**Assignment 14.1**

**Problem​ ​Statement**

1. **What are the three stages to build the hypotheses or model in machine learning?**

**Answer:**

The more disciplined you are in handling your data, the more consistent and better results you are

Likely to achieve. The process of getting data ready for Machine Learning algorithm can be

Summarized in three steps:

Step 1: Select Data

Step 2: Pre-process Data

Step 3: Transform Data

You can follow this process in a linear manner, but it is very likely to be iterative with many loops.

Step 1: Select Data

This step is concerned with selecting the subset of all available data that you will be working with.

There is always tendency for including all data that is available, that the maxim “more is better” will

Hold. This may or may not be true.

You need to consider what data you actually need to address the question or problem you are

working on. Make some assumptions about the data you require and be careful to record those

assumptions so that you can test them later if needed.

Below are some questions that'll help you think through this process:

What is the extent of the data that is available? For example through time, database tables, connected systems. Ensure you have a clear picture of everything that you can use. What data is not available that you wanted? For example data that is not recorded or cannot be recorded. You may be able to derive or simulate this data. What data is irrelevant to address the problem?

Excluding data is almost always easier than including data. Note down which data you excluded and why. It is only in small problems, like competition or toy datasets where the data has already been selected for you.

Step 2: Preprocess Data

After you have selected the data, you need to consider how you are going to use the data. This

preprocessing step is about getting the selected data into a form that you can work.

Three common data preprocessing steps are formatting, cleaning and sampling:

Formatting: The data you have selected may not be in a format that is suitable for you to work with.

The data may be in a relational database and you would like it in a flat file, or the data may be in a

proprietary file format and you would like it in a relational database or a text file. Cleaning: Cleaning

data is removing or fixing of missing data. There may be data instances that are incomplete and do

not carry the data that might be useful to address the problem. These instances may need to be

removed. Additionally, there may be sensitive information in some of the attributes and these

attributes may need to be anonymized or removed from the data entirely. Sampling: There may be

far more selected data available than you need to work with. More data can result in much longer

running times for algorithms and larger computational and memory requirements. You can take a

smaller representative sample of the selected data that may be much faster for exploring and

prototyping solutions before considering the whole dataset. It is very likely that the Machine learning tools you use on the data will influence the preprocessing you will be required to perform. You will probably revisit this step.

So much data So much data Photo attributed to Marc Smith, some rights reserved

Step 3: Transform Data

The final step is to transform the processed data. The specific algorithm you are working with and

the knowledge of the problem domain will influence this step and you might have to revisit different

transformations of your pre-processed data as you work on your problem.

Three common data transformations are scaling, attribute decompositions and attribute aggregations. This step is also referred to as feature engineering.

Scaling: The preprocessed data may contain attributes with a mixtures of scales for various quantities such as dollars, kilograms and sales volume. Many machine learning methods like data attributes to have the same scale such as between 0 and 1 for the smallest and largest value for a given feature. Consider any feature scaling you may need to perform.

Decomposition: There may be features that represent a complex concept that may be more useful to a machine learning method when split into the constituent parts. An example is a date that may have day and time components that in turn could be split out further. Perhaps only the hour of day is relevant to the problem being solved. Consider what feature decompositions you can perform.

Aggregation: There may be features that can be aggregated into a single feature that would be more meaningful to the problem you are trying to solve. For example, there may be a data instances for each time a customer logged into a system that could be aggregated into a count for the number of logins allowing the additional instances to be discarded. Consider what type of feature aggregations could perform.

1. **What is the standard approach to supervised learning?**

**Answer:**

Supervised learning algorithms are trained using labeled examples, such as an input where the desired output is known. For example, a piece of equipment could have data points labeled either

“F” (failed) or “R” (runs). The learning algorithm receives a set of inputs along with the

corresponding correct outputs, and the algorithm learns by comparing its actual output with correct

outputs to find errors. It then modifies the model accordingly. Through methods like classification,

regression, prediction and gradient boosting. Supervised learning uses patterns to predict the values of the label on additional unlabeled data. Supervised learning is commonly used in applications where historical data predicts likely future events. For example, it can anticipate when credit card transactions are likely to be fraudulent or which insurance customer is likely to file a claim.

Supervised algorithms can further be divided into following:

**Classification**. When the data is being used to predict a category, supervised learning is also called classification. This is the case when assigning an image as a picture of either a 'cat' or a 'dog'. When there are only two choices, it's called two-class or binomial classification. When there are more categories, like predicting the winner of the NCAA March Madness tournament, this problem is known as multi-class classification.

**Regression**. When a value is being predicted, as with stock prices, supervised learning is called regression.

**Anomaly detection**. Sometimes the goal is to identify data points that are simply unusual.

In fraud detection, for example, any highly unusual credit card spending patterns is considered to be a suspect. The possible variations are so numerous and the training examples so few, that it's not feasible to learn what fraudulent activity looks like. The approach that anomaly detection takes is to simply learn what normal activity looks like (using a history of non-fraudulent transactions) and identify anything that is significantly different.

1. **What is Training set and Test set?**

**Answer**

In Machine Learning, a training set is a dataset used to train a model. In training the model, specific

features are picked out from the training set. These features are then incorporated into the model.

Thereby, if the training set is labeled correctly, the model should be able to learn something from

these features.

**What is a Test Set?**

The test set is a dataset used to measure how well the model performs at making predictions on

that test set. If the prediction scores for the test set are unreasonable, we’ll have to make some

adjustments to our model and try again.

**4. What is the general principle of an ensemble method and what is bagging and**

**boosting in ensemble method?**

**Answer:**

Ensemble methods are supervized learning models which combine the predictions of multiple

smaller models to improve predictive power and generalization.

The smaller models that combine to make the ensemble model are referred to as base models.

Ensemble methods often result in considerably higher performance than any of the individual base

models could achieve.

**When to use ensembles**

- Medical diagnoses

- Predicting disease outbreak, natrual disasters

- Stock market predictions

- AI

Or any case where the highest performance is desired at the expense of model interpretability.

**BAGGING**

Several estimators are built independently on subsets of the data and their predictions are averaged. Typically the combined estimator is usually better than any of the single base estimator.

Bagging can reduce variance with little to no effect on bias.

ex: Random Forests

**BOOSTING**

Base estimators are built sequentially. Each subsequent estimator focuses on the weaknesses of

the previous estimators. In essence several weak models "team up" to produce a powerful ensemble model. (We will discuss these later this week.)

Boosting can reduce bias without incurring higher variance.

ex: Gradient Boosted Trees, AdaBoost

1. **How can you avoid overfitting?**

**Answer:**

Inconsistent instances in a training set. This leads to a discussion of methods for avoiding or reducing overfitting of a decision tree to training data. Overfitting arises when a decision tree is excessively dependent on irrelevant features of the training data with the result that its predictive power for unseen instances is reduced.

Two approaches to avoiding overfitting are distinguished: pre-pruning (generating a tree with fewer branches than would otherwise be the case) and post-pruning (generating a tree in full and then removing parts of it). Results are given for pre-pruning using either a size or a maximum depth cutoff. A method of post-pruning a decision tree based on comparing the static and backed-up estimated error rates at each node is also described.