**MACHINE LEARNING ASIGNMENT\_7**

**1.What is the definition of a target function? In the sense of a real-life example, express the target function. How is a target function’s fitness assessed?**

The target variable is the feature of a dataset that you want to understand more clearly. It is the variable that the user would want to predict using the rest of the dataset. In most situations, a supervised [machine learning algorithm](https://h2o.ai/wiki/machine-learning-algorithms) is used to derive the target variable. Such an algorithm uses historical data to learn patterns and uncover relationships between other parts of your dataset and the target. Target variables may vary depending on the goal and available data.

When building a machine learning solution for measuring customer attrition in the telecommunication industry, you need to spend significant time observing, weeks or even months, as some customers unsubscribe and others renew. Once you have enough training instances to build an accurate machine learning model, you can begin using machine learning in production.

In the absence of a labeled target, supervised machine learning algorithms would not be able to map available data to outcomes.

**Example:** A child would be incapable of figuring out that dogs are called dogs without being told a few times. Well-defined targets are important, as the only thing the algorithm does is learn a function that maps the relationship between input data and the target. The model’s outcomes mean nothing if the target isn’t well understood.

**2. What are predictive models, and how do they work? What are descriptive types, and how do you use them? Examples of both types of models should be provided. Distinguish between these two forms of models.**

In short, predictive modeling is a statistical technique using machine learning and data mining to predict and forecast likely future outcomes with the aid of historical and existing data. It works by analyzing current and historical data and projecting what it learns on a model generated to forecast likely outcomes. Predictive modeling can be used to predict just about anything, from TV ratings and a customer’s next purchase to credit risks and corporate earnings.

The top five predictive analytics models are:

1. Classification model: Considered the simplest model, it categorizes data for simple and direct query response. An example use case would be to answer the question “Is this a fraudulent transaction?”
2. Clustering model: This model nests data together by common attributes. It works by grouping things or people with shared characteristics or behaviors and plans strategies for each group at a larger scale. An example is in determining credit risk for a loan applicant based on what other people in the same or a similar situation did in the past.
3. Forecast model: This is a very popular model, and it works on anything with a numerical value based on learning from historical data. For example, in answering how much lettuce a restaurant should order next week or how many calls a customer support agent should be able to handle per day or week, the system looks back to historical data.
4. **Outliers model:** This model works by analyzing abnormal or outlying data points. For example, a bank might use an outlier model to identify fraud by asking whether a transaction is outside of the customer’s normal buying habits or whether an expense in a given category is normal or not. For example, a $1,000 credit card charge for a washer and dryer in the cardholder’s preferred big box store would not be alarming, but $1,000 spent on designer clothing in a location where the customer has never charged other items might be indicative of a breached account.
5. **Time series model:** This model evaluates a sequence of data points based on time. For example, the number of stroke patients admitted to the hospital in the last four months is used to predict how many patients the hospital might expect to admit next week, next month or the rest of the year. A single metric measured and compared over time is thus more meaningful than a simple average.

**Descriptive Analysis** in Machine Learning is all about perspective to understand the data and its different existing patterns. Basically, it is part of four types of Data Analysis concepts. Descriptive Analysis mainly tends more towards unsupervised learning for outlining, classifying, and drawing out information to get the answers for what had happened in past. It also helps people in the proper understanding of certain scenarios and its outcome. Thus helping them to take proper decisions related to business. Actually, it is one of the simplest and easiest ones to understand and implement with the help of a number of available tools and minimum skills. Hence it is quite beneficial for[AI Startups](http://appengine.ai/) to analyze their business. Also, it is extremely useful in recognizing the structure of data, and different patterns of data.

**Different Categories of Descriptive Analysis**

Descriptive Analysis is normally differentiated into groups. Now we shall be having a look at them,

* **Operational Reports**: Operational Reports in simple words would be the report card of an organization. Which is helpful in understanding the past working and performance of a certain organization.
* Statistical Reports: Statistical Reports are nothing but reports that show data of an organization in a statistical form which is often easy to understand compare with past and present outcomes. To represent statistical reports various statistical analysis methods are used.
* Data Mining approaches: Data mining is all about extracting exact data from different unknown data patterns. Basically, it is discovery processes of data that are useful in understanding different properties of data.

