32 64

Comment on the Pros and Cons of ML Techniques

All Sections

In this discussion assignment, I want you to comment on the advantages and disadvantages of the Machine Learning techniques (e.g., ANN, kNN, SVM, DT, RF, GBT, LR, Deep Learning, etc.), in your own words, based on your own knowledge and experiences.

First, post your own message, and then read and reply to others'. This assignment will be graded.

D. Delen

This topic was locked May 1 at 11:59pm.

Search entries or author | Unread | ✓ Subscribed

○

(https:/

Md Suman Ahammed (https://canvas.okstate.edu/courses/118118/users/157437)

Mar 31, 2022

We can understand trends and patterns of data easily with machine learning technique

Advantages and Disadvantages of Machine Learning technique:

Artificial Neural Networks:

ANN can be used with incomplete data. In contrast, ANN does not provide details of solution and no understanding of most important variables.

Decision Tree:

We do not need to handle missing data in decision tree, no need of normalization of data. On the other hand, small change in data will affect results greatly.

K Nearest Neighbours:

KNN does not need additional step for training data. Inversely, KNN is not good algorithm for large data set and KNN assumes equal importance for all variables.

Linear Regression:

After applying LR, it is easy to interpret the results. There are stepwise, forward and backward options to identify important variables. The cons of LR are it assumes linear relation between independent and dependent variables.

Random Forest:

RF reduces overfitting as it used ensemble technique which increase the accuracy of this model. RF can automatically handles missing values. On the other hand, RF creates a lot of tress that is a complex process comparing with decision tree.

○ (http [Moises Marin Martinez (https://canvas.okstate.edu/courses/118118/users/198182)](https://canvas.okstate.edu/courses/118118/users/198182)

Apr 28, 2022

Hi Md!

I agree with you, the lack of transparency of ANN is a reason why it wouldn't be preferred over other methods for some problems, specially banking where fairness in the decisions is important.

Not having to normalize data for a decision tree, that is reducing preparation work, is a benefit of this method.

○ (https:/ [Andrea Zerman (https://canvas.okstate.edu/courses/118118/users/214524)](https://canvas.okstate.edu/courses/118118/users/214524)

Apr 7, 2022

ANN (Artificial Neural Network)

Pros: used for regression and classification, good for nonlinear data, works better with large datasets with more variables.

Cons: Do not know how much each independent variable is influencing the outcome, time consuming to train the data, needs a lot of data.

In my experience so far, this hasn’t produced the best results and it has been a pain to train the data. However, I do look forward to using it more as I have heard about this technique most frequently in different texts I have read.

SVM (Support Vector Machine)

Pros: better for higher dimensions, classes are easily separable, outliers are less impactful, good for extreme case binary classification.

Cons: slower for larger data sets, not good for classes that overlap.

I do not have any experience using this technique.

LG (Logistic Regressions)

Pros: easy and effective to implement, input features do not need to be scaled, no need for tuning hyperparameters.

Cons: performs poorly for non-linear data and highly correlated features, requires identification of important variables.

Works best for binary classification

RF (Random Forest)

An ensemble of decision trees

Pros: decorrelates trees, reduces error by taking inputs from all trees, better performance on imbalanced data, more generalization. Deals well with large data, missing data, and outliers. Can help determine feature importance.

Cons: variables need predictive power, predictions need to be uncorrelated.

This has been my favorite technique to use. Not only does it provide insight into variable importance but it produced very accurate results.

DT (Decision Tree)

Pros: doesn't need normalization, not highly impacted by missing values, easy to visualize, automatically detects variables.

Cons: can overfit at times, outcomes can vary greatly if there are slight changes to the data, more time needed for decision tree training.

