Due: 27.11.22

# CSE454 - Data Mining

## Homework - I

1) What are the charachteristics of random forest classificaton model? What is the difference between random forest model? Give the pseudo code of it. Explain the code.

The random forest is classificaton algorithm consisting of many decision trees. It uses bagging and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by commitee is more accurate than that of any individual tree

The charachteristics of this model;

- It emphasizes feature selection
- It doesnot assume that the model has a linear relationship.
- It utilizes ensemble learning. If we were to use just 1 decision tree, we wouldn't be using ensemble learning.

A random forest takes random trees, forms many decision trees, then averages out the leaf nodes to get a clearer model

Pseudo-code;

Precondition: A training set $S := (x_1, y_1), \ldots (x_n, y_n)$, features $F$, number of trees in forest $B$.

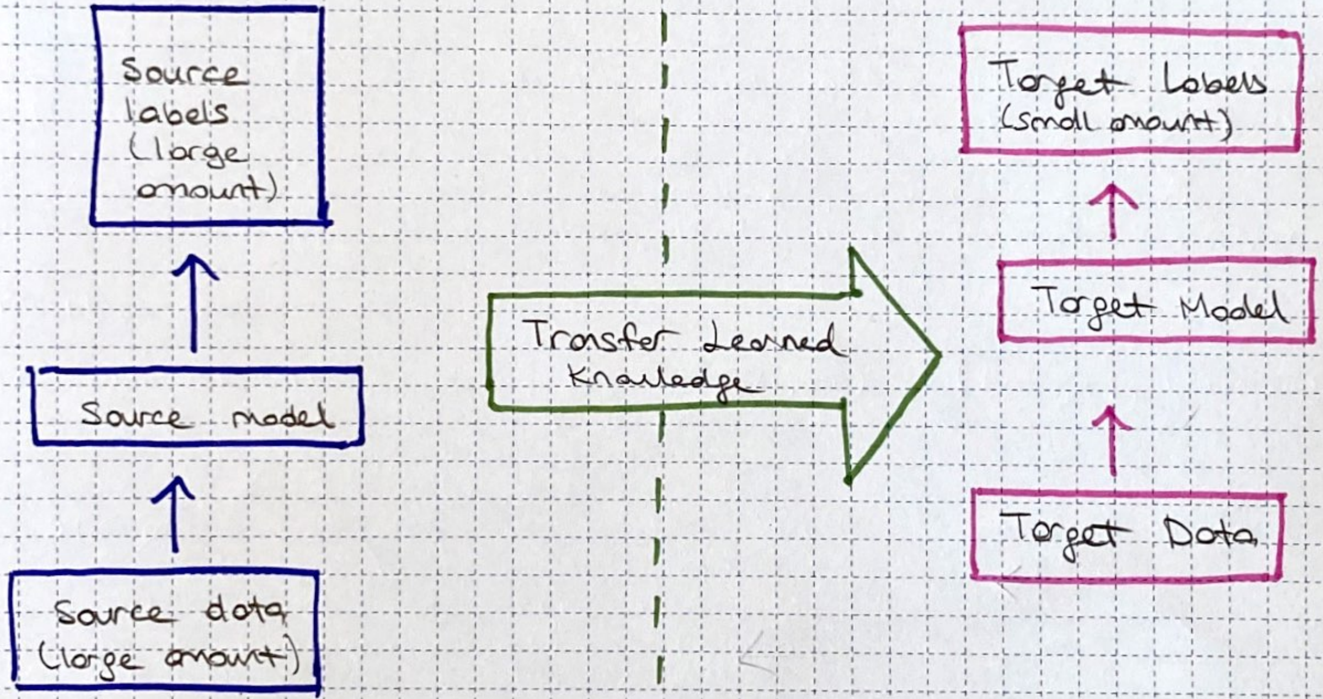
```
function RandomForest(S,F)
    $H \leftarrow \emptyset$
    for $i \in 1, \ldots, B$ do
        $S^{(i)} \leftarrow$ A bootstrap sample from S
        $h_i \leftarrow$ RandomizedTreeLearns($S^{(i)}$, F)
        $H \leftarrow H \cup \{h_i\}$
    end for
    return H
end function

function RandomizedTreeLearns(S,F)
    At each node:
        $f \leftarrow$ very small subset of F
        Split on best feature in F
    retun The learned tree
end function
```