Hence we can conclude that Descriptive Analysis is an extremely useful initial process in Data analysis. And its application in different Data Analysis techniques is always fruitful.

**3. Describe the method of assessing a classification model’s efficiency in detail. Describe the various measurement parameters.**

After building a predictive classification model, we need to evaluate the performance of the model, that is how good the model is in predicting the outcome of new observations test data that have been not used to train the model.

In other words you need to estimate the model prediction accuracy and prediction errors using a new test data set. Because we know the actual outcome of observations in the test data set, the performance of the predictive model can be assessed by comparing the predicted outcome values against the known outcome values.

This chapter describes the commonly used metrics and methods for assessing the performance of predictive classification models, including:

* Average classification accuracy, representing the proportion of correctly classified observations.
* Confusion matrix, which is 2x2 table showing four parameters, including the number of true positives, true negatives, false negatives and false positives.
* Precision, Recall and Specificity, which are three major performance metrics describing a predictive classification model
* ROC curve, which is a graphical summary of the overall performance of the model, showing the proportion of true positives and false positives at all possible values of probability cutoff. The Area Under the Curve (AUC) summarizes the overall performance of the classifier

**4.i. In the sense of machine learning models, what is underfitting? What is the most common reason for underfitting?**

**ii. What does it mean to overfit? When is it going to happen?**

**iii. In the sense of model fitting, explain the bias-variance trade-off.**

**Underfitting** is a scenario in data science where a data model is unable to capture the relationship between the input and output variables accurately, generating a high error rate on both the training set and unseen data. It occurs when a model is too simple, which can be a result of a model needing more training time, more input features, or less regularization. Like overfitting, when a model is underfitted, it cannot establish the dominant trend within the data, resulting in training errors and poor performance of the model. If a model cannot generalize well to new data, then it cannot be leveraged for classification or prediction tasks. Generalization of a model to new data is ultimately what allows us to use machine learning algorithms every day to make predictions and classify data.

High bias and low variance are good indicators of underfitting. Since this behavior can be seen while using the training dataset, underfitted models are usually easier to identify than overfitted ones.

**Overfitting** is a concept in data science, which occurs when a statistical model fits exactly against its training data. When this happens, the algorithm unfortunately cannot perform accurately against unseen data, defeating its purpose. Generalization of a model to new data is ultimately what allows us to use machine learning algorithms every day to make predictions and classify data.

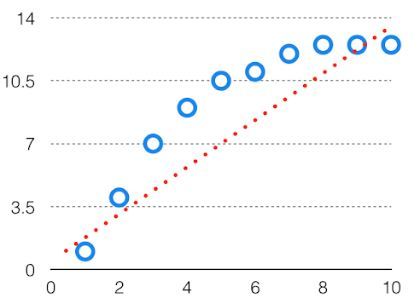
When machine learning algorithms are constructed, they leverage a sample dataset to train the model. However, when the model trains for too long on sample data or when the model is too complex, it can start to learn the “noise,” or irrelevant information, within the dataset. When the model memorizes the noise and fits too closely to the training set, the model becomes “overfitted,” and it is unable to generalize well to new data. If a model cannot generalize well to new data, then it will not be able to perform the classification or prediction tasks that it was intended for.

Low error rates and a high variance are good indicators of overfitting. In order to prevent this type of behavior, part of the training dataset is typically set aside as the “test set” to check for overfitting. If the training data has a low error rate and the test data has a high error rate, it signals overfitting.

You only get accurate predictions if the machine learning model generalizes to all types of data within its domain. Overfitting occurs when the model cannot generalize and fits too closely to the training dataset instead. Overfitting happens due to several reasons, such as:  
•    The training data size is too small and does not contain enough data samples to accurately represent all possible input data values.  
•    The training data contains large amounts of irrelevant information, called noisy data.  
•    The model trains for too long on a single sample set of data.  
•    The model complexity is high, so it learns the noise within the training data.

