Effectiveness of Social Media Ads.

SEMI-SUPERVISED LEARNING (SSL) | Assignment - 2 | Pattern Recognition

RAMIT NANDI || MD2211

2023-09-30

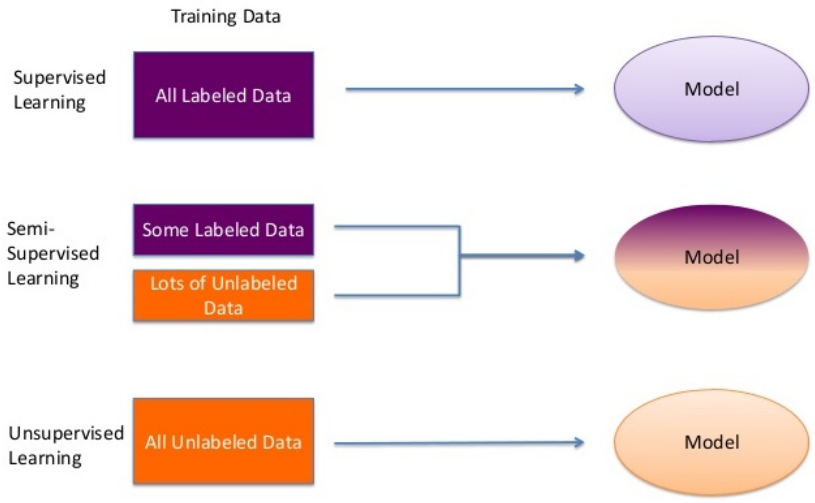
Contents

INTRODUCTION

Currently , Machine Learning algorithms are broadly classified in two categories , both having their own pros & cons -

- **Supervised Learning :** It needs labeled data, and updates model weights to minimize the average difference between predictions and labels.
  However, labeling an entire dataset may be time-consuming and expensive, and without enough labeled data it fails to reach desired quality.
- **Unsupervised Learning :** It does not need labeling, but tries to cluster points together based on similarities in some feature-space.
  But, without labels to guide training, this algorithm may find clusters which are not relevant at all.

What if we have access to both types of data? Or what if we only want to label a percentage of our dataset? How can we combine both our labeled and unlabeled datasets to improve model performance? This is where *Semi-Supervised Learning* appears as a hybrid solution.



Underlying Assumptions :

i). *Continuity assumption* : Points which are close to each other are more likely to share a label.

ii). *Cluster assumption* : The data tend to form discrete clusters, and points in the same cluster are more likely to share a label.

iii). *Manifold assumption* : The data lie approximately on a manifold of much lower dimension than the input space.

In the semi-supervised setting, we use both labeled(usually a small proportion of entire dataset) and unlabeled data(remaining larger portion of dataset).

- Labeled points act as a sanity check; they add structure to the learning problem by establishing how many classes there are, and which clusters correspond to which class.
- Unlabeled datapoints provide context; by exposing our model to as much data as possible, we can accurately estimate the shape of the whole distribution.

With both parts, semi-supervised learning can step closer to the true distribution compared to applying supervised or unsupervised learning separately.

DATA DESCRIPTION

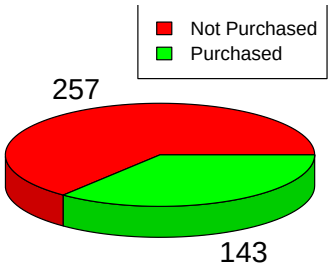
The dataset contains details of the purchase of a product based on social network advertisements. The data has 400 observations, looks as follows ...

User.ID	Gender	Age	EstimatedSalary	Purchased
15624510	Male	19	19000	0
15810944	Male	35	20000	0
15668575	Female	26	43000	0
15603246	Female	27	57000	0
15804002	Male	19	76000	0
15728773	Male	27	58000	0
15598044	Female	27	84000	0
15694829	Female	32	150000	1
15600575	Male	25	33000	0
15727311	Female	35	65000	0