○ (http Sumanjali Etlam (https://canvas.okstate.edu/courses/118118/users/165323) ⋮

Apr 16, 2022

Hi Andrea,

Thankyou for sharing your insights on some of the most popular machine learning techniques. When it comes to machine learning, and especially classification, logistic regression is the most basic and extensively used method. The logistic regression approach is ideal for novices since it is simple to learn and execute. Logistic regression is a type of supervised learning that uses a logistic/sigmoid function to estimate probabilities and assess the association between a categorical dependent variable and one or more independent variables. Despite its name, logistic regression is not employed for regression problems in which the goal is to predict real-valued outcomes. It's a classification issue in which a collection of independent factors is utilized to predict a binary result (1/0, -1/1, True/False).

We may think of logistic regression as an extended linear model that is comparable to linear regression.

○

(http

Moises Marin Martinez (https://canvas.okstate.edu/courses/118118/users/198182)

Apr 30, 2022

Hi Andrea,

thanks for sharing this summary of pros and cons.

I agree with you about random forest method, it is also my favorite technique. I like how it easily takes us to an ensemble technique that builds many decision trees at once.

○

(https:/

Rudrakumar Ankaiyan (https://canvas.okstate.edu/courses/118118/users/190716)

Apr 8, 2022

Decision tree:

Pros: The very first machine learning algorithm that I was exposed to was the Decision tree Algorithm. I think the decision tree is one of the simple algorithms and easy to understand when we compare it with other algorithms. Also, it is not data specific. we can use it for any data type like categorical, numerical or boolean. It can be used for both classification and regression, as we all know. As a result, it can be used to forecast both discrete and continuous variables. We can work with the decision tree algorithm with minimal or fewer data.

Cons: When we look at the disadvantages of decision trees, we can say that the complexity increases with the size of the data. One of the main disadvantages of the decision tree algorithm is that it takes more time to train the model when compared to other algorithms. Any small change in data can create a big impact on the overall outcome of the prediction results. Thus, it's unstable when compared to other algorithms. Also, the decision tree algorithm is not preferred for very large datasets as it can take more time and can lead to more complexity.

Artificial Neural Network:
As we know, Artificial neural network is based on the model of the human brain in which neurons are the building blocks. One of the most advantages of ANN is its ability to produce results with incomplete data. The noise in the training data will not affect the performance of the model.

The cons of ANN are the requirement for more processing power due to its parallel processing nature. Also, it's very difficult to comprehend how we have got the solution. The data needs to be converted to numerical information before using it in ANN.

kNN (K Nearest Neighbours):

Pros: K Nearest Neighbours is much faster than other algorithms as it does not require a training period. This is because it makes real-time use of the training data while making predictions. In other algorithms, the addition of new data will lead to drastic changes in the output results. But, accuracy will not get impacted much in kNN even with the inclusion of new data.

Cons: kNN is a more sensitive algorithm as it can produce wrong predictions if the input dataset has outliers and missing values. Thus, we need to impute these values before generating predictions.

Support Vector Machine:

SVM uses hyperlanes to classify data into classes. The hyperlane acts as a boundary to make predictions based on the classification of data.

Pros: It can be used for both classification and regression problems. It can handle non-linear data using kernels. The processing power required for SVM is less as we mostly work with small datasets.

Cons: It is not preferable for large datasets. This is because it takes more time to train large datasets while using SVM. Its prediction gets affected by the presence of more noise in the dataset.

Random Forest:

Pros: Random Forest Algorithm is one of the most accurate and powerful models. It performs well with large datasets. It does not require scaling and normalization. The addition of new data will have less impact on the prediction outcome as the newly added data will not affect all the trees.

Cons: The prediction results produced are very difficult to interpret which is not in the case of decision trees. Since large datasets are used, the training period will also be longer as it will generate a huge number of trees.

Logistic Regression:

Pros: Logistic Regression is one of the simple machine learning algorithms. The training period for logistic regression is minimal when compared with other algorithms. It is very easy to perform and at the same time, it is very effective. The prediction results produced are highly interpretable.

Cons: It is not preferable for solving non-linear problems and cannot work with non-linear data like image data. Also, it is not preferred for solving complex problems. It requires more data and training to produce good predictions.