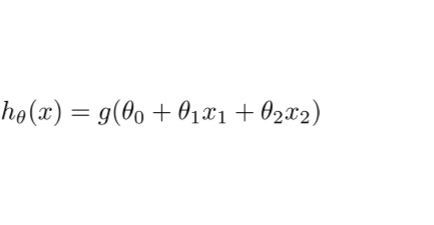
It is important to understand prediction errors (bias and variance) when it comes to accuracy in any machine learning algorithm. There is a tradeoff between a model’s ability to minimize bias and variance which is referred to as the best solution for selecting a value of Regularization constant. Proper understanding of these errors would help to avoid the overfitting and underfitting of a data set while training the algorithm.

Bias  
The bias is known as the difference between the prediction of the values by the ML model and the correct value. Being high in biasing gives a large error in training as well as testing data. Its recommended that an algorithm should always be low biased to avoid the problem of underfitting.  
By high bias, the data predicted is in a straight line format, thus not fitting accurately in the data in the data set. Such fitting is known as Underfitting of Data. This happens when the hypothesis is too simple or linear in nature. Refer to the graph given below for an example of such a situation.

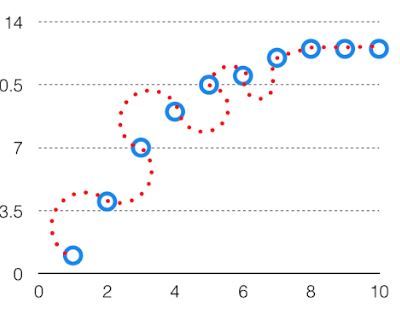


HighBias

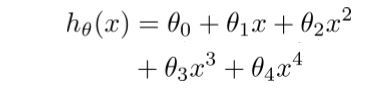
In such a problem, a hypothesis looks like follows.

Variance  
The variability of model prediction for a given data point which tells us spread of our data is called the variance of the model. The model with high variance has a very complex fit to the training data and thus is not able to fit accurately on the data which it hasn’t seen before. As a result, such models perform very well on training data but has high error rates on test data.  
When a model is high on variance, it is then said to as Overfitting of Data. Overfitting is fitting the training set accurately via complex curve and high order hypothesis but is not the solution as the error with unseen data is high.While training a data model variance should be kept low.

The high variance data looks like follows.

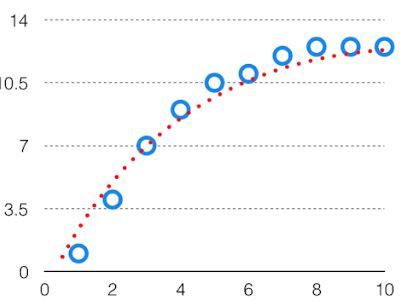


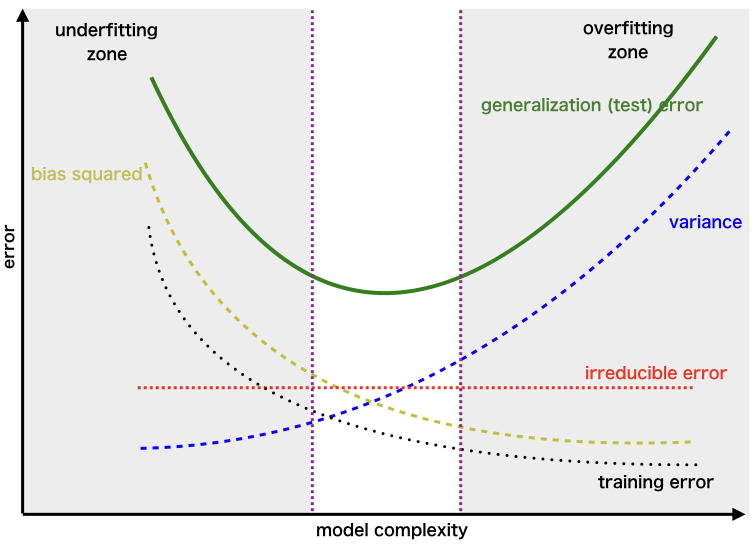
High Variance

In such a problem, a hypothesis looks like follows.  
  