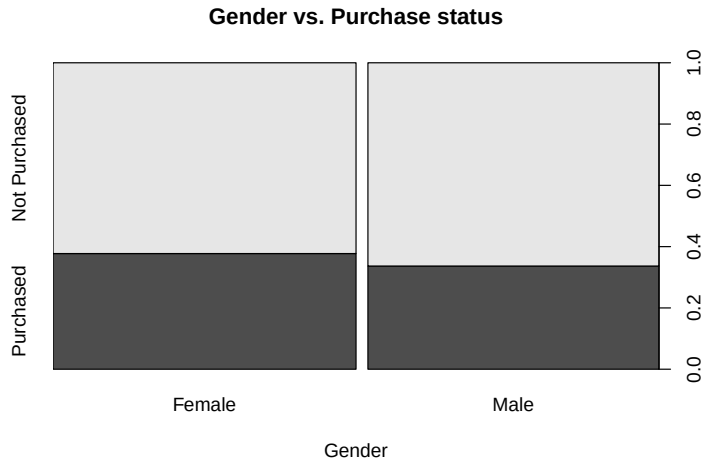
GOAL: Predicting whether a person will buy a product displayed on a social network advertisement based on his/her gender , age and approximate Salary.

EDA

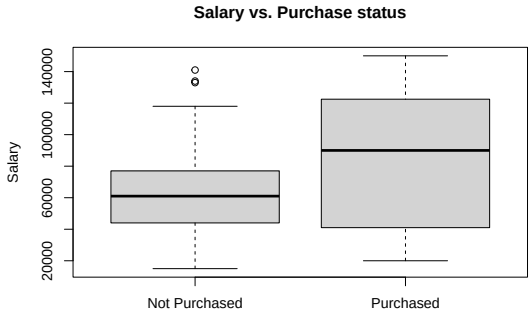
- Target class belongs to two discrete categories of purchased and not purchased. [Throughout the report , Red colour will denote 'Not Purchased' , Green colour will denote 'Purcased']



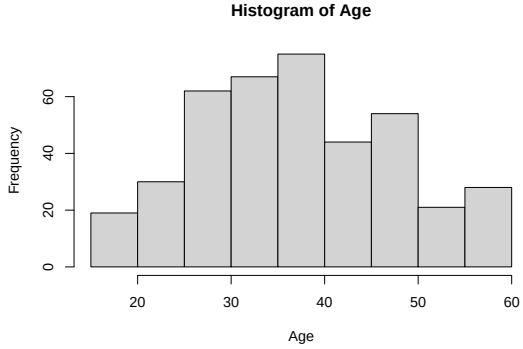
- *Gender:* Gender does not affect the purchase status much here. Given a person is male, the chance that he will buy the product is very similar to the chance of purchase ,given the person is female .

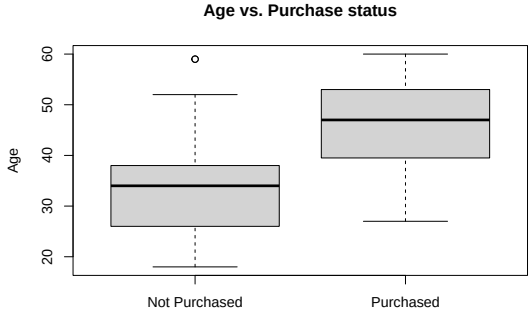


- *Salary:* Those who purchase the product , have on average higher salary than those who do not.



- *Age:* Those who purchase the product , are on average older than those who do not.





PREPARING Labeled & Unlabeled DATA for SSL

- First we will keep aside 20% data as our Final Test Set. This split will not change throughout the discussion.
- Everything will be done on the rest 80% data (i.e. 320obs.) from now on.

Here our data is such that labels are available for all observation. But to demonstrate how semi-Supervised learning works, we need both labeled & unlabeled data.
So we will split the data in 10folds, deliberately treat 6 out of 10folds in Training Set as 'Unlabeled Data' [That is simply, choose 6/10 portion of training data , delete the label column and pretend like it was not there in the first place. This split will not change throughout the discussion.] .
From 4folds with labels , we will use 2folds as 'Labeled Training Data' and remaining 2folds as 'Validation Data' for various demonstration.