Thank you,

Rudrakumar Ankaiyan.

Edited by **Rudrakumar Ankaiyan (https://canvas.okstate.edu/courses/118118/users/190716)** on Apr 24 at 3:19pm

○ (http **Nithya Satheneni (https://canvas.okstate.edu/courses/118118/users/187996)** ⋮

Apr 22, 2022

Hi Rudra,

Thanks for sharing these great insights with us regarding the pros and cons of Machine Learning techniques. While working with KNIME tool, we do come across these models each having its own importance based on the given dataset. I like the way how you have explained each model in detail and type of data each model excels in. I would like to give an example of the logistic regression which I felt was very useful. When a transaction happens in a credit card, there are several factors that the system takes into account such as Date, type of purchase, time etc. Logistic Regression considers all these factors and lets us know whether a transaction is fraud. There are a lot of other real-time examples where such machine learning techniques are useful.

Thanks,

Nithya

○

(http

Thirumala Krishna Kurakula (https://canvas.okstate.edu/courses/118118/users/161132)

May 1, 2022

Hi Rudra,

Thanks for sharing this information. You mentioned that SVM classifies data using hyperplanes. These hyperplanes serve as a boundary for data categorization predictions. Because of this, minor data changes have little influence on the higher dimensional space. You also mentioned that the new data has less impact on the prediction result (decision tree). While this is a benefit, it also has drawbacks.

○

(https:/

Cassie Adams (https://canvas.okstate.edu/courses/118118/users/196408)

Apr 10, 2022

My knowledge and experience with machine learning techniques is limited to what I am learning in this class, so I'm eager to see the input others have on this topic.

Decision Trees make the most sense to me, and I think this is one of the advantages. They don't take much effort to implement, and the results are easy to understand since you can see each decision variable and the predictive outcomes. One disadvantage is that they don't handle missing data or large variances very well, which can cause over-fitting.

To me, a Random Forrest is an enhanced Decision Tree with better prediction capability. Similar to Decision Trees, they are fairly easy to implement, but the accuracy is much better. They handle missing and highly variant data better, thus avoiding overfitting. The main disadvantages are that they are harder to understand and can be slower due to increased computation needs.

KNN is fairly easy to implement and doesn't require training, making it faster than other options. It doesn't work well with large datasets and is sensitive to noisy data, so outliers need to be removed and missing data needs to be imputed or the prediction accuracy can be significantly affected.

I don't understand SVM as well, other than it works well for small datasets where you might have more variables than data points. The disadvantages are that it can lean towards over-fitting and wouldn't work well for large data sets because the training period would take too long.

ANNs seem to have a lot of strengths, such as being able to handle incomplete information and it works much like the human brain. The main disadvantages seem to be that it is difficult to setup correctly and difficult to understand how it works and how it comes to a decision. For me, this makes me kind of avoid using it because I really don't understand it.

To me, Linear Regression seems very similar to Decision Trees. It has similar advantages and disadvantages, but LR can model both continuous and categorical data, whereas Decision Trees can really only model categorically. Also, LR is easy to understand, but Decision Trees seem even easier due to their graphical representation.

GB seems similar to Random Forrest learning, but it can be used for numerical in addition to categorical data. Also, GB learns from mistakes in iterations, whereas RF does not. However, it can take longer to train because it cannot be parallelized like RF, and it is more likely than RF to overfit.

I'm looking forward to seeing input from others as these concepts are all fairly new to me.

Edited by **Cassie Adams (https://canvas.okstate.edu/courses/118118/users/196408)** on Apr 10 at 10:50pm

○ (http **Paul Davis (https://canvas.okstate.edu/courses/118118/users/194957)**

Apr 23, 2022

Cassie - I really agree that this has been fun to learn about models in this class, and it is also the extent of my experience with this too. Because of that it is so much easier to understand the models that are much more transparent with how they work such as DT and RF. You can see the results and the process and it really helps reinforce the concepts while I’m using it, versus some of the other models where you really need to understand the concepts and the rules prior to using them as setting them up in KNIME doesn’t necessarily reinforce the understanding of them.