**Bias Variance Trade off**

If the algorithm is too simple (hypothesis with linear eq.) then it may be on high bias and low variance condition and thus is error-prone. If algorithms fit too complex ( hypothesis with high degree eq.) then it may be on high variance and low bias. In the latter condition, the new entries will not perform well. Well, there is something between both of these conditions, known as Trade-off or Bias Variance Trade-off.

This tradeoff in complexity is why there is a tradeoff between bias and variance. An algorithm can’t be more complex and less complex at the same time. For the graph, the perfect tradeoff will be like.

  
The best fit will be given by hypothesis on the tradeoff point.

The error to complexity graph to show trade-off is given as –  
  
This is referred to as the best point chosen for the training of the algorithm which gives low error in training as well as testing data.

1. **Is it possible to boost the efficiency of a learning model? If so, please clarify how.**

There are five ways to improve efficiency of a learning model:

### 1. Choosing the Right Algorithms

Algorithms are the key factor used to train the ML models. The data feed into this that helps the model to learn from and predict with accurate results. Hence, choosing the right algorithm is important to ensure the performance of your machine learning model.

Linear Regression, Logistic Regression, Decision Tree, SVM, Naive Bayes, kNN, K-Means, Random Forest and Dimensionality Reduction Algorithms and Gradient Boosting are the leading ML algorithms you can choose as per your ML model compatibility.

### 2. Use the Right Quantity of Data

The next important factor you can consider while developing a machine learning model is choosing the right quantity of data sets. And there are multirole factors and for deep learning-based ML models, a huge quantity of datasets is required for algorithms.

Depending on the complexities of problem and learning algorithms, model skill, data size evaluation and use of statistical heuristic rule are the leading factors determine the quantity and [types of training data sets](https://www.cogitotech.com/blog/what-are-the-various-types-of-data-sets-used-in-machine-learning/) that help in improving the performance of the model.

### 3. Quality of Training Data Sets

Just like quantity, the quality of machine learning training data set is another key factor, you need to keep in mind while developing an ML model. If the quality of [machine learning training data sets](https://www.cogitotech.com/services/machine-learning/) is not good or accurate your model will never give accurate results, affecting the overall performance of the model not suitable to use in real-life.

Actually, there are different methods to measure the quality of the training data set. Standard quality-assurance methods and detailed for in-depth quality assessment are the leading two popular methods you can use to ensure the quality of data sets. Quality of data is important to get unbiased decisions from the ML models, so you need to make sure to use the right quality of training data sets to improve the performance of your ML model.

### 4. Supervised or Unsupervised ML

Moreover, the above-discussed ML algorithms, the performance of such AI-based models are affected by methods or process of machine learning. And supervised, unsupervised and reinforcement learning are the algorithm consist of a target/outcome variable (or dependent variable) which is to be predicted from a given set of predictors (independent variables).

In unsupervised machine learning, a model is given any target or outcome variable to predict/estimate. And, it is used for clustering population in different groups, which is widely used for segmenting customers in different groups for specific intervention. For supervised ML, labeled or annotated data is required, while for unsupervised ML the approach is different.

Similarly, reinforcement Learning is another important algorithm, used to train the model to make specific decisions. In this training process, the machine learns from previous experiences and tries to store the best suitable knowledge for the right predictions.

### 5. Model Validation and Testing

Building a machine learning model is not enough to get the right predictions, as you have to check the accuracy and need to validate the same to ensure get the precise results. And validating the model will improve the performance of the ML model.

Actually, there are various types of validation techniques you can follow but you need to make sure choose the best one that is suitable for your [ML model validation](https://www.cogitotech.com/ml-model-validation-services/) and help you to improve the overall performance of your ML model and predict in an unbiased manner. Similarly, testing of the model is also important to ensure its accuracy and performance.

