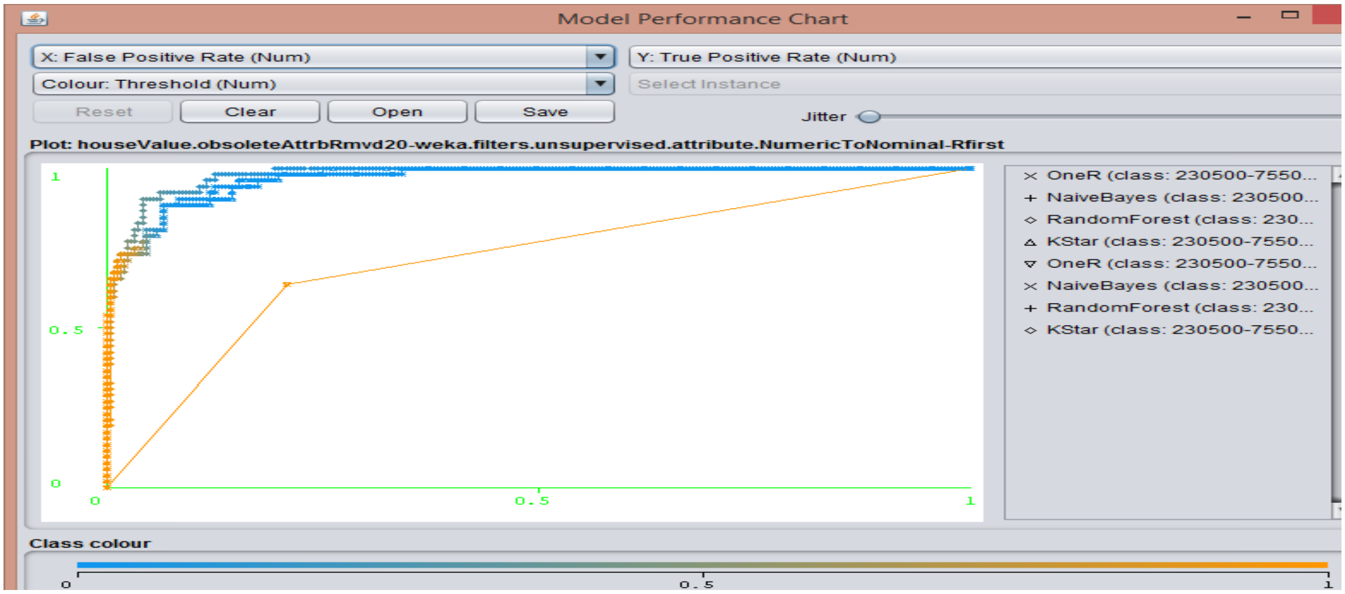
Y: True Positive Rate (Num)

Colour: Threshold (Num)

Plot: houseValue.obsoleteAttrbRmvd-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst

Select Instance
Reset
Clear
Open
Save
Jitter
1
0.5
0
0
0.5
1
× OneR (class: 230500-7550...
+ NaiveBayes (class: 230500...
◇ RandomForest (class: 230...
△ KStar (class: 230500-7550...
Class colour

5.2.2 ROC Curve for the Top20 Set



Model Performance Chart

X: False Positive Rate (Num)

Y: True Positive Rate (Num)

Colour: Threshold (Num)

Plot: houseValue.obsoleteAttrbRmvd20-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst

Select Instance
Reset
Clear
Open
Save
Jitter
1
0.5
0
0
0.5
1
× OneR (class: 230500-7550...
+ NaiveBayes (class: 230500...
◇ RandomForest (class: 230...
△ KStar (class: 230500-7550...
▽ OneR (class: 230500-7550...
× NaiveBayes (class: 230500...
+ RandomForest (class: 230...
◇ KStar (class: 230500-7550...
Class colour
0
0.5
1

The analysis of the ROC curves lets us determine that the accuracy of '**All Attributes**' is higher than '**Selected attributes'(**top 20**)** because the Area under the ROC curve is larger for '**All attributes**'.

5.3 Classifier Analysis

CLASSIFIER NAME	ACCURACY	VARIANCES
RANDOM FOREST	67.26%	0.07%
NAÏVE BAYES	65.10%	0.06%
KSTAR	61.34%	0.08%
ONER	53.12%	0.11%

The above table contains the highest accuracies and the lowest variances obtained for all the classifiers in 80/20 split condition for all attributes (condition 1) used in our experiment. We can clearly see from the table that Random Forest classifier gives the highest accuracy compared to the other classifiers and although it has the second lowest variance (Naïve Bayes has the highest), the difference between the variance of Naïve Bayes and Random Forest is very low. Therefore, we can conclude that Random Forest classifier builds the best predictive model in our experiment.

5.4 ATTRIBUTE ANALYSIS

In our experiment, we used two different sets of attributes as a factor- ALL Attributes and top 20 attributes. We tried to infer which of these two sets of attributes produced a better model for predicting the class attribute. We conclude that the set of "All attributes" builds a better predictive model for two reasons stated below:

- The ROC curve analysis revealed that the set with All Attributes with various combinations of factors produces a better area under the ROC curve.

- Looking at the results for average accuracy and variance of all classifiers, we observe that the accuracy is the highest when All Attributes set is being used. This also suggests that the classifiers build a better predictive model when provided with All attributes in the experiment.

5.5 CONCLUSION

With the average accuracy and variance, ROC curves, Attribute and Classifier evaluation we are recommending the following for our dataset:

- **Classifier:** Random forest has performed with highest accuracy and a very good stability with this dataset.

- **Number of Attributes factor:** We employed two different sets of attributes in our experiment (Top 20 and All attributes) and the All attributes set emerged to be the best factor.

- **Percentage split factor:** We employed *80/20 and 30/70 percentage split* in training/test data and found that the 80/20 split gave a better prediction model compared to the other split.