○ (http **Jacob Wood (https://canvas.okstate.edu/courses/118118/users/214790)**

Apr 30, 2022

Hi Cassie,

I have the same experience level as you. Your point about the ANNs and the inability to understand them is a good one. I struggle with this too, but suspect further use will make us more comfortable with the topic. However, in real world uses if there are stakeholders new to ANNs, they will likely have similar reactions to ANNs. We will have to remember our thoughts from the first exposure we've had to ANNs in this course.

○

(https:/

Jay West (https://canvas.okstate.edu/courses/118118/users/57886)

Apr 10, 2022

Random Forest

I believe that the advantage of using Random Forest is to predict better accuracy than single decision trees. Due to the random sample of the training data, it reduces the over-fitting problem in decision trees and their variance, and thus, improves accuracy. Another advantage of Random Forest is that it works well for both categorical and continuous variables. Lastly, missing values are automatically handled by Random Forest.

A disadvantage of using Random Forest is its complexity. Due to the creation of multiple trees, it may suffer interpretability and also requires longer time to train than decision trees to determine the class. In addition, as Random Forest generates numerous trees to integrate their outcomes, it requires a substantial amount of computing power and resources.

Artificial Neural Network

An advantage of Artificial Neural Network is the capability of detecting nonlinear relationships between dependent and independent variables. It is also good for numerical information.

A disadvantages of Artificial Neural Network are that it requires a lot of data, complexity of computation, and difficulty of presenting results/outcomes.

Decision Tree

An advantage of Decision Tree is that it is easy and efficient to interpret data. Another advantage is the flexibility of handling both continuous and categorical variables. Lastly, it does not require much time for data preparation. In general, it is great for data exploration.

A disadvantage of Decision Tree is that if there are a small changes in data that could lead to a large changes in the structure of the model thus, leading to instability and inaccuracy in the output. In addition, if the size of data is large, a single tree can create a large number of nodes, resulting in complexity and over-fitting.

Linear Regression

An advantage of linear regression is that it is also easy to understand and interpret. It works great for linearly separable data.

A disadvantage of linear regression is that if data is poorly prepared, it does not perform well as it does not recognize and handle missing values, redundant data or outliers. Also, it does not work well with non-linear relationship.

Support Vector Machine

An advantage of Support Vector Machine is that when the number of dimensions exceeds the number of samples, support vector machine is efficient and it performs well when compared to other machine learning techniques.

The downside of the Support Vector Machine is its inability to handle large data sets and its poor performance when the data set contains noise.

Edited by Jay West (https://canvas.okstate.edu/courses/118118/users/57886) on Apr 11 at 6:51pm

○

(http

Rudrakumar Ankaiyan (https://canvas.okstate.edu/courses/118118/users/190716)

Apr 29, 2022

Hi Jay,

Thank you for sharing these valuable insights about the advantages and disadvantages of Machine Learning methods with us. We come across these models while working with the KNIME tool, each with its unique relevance dependent on the dataset. I loved how you described each model in detail and the kind of data that each model excelled at.

○

(https:/

Bhanu Teja Pulipalupula (https://canvas.okstate.edu/courses/118118/users/159029)

Apr 12, 2022

Hello everyone,

SVM:

Pros of SVM: SVM performs better when the number of features/columns is high. It is considered the best algorithm when classes are separable. SVM is most suitable for binary classification. Cons: Takes a large amount of time to process. The performance of SVM is comparatively low in the case of overlapped classes.

kNN:

Pros of kNN: kNN is a constantly evolving model, that is, it changes accordingly with new data fed to the model. Multi-class problems can also be resolved. Cons: kNN performance is relatively slower for large datasets. kNN cannot deal better with missing values. kNN does not work better on imbalanced data sets.