**7.How would you rate an unsupervised learning model’s success? What are the most common success indicators for an unsupervised learning model?**

Unsupervised learning algorithms are comparatively difficult to evaluate as we don’t necessarily have any labeled data to go ahead with the testing process. The following are some of the mathematical measures to assess the quality of our predictions for unsupervised learning:

## Adjusted Rand Index:

### This is one of the variations of the classic Rand Index that tries to get the proportion of the cluster assignments which are “accurate.” It does so by computing the similarity measure between two different clusterings, taking into consideration all pairs of inputs and numbering those pairs that are assigned in the same or a different cluster, then comparing the same with the random probability of assignment of these clusters. The measuring metric is available in sklearn and can be directly used to create a quantifiable measure for [clustering algorithms](https://deepchecks.com/glossary/clustering-algorithms/) through the following clustering metrics: 1.) Fowlkes-Mallows Score

It is a bit similar in its results to the ARI as it also attempts to look at the cluster assignments that are accurate. This clustering metric computes the GM (Geometric Mean) between the precision and recall metric, and like most of the supervised learning measuring metrics, it is capped between the values of 0 – 1, with a larger value indicating that the algorithm did a good job at assigning individual samples to respective clusters. This is also available as a metric in sklearn and is therefore easy to implement.

### 2.) Silhouette Score

This algorithmtries to explain the extent to which two data points are similar to each other, given that they are assigned to the same cluster. The score is computed based on the aggregation of all the data points so we can have an average of the performance of the implementation in the entirety of the dataset. This metric measure lies between negative 1 and positive 1. Positive values are desired in this mechanism as a negative value indicates that the data points in the clusters are not very similar to each other.

## Calinski-Harabasz Index

This implementation looks at the ratio of the variance of individual data points compared to the points in the clusters other than the one in which the current instance is attributed and the data points within the current cluster. For this metric, higher values are preferred

**7. Is it possible to use a classification model for numerical data or a regression model for categorical data with a classification model? Explain your answer.**

**8. Describe the predictive modeling method for numerical values. What distinguishes it from categorical predictive modeling?**

Predictive modeling is a statistical approach that analyzes data patterns to determine future events or outcomes. It's an essential aspect of predictive analytics, a type of [data analytics](https://www.projectpro.io/article/big-data-analytics-projects-for-students-/436) that involves [machine learning and data mining](https://www.projectpro.io/article/data-mining-vs-machine-learning-differences/524) approaches to predict activity, behavior, and trends using current and past data.

For instance-

* healthcare organizations apply predictive modeling techniques to optimize diagnostic procedures,
* banking institutions use these techniques to detect and avoid fraudulent activities,
* retail stores implement such techniques to optimize their inventory stock and boost customer satisfaction, etc.

**The top five predictive analytics models are:**

1. Classification model: Considered the simplest model, it categorizes data for simple and direct query response. An example use case would be to answer the question “Is this a fraudulent transaction?”
2. Clustering model: This model nests data together by common attributes. It works by grouping things or people with shared characteristics or behaviors and plans strategies for each group at a larger scale. An example is in determining credit risk for a loan applicant based on what other people in the same or a similar situation did in the past.
3. Forecast model: This is a very popular model, and it works on anything with a numerical value based on learning from historical data. For example, in answering how much lettuce a restaurant should order next week or how many calls a customer support agent should be able to handle per day or week, the system looks back to historical data.
4. Outliers model: This model works by analyzing abnormal or outlying data points. For example, a bank might use an outlier model to identify fraud by asking whether a transaction is outside of the customer’s normal buying habits or whether an expense in a given category is normal or not. For example, a $1,000 credit card charge for a washer and dryer in the cardholder’s preferred big box store would not be alarming, but $1,000 spent on designer clothing in a location where the customer has never charged other items might be indicative of a breached account.
5. Time series model: This model evaluates a sequence of data points based on time. For example, the number of stroke patients admitted to the hospital in the last four months is used to predict how many patients the hospital might expect to admit next week, next month or the rest of the year. A single metric measured and compared over time is thus more meaningful than a simple average.

**Common Predictive Algorithms**

Predictive algorithms use one of two things: machine learning or deep learning. Both are subsets of artificial intelligence (AI). Machine learning (ML) involves structured data, such as spreadsheet or machine data. Deep learning (DL) deals with unstructured data such as video, audio, text, social media posts and images—essentially the stuff that humans communicate with that are not numbers or metric reads.