Semi-SUPERVISED LEARNING

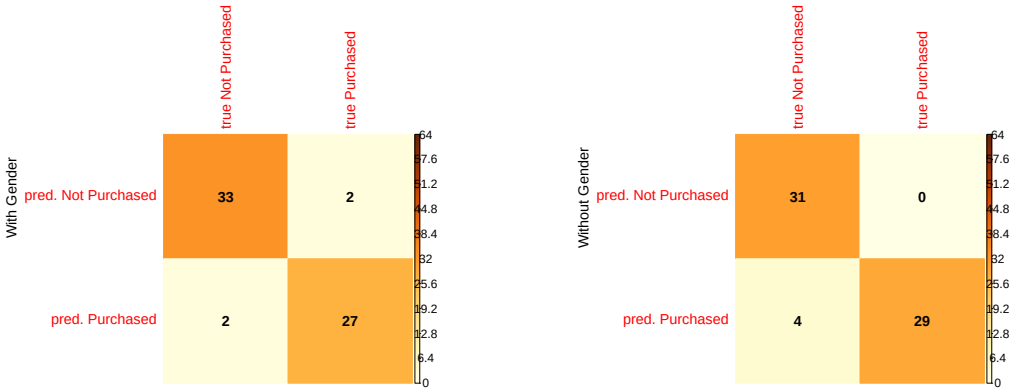
Illustration for a particular split of folds

```
## Total number of observations : 320

## All possible folds index of Training Data :  1 2 3 4 5 6 7 8 9 10

##    index for folds containing Labels [fixed] :  1 6 8 9

##         index for Validation folds [changable] :  1 9

##         index for Labeled Training folds [changable] :  6 8

##    index for Unlabeled folds [fixed] :  2 3 4 5 7 10
```

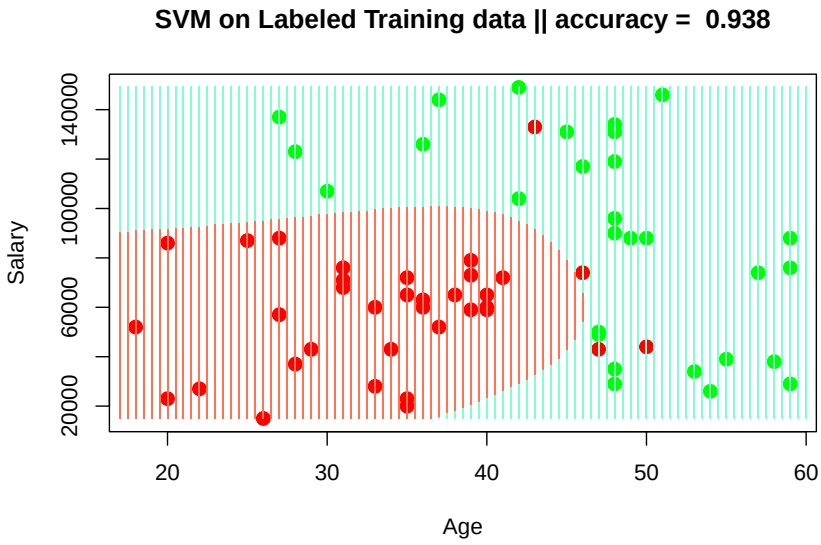
We will discuss the simplest of Semi-Supervised Learning Methods , known as 'self-training' [Other available methods are 'co-training','label propagation' etc]

step-1) Supervised Learning on Labeled Data

Based on the Labeled Training data , we will apply SVM with cubic polynomial kernel. (Cost hyperparameter C is chosen by crossvalidation, larger the value smaller the margin)