2> What is transfer learning? Give a model and explain the model,

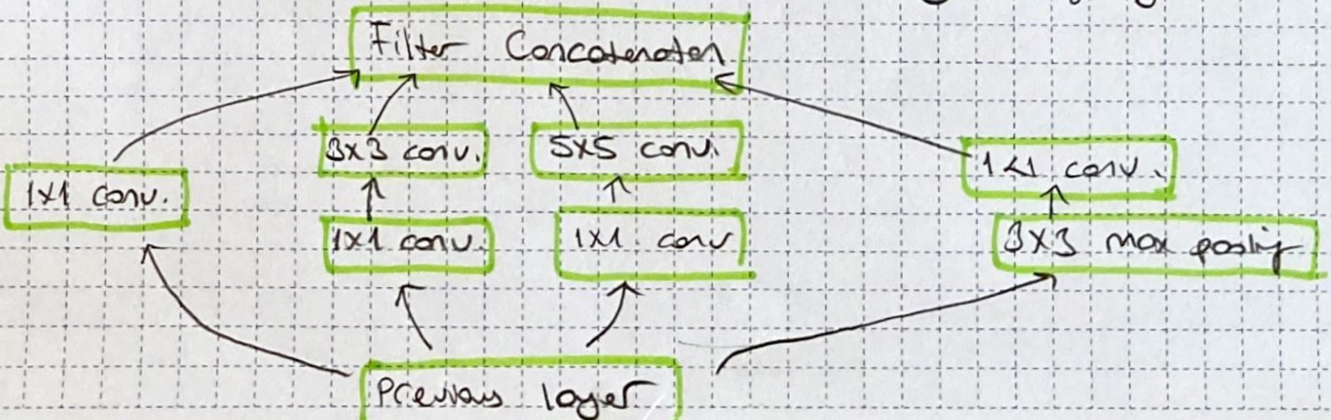
Transfer learning is a method for feature representation from pre-trained model facilitating us that we dont need to train a new model from scratch.

A pre-trained model is usually trained on a huge dataset such as ImageNet and the weights obtained from the trained can be used for any other related application with your custom neural network. These newly built models can directly be used for predictions on relatively new tasks or can be used in training processes for related applications. This approach not only reduces the training time but also lowers the generalization error.


Source labels (large amount)
Source model
Source data (large amount)
Transfer Learned Knowledge
Target Labels (small amount)
Target Model
Target Data

- Inception

This microarchitecture was introduced by Szegedy in 2014.



## 2. cont >

The goal of this module is to act as a multi-level feature extracter by computing 1x1, 3x3, 5x5 convolution within the same module of the network. The output of these filters is stacked along the channel dimension and before being fed into the next layer in the network.
The architecture of this model includes;

- 1x1 convolutional with 128 filters for dimensions and reductions and rectified linear activations.

- Fully connected layer with 1024 units and a rectified linear activation

- Dropout layer with 70% ratio

- Linear layer with softmax loss as the classifier.

- High performance gain on CNN

- Trains faster than the family of VGG.

- The size of the model is relatively smaller than VGG, where VGG can weigh upto 500 MB's inception is about 100MB's.

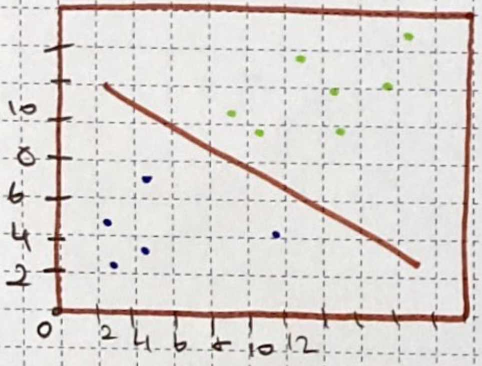
3> Explain support vector machine model in details. What are the advantages and disadvantages of it?

Support vector machines are a set of supervised learning methods used for classification, regression and outlier - detection. They can be used to detect cancerous cells based on millions of images or to predict future driving routes with a well-fitted regression model.