**Some of the more common predictive algorithms are:**

1. **Random Forest:**This algorithm is derived from a combination of decision trees, none of which are related, and can use both classification and regression to classify vast amounts of data.
2. **Generalized Linear Model (GLM) for Two Values:** This algorithm narrows down the list of variables to find “best fit.” It can work out [tipping points](https://www.merriam-webster.com/dictionary/tipping%20point) and [change data capture](https://www.hvr-software.com/blog/change-data-capture/) and other influences, such as [categorical predictors](https://online.stat.psu.edu/stat462/node/86/), to determine the “best fit” outcome, thereby overcoming drawbacks in other models, such as a regular linear regression
3. Gradient Boosted Model: This algorithm also uses several combined decision trees, but unlike Random Forest, the trees are related. It builds out one tree at a time, thus enabling the next tree to correct flaws in the previous tree. It’s often used in rankings, such as on search engine outputs.
4. K-Means: A popular and fast algorithm, K-Means groups data points by similarities and so is often used for the clustering model. It can quickly render things like personalized retail offers to individuals within a huge group, such as a million or more customers with a similar liking of lined red wool coats.
5. Prophet: This algorithm is used in time-series or forecast models for capacity planning, such as for inventory needs, sales quotas and resource allocations. It is highly flexible and can easily accommodate [heuristics](https://en.wikipedia.org/wiki/Heuristic) and an array of useful assumptions.

**9. The following data were collected when using a classification model to predict the malignancy of a group of patients’ tumors:**

**i. Accurate estimates – 15 cancerous, 75 benign**

**ii. Wrong predictions – 3 cancerous, 7 benign**

**Determine the model’s error rate, Kappa value, sensitivity, precision, and F-measure.**

**10. Make quick notes on:**

**1. The process of holding out**

**Holdout Method** is the simplest sort of method to evaluate a classifier. In this method, the data set (a collection of data items or examples) is separated into two sets, called the Training set and Test set.

A classifier performs function of assigning data items in a given collection to a target category or class.

Example –  
E-mails in our inbox being classified into spam and non-spam.

Classifier should be evaluated to find out, it’s accuracy, error rate, and error estimates. It can be done using various methods. One of most primitive methods in evaluation of classifier is **‘Holdout Method’**.

**2. Cross-validation by tenfold**

**K-fold Cross-Validation**

Cross-validation is usually used in machine learning for improving model prediction when we don’t have enough data to apply other more efficient methods like the 3-way split (train, validation and test) or using a holdout dataset. This is the reason why our dataset has only 100 data points. If you want to know more about the math behind this approach, I recommend reading [this](https://en.wikipedia.org/wiki/Cross-validation_(statistics)) article.

In k-fold cross-validation, we first shuffle our dataset so the order of the inputs and outputs are completely random. We do this step to make sure that our inputs are not biased in any way. Then, we split the dataset into *k* parts of equal sizes. In this analysis, we’ll use the 10-fold cross-validation. So, the first step is to shuffle and split our dataset into 10 folds.

With this method we have one data set which we divide randomly into 10 parts. We use 9 of those parts for training and reserve one tenth for testing. We repeat this procedure 10 times each time reserving a different tenth for testing.

**3. Adjusting the parameters**

**11. Define the following terms:**

**1. Purity vs. Silhouette width**

2. Boosting vs. Bagging

[Ensemble learning](https://www.geeksforgeeks.org/ensemble-classifier-data-mining/) helps improve machine learning results by combining several models. This approach allows the production of better predictive performance compared to a single model. Basic idea is to learn a set of classifiers (experts) and to allow them to vote. Bagging and Boosting are two types of Ensemble Learning. These two decrease the variance of a single estimate as they combine several estimates from different models. So the result may be a model with higher stability. Let’s understand these two terms in a glimpse.

1. Bagging: It is a homogeneous weak learners’ model that learns from each other independently in parallel and combines them for determining the model average.
2. Boosting: It is also a homogeneous weak learners’ model but works differently from Bagging. In this model, learners learn sequentially and adaptively to improve model predictions of a learning algorithm.
3. **The eager learner vs. the lazy learner**

A lazy learner delays abstracting from the data until it is asked to make a prediction while an eager learner abstracts away from the data during training and uses this abstraction to make predictions rather than directly compare queries with instances in the dataset.