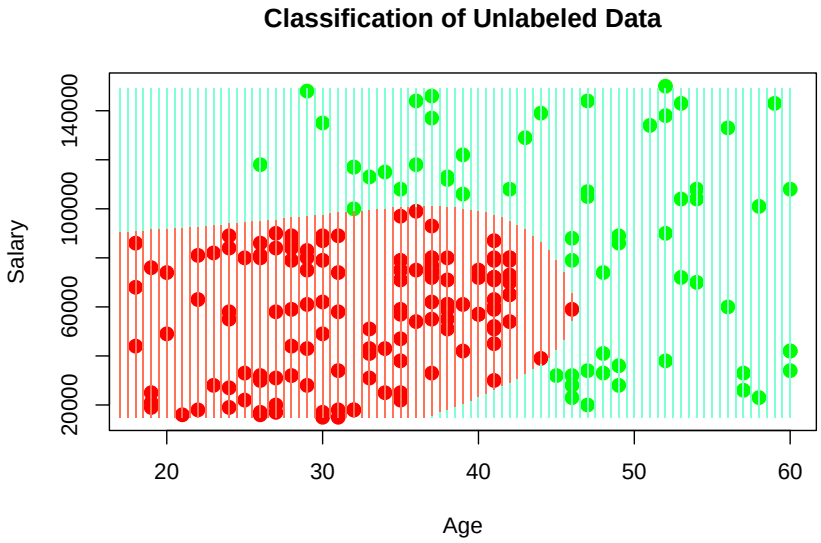


Also , We dont see very much difference in quality after dropping the feature 'Gender'. So from now on , SVM will be fitted on Age & Salary only , for simpler model.



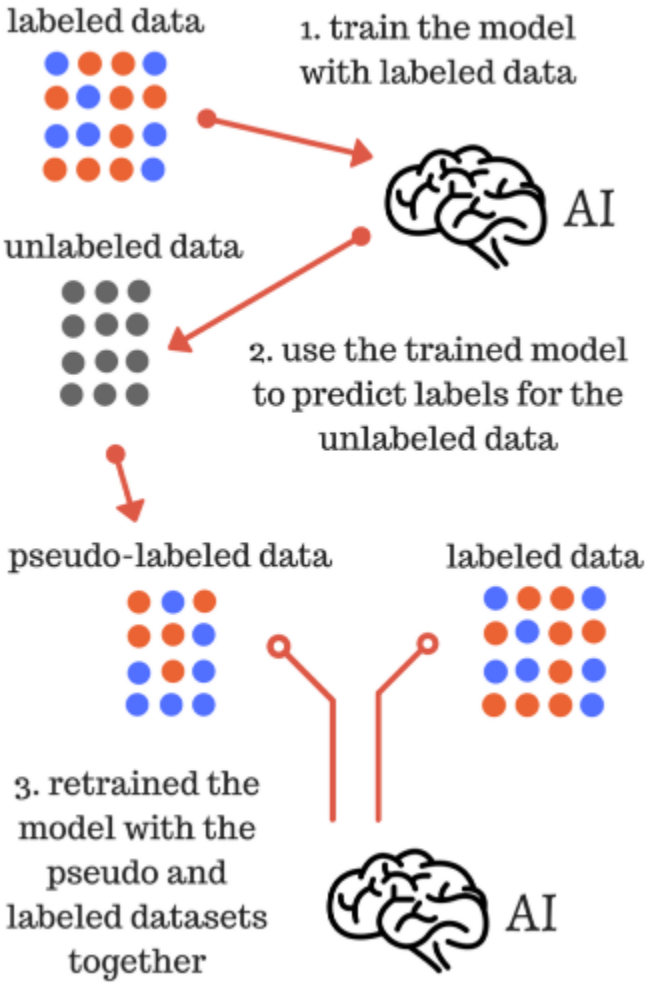
step-2) Pseudo-Labeling of Unlabeled data

Based on our SVM model fitted above , try to classify the Unlabeled observations. These predicted labels will be used as the labels of the Unlabeled data.

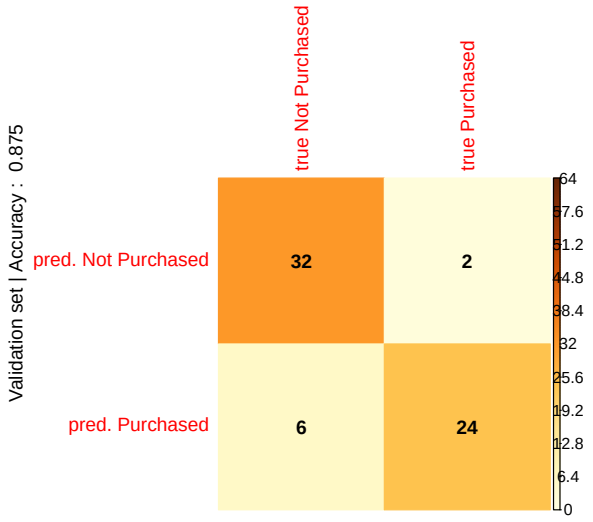


step-3) Updating the Model based on Combined data

Pool the pseudo-Labeled & true Labeled Training data together. Refit the SVM based on this pooled data.