Naive Bayes:

Pros of Naive Bayes: It is insensitive to irrelevant features, and gives fast and real-time predictions. Multi-class prediction is effectively done, Naive Bayes gives good performance with high dimensional data. Cons: If the training data has no occurrences of a class and a certain attribute/column together, Naïve Bayes does not work well.

Decision Tree:

Pros of Decision Tree: Feature selections is automatic and irrelevant features won't affect decision trees. It's comparatively easier to explain to business teams with minimal technical knowledge. Decision trees handle missing values better than other techniques. Scaling or normalization of data is not required while using decision trees.

Cons: Decision Trees are sensitive to data, i.e., if data changes slightly, the outcomes can change to a very large extent. It's time-consuming, as training decision trees take a longer time.

Random Forest:

Pros of Random Forest: Random forest is an association of decision trees, that takes inputs from all the trees, and predicts the target variable, this makes sure that the individual errors of trees are minimized and overall variance, error is low. It can handle huge datasets and handles missing data very well. Outliers do not have a high impact on Random forest. As Random Forest considers only a subset of features, final outcome depends on all the trees, and there's no problem of overfitting.

Cons: Selected features should have some predictive power else it doesn't work. Random forest often appears to be a black box, i.e., it is tough to know what is happening, Trial and error with different parameters, random seeds can be used to check performance and predictions.

Logistic Regression:

Pros of Logistic Regression: This is simple to implement and effective, normalization of data is not mandatory but works with normalized data too.

Cons: Gives poor performance with irrelevant features and in the case of non-linear data (image data for example). It relies more on the presentation of the proper data, i.e., all important variables should be identified for it to work well.

Deep Learning:

Pros of Deep Learning: The deep learning architecture is adaptable, which means it may be used to solve new challenges in the future. Features are automatically inferred and modified to get the desired result. It is not necessary to extract features ahead of time. It provides superior performance outcomes while dealing with large amounts of data.

Cons: To perform better than other models, it needs a huge amount of data. Very expensive to train in case of complex data models, needs more computing power which increases the need for more efficient and expensive hardware.

ANN:

Pros of ANN: Has parallel processing capabilities, i.e., it can perform more than one job at the same time giving quick results. ANNs have the ability to make machine learning, learn from comparable occurrences, and make predictions based on them.

Cons: Highly dependent on hardware as it requires more processing power for parallel processing. When ANN gives a prediction, it doesn't explain the why and how of the result.

Overall, if the dataset is medium-sized and there are too many features, SVM is a good choice. If there is a linear relationship between the dependent and independent variables/columns, then linear/logistic regression and SVM are good. When there are features that are highly independent, Naïve Bayes will work great. If a dataset is small and we don't know the relationship between dependent and independent variables, kNN is preferred. Most importantly, for a given business problem, before we choose any machine learning algorithm, Data Understanding is very critical.

Edited by Bhanu Teja Pulipalupula (https://canvas.okstate.edu/courses/118118/users/159029) on Apr 23 at 1:59am

○ (http

Anudeep Nare (https://canvas.okstate.edu/courses/118118/users/188001)

Apr 17, 2022

Hello bhanu

Thank you for taking the time to provide your thoughts on some of the most popular machine learning approaches. When it comes to machine learning, particularly classification, logistic regression is the most fundamental and widely used approach. Despite its name, logistic regression is not used to predict real-valued outcomes in regression situations. Logistic regression may be thought of as an extended linear model similar to linear regression. Thank you for the valuable information

○ (http

Josh Basquez (https://canvas.okstate.edu/courses/118118/users/158078)

Apr 29, 2022

Thank you for sharing your pros and cons of ANNs. I find that the ANN seems to be the most powerful implementation of major learning as it builds on the human like predictions based on learning some multiplying those calculations in parallel computations and predictions , however from a data scientist perspective it could also be frustrating without having full indict into the decision making process and calculations .