The main thig to keep in mind here is that they are just math equations turned to give you the most accurate answer possible as quickly as possible. SVM's are different from other classification algorithms because of the way they choose the decision bounday that maximizes the distance from the nearest data points of all the classes. The decision bounday created by SVM's is called the max. margin classifier or the max margin hyper-plane

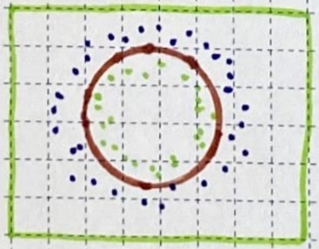
A simple linear SVM classifier works by making a straight line between two classes. That means all of the data points on one side of the line will represet a category and the data points on the other side of the line will be put into a different category.

It chooses the line that separates the data and is the furtest away from the closest data points as possible.



- linear SVM -

The decision bounday doesn't have to be a line.



- non linear SVM -

**- PROS -**

- Effective on datasets with multiple features.
- Effectice in cases where number of features is greater than the number of data points.
- Uses a subset of training points in the decision function called support vectors which makes it memory efficient.

**- CONS -**

- If the number of features is a lot bigger than the number of datapoints, avoiding over-fitting when choosing kernel functions and regularization term is crucial.
- SVM doesn't directly provide estimates.
- Works best on small sample datasets because of its high training time.

4) Explain fastText classification model in details. What are the advantages and disadvantages of it?

FastText is an open-source, free library from FAIR for learning word embeddings and word classification. This model allows creating unsupervised learning or supervised learning algorithm for obtaining vector representations for words. It also evaluates these models.

FastText supports both CBOW and skip-gram models.

→ Uses of FastText:

1- It used for finding semantic similarities.
2- It can also be used for text classification (ex: spam filtering)
3- It can train large datasets in minutes.

→ Working of fastText:

FastText is very fast in training word vector models. You can train about 1 billion words in less than 10 minutes. The models built through deep neural network can be slow to train and test. These methods use a linear classifier to train the model.

Linear classifier: The vector corresponding to the text is closer to its corresponding label.
To find the P of correct label given, we use "softmax".

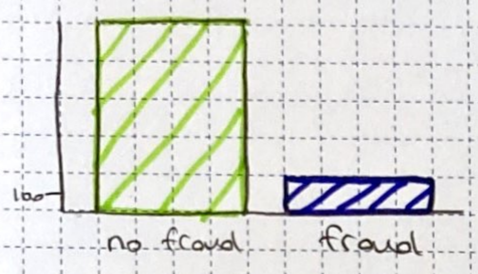
$$\boxed{\text{softmax}(w_{travel}) = \frac{e^{w_{car}^{T} * w_{travel}}}{\sum_{\substack{labels\\T}} e^{w_{car}^{T} * w_{t}}}}$$

Here, travel is the label and car is the text associated to it. To maximize this probability, of the correct label, we can use Gradient Descent algorithm.

- Word vectors generated through fastText hold extra information about their sub-words.
- It also allows for capturing the meaning of suffixes/prefixes for the given words in the corpus.
- It allows for generating better word embeddings for different or rare words as well.
- It can also generate word embeddings for OOV words.
- While using fastText, even if you don't remove the stopwords, still the accuracy is not compromised.

5) What are the techniques used for class imbalance problem. Give specific techniques and explain each of them.

When observation in one class is higher than the observation in other classes then there exists a class imbalance EX: To detect fraudulent credit card transactions. As you can see in the below graph fraudulent transactions is around 400 when compared with non-fraudulent transaction around 90000



Class imbalance is a common problem in ML, especially in classification problem. Imbalance data can hamper our model accuracy big time

Class imbalance appear in many domains, including;
- Fraud detection
- Spam filtering
- Disease screening
- SaaS subscription churn
- Advertising click-throughs

- Resampling Techique;
It consists of removing samples from the majority class (under-sampling) and/or adding more examples from the minority class (over-sampling).

- Random Under-sampling:
Under-sampling can be good choice when you have a ton of data. It can be defined as "removing some observations of the majority class".

- Random over-sampling:
It can be defined as adding more copies to the minority class. It is a good choice when you don't have a ton of data.

- Random under-sampling with imblearn (library)
- from imblearn.under_sampling import RandomUnderSampling.
RandomUnderSampling is a fast and easy way to balance the data.

- Random over-sampling with imblearn (same with above)

- Under-sampling: Tomek links.
Tomek links are pairs of very close instances but of opposite classes. Removing them increases the space between the two classes.

- Synthethic Minority over-Sampling Techique (SMOTE)
It works by randomly picking a point from the minority class and computing the k-nearest neighbors for this point.