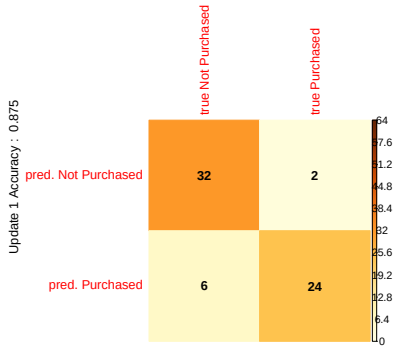


Now, predict the validation set based on this pooled model. The confusion matrix is given by ,



step-4) Further Refittings

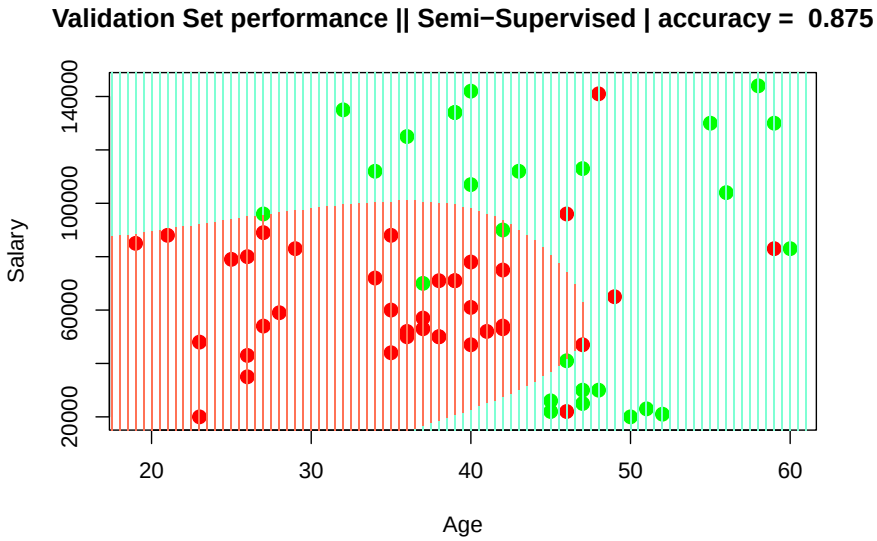
- We can predict the labels of our Unlabeled data based on our current model.
- Use these predictions as the updated pseudo-labels and refit the model based on updated pooled data.
- Also predict the validation set based on this updated model.



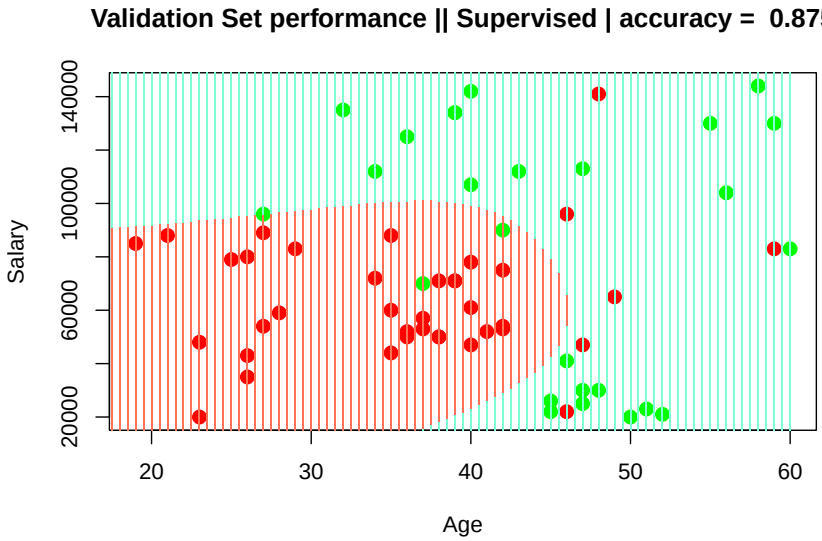
We can iterate the process upto some convergence.

COMPARISON

First, let evaluate the most updated model performance on our validation set.