○ (http

Srikanth Daruru (https://canvas.okstate.edu/courses/118118/users/193729)

May 1, 2022

Hello Bhanu

Thank you for sharing these valuable insights into the advantages and disadvantages of Machine Learning techniques with us. We come across these models while working with the KNIME tool, each with its unique importance based on the dataset. I loved how you described each model in detail and the kind of data that each model excelled at. I noticed that When the number of features for each data point exceeds the number of training data samples, then SVM will definitely performs poor. But as you said Linear regression and SVM are useful when the dependent and independent variables have a linear relationship.

○ (https:/

Anudeep Nare (https://canvas.okstate.edu/courses/118118/users/188001)

Apr 13, 2022

Hello class,

I would like to share about few topics have a look into it,

Machine Learning is a technology that has exploded in popularity and utilization in recent years. A large number of aspirants from all over the world are rapidly learning this technology and using what they've learned in a variety of ways.

Machine Learning algorithms may learn from the information we supply. With each new piece of data added, the model's accuracy and efficiency in making judgments improves. Every day, companies like Amazon, Walmart, and others acquire a massive amount of fresh data. With such a large amount of training data, the accuracy of locating related goods or using a recommendation engine improves.

Support Vector Machine:

Pros :

When there is a clear margin of distinction between classes, SVM performs effectively. In high-dimensional spaces, SVM is more effective. When the number of dimensions is more than the number of samples, SVM is successful. SVM uses a little amount of memory. For big data sets, the SVM method is ineffective.

Cons:

When the data set contains additional noise, such as overlapping target classes, SVM does not perform well. The SVM will underperform if the number of features for each data point exceeds the number of training data samples. There is no probabilistic justification for the classification because the support vector classifier operates by placing data points above and below the classifying hyperplane.

Deep Learning :

Deep Learning is one of the machine learning technique requires very large amount of data in order to perform better than other techniques. It is extremely expensive to train due to complex data models. There is no standard theory to guide you in selecting right deep learning tools as it requires knowledge of topology, training method and other parameters.

Pros:

Improved Self-Learning Capabilities while Working with Unstructured Data
Ability to offer high-quality outcomes without the need for feature engineering

Cons:

It needs a big amount of data in order to outperform other strategies
Due to the complexity of the data models, training is quite costly.
Furthermore, deep learning necessitates the use of pricey GPUs and hundreds of workstations. This raises the cost to the users.

Logistic regression :

Logistic regression is a statistical analysis approach that uses independent characteristics to try to predict precise probability outcomes. On high-dimensional datasets, this may cause the model to be over-fit on the training set, overstating the accuracy of predictions on the training set, and so preventing the model from accurately predicting outcomes on the test set. This is most common when the model is trained on a little amount of training data with a large number of features. Regularization strategies should be explored on high-dimensional datasets to minimize over-fitting (but this makes the model complex). The model may be under-fit on the training data if the regularization variables are too high.

Pros:

Logistic regression is less difficult to apply, analyze, and train.
Logistic regression is a statistical analysis approach that predicts a binary result, such as yes or no, based on past data set observations.

cons:

The assumption of linearity between the dependent and independent variables is the primary restriction of Logistic Regression.

logistic If the number of data is smaller than the number of features, regression should not be utilized; otherwise, overfitting may occur.

Decision tree:

Both classification and regression issues may be solved with decision trees: Decision trees are effective in both regression and classification applications because they can predict both continuous and discrete variables. Because decision trees are straightforward, they need less effort to comprehend. The time complexity of doing this process grows exponentially as the number of records grows. Training a decision tree with numerical variables takes a long time.

Pros:

Decision trees need less work for data preparation during which was before than other methods.
A decision tree does not need data normalization.
A decision tree does not need data scalability.

In addition, missing values in the data have no significant impact on the decision tree-building process.

A decision tree model is simple to understand and convey to technical teams and stakeholders.

Cons:

A slight change in the data can result in a substantial change in the decision tree's