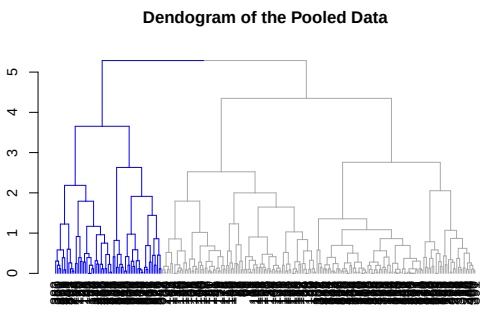


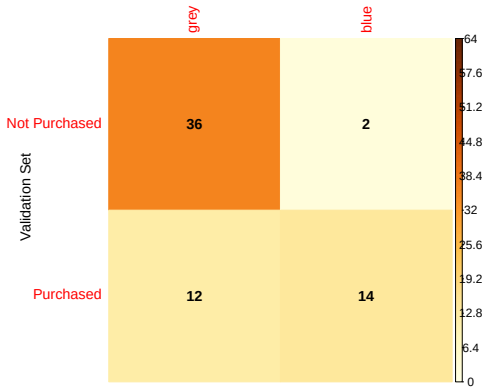
- **SUPERVISED LEARNING Only:** We already have fitted SVM based on Labeled Training data alone. This is the maximum possible scope for supervisd learning in this dataset.
  Lets check its performance on Validation set.

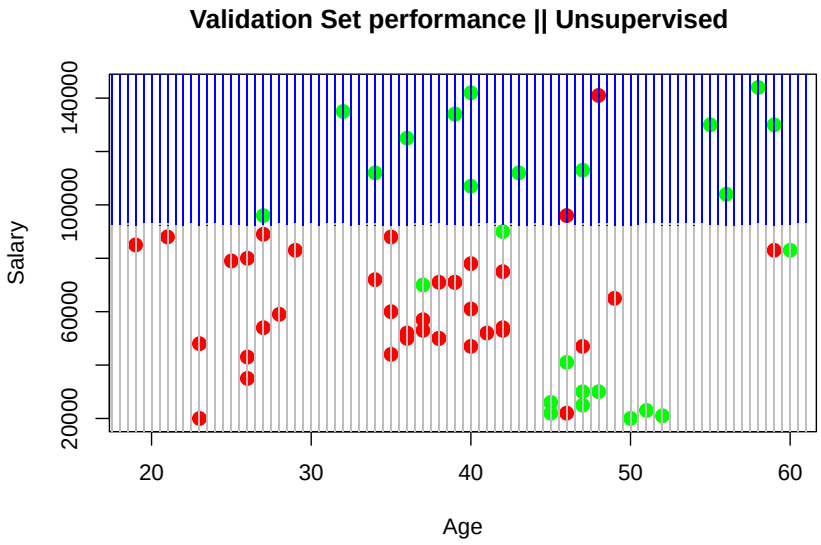


Comment: Atleast on this data , the performance is comparable.

- **UNSUPERVISED LEARNING Only:** We can ignore the labels of Labeled Training data and combine it with the Unlabeled data. As we saw earlier , Gender is not important, we will ignore that column.
  -Apply Hierarchial Clustering on the data [distance: euclidean, agglomeration: complete]. Now consider the two top-most groups.
  -Apply kNN algorithm[k: odd number $\geq \sqrt{\text{sample size}}$] to classify the validation set in those two groups identified by clustering.







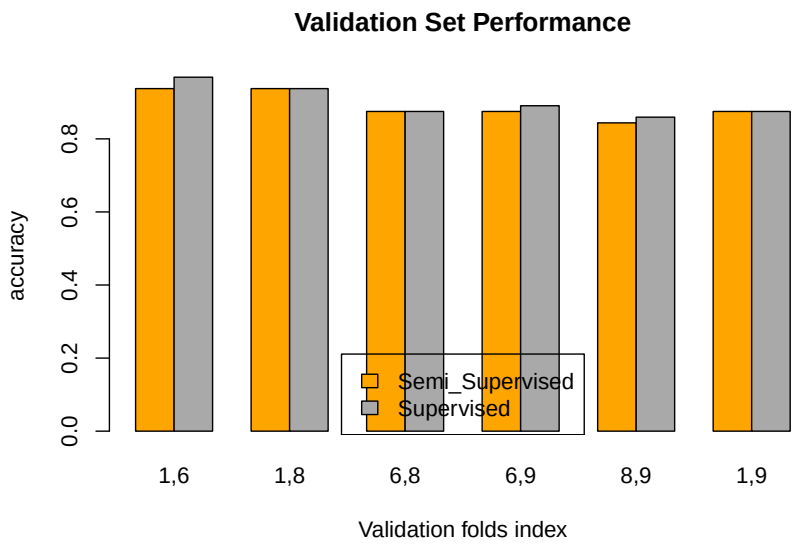
Comment: Though the persons who do not purchase are mostly classified as grey group , but there is no way to say that the decision boundary is actually for 'Purchased vs Not Purchased' classification.

Repeating the calculations for different choices of folds

Till now we were using -

```
## All possible folds index of Training Data :  1 2 3 4 5 6 7 8 9 10

##    index for folds containing Labels [fixed] :  1 6 8 9

##         index for Validation folds [changable] :  1 9

##         index for Labeled Training folds [changable] :  6 8

##    index for Unlabeled folds [fixed] :  2 3 4 5 7 10
```

Now, we will randomly choose some different indexes of folds as Validation Set & Labeled Training Set and will repeat all the steps again to have an idea of average performance. [To obtain semi-supervised learning we need supervised learning in the intermediate step. So that performances are also reported.]


Validation Set Performance
accuracy
0.0
0.2
0.4
0.6
0.8
$semi_Supervised
$supervised
1,6
1,8
6,8
6,9
8,9
1,9
Validation folds index

FINAL MODEL

Trained Model

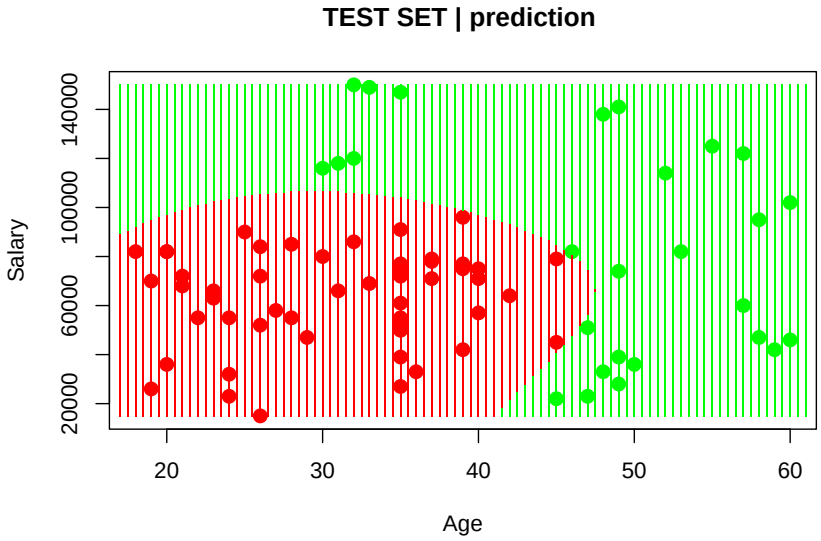
Our Final Model trained on all 320obs. , using 4folds of Labeled data & 6folds of Unlabeled data and using average value of C hyperparameter from earlier calculations is given as

```
##
## Call:
## svm(formula = Purchased ~ . - Purchased - Gender, data = pooledData,
##     cost = Hyp, kernel = "polynomial", coef0 = 1)
##
##
## Parameters:
##    SVM-Type:  C-classification
##  SVM-Kernel:  polynomial
##        cost:  3.356111
##      degree:  3
##      coef.0:  1
##
## Number of Support Vectors:  34
##
##  ( 16 18 )
##
##
## Number of Classes:  2
##
## Levels:
##  Not Purchased Purchased
```

Note that number of Support Vector is much smaller than number of observations , which implies sparse model.

```
## num. of SV : 16 18  || num. of total obs. in use : 320
```

Decision Boundary and Test Set Prediction



CONCLUSION

Looking at the nature of decision boundary we have, we can conclude
Those with high Salary are likely to purchase a product , irrespective of their Age. But younger persons are unlikely to purchase a product if the Salary is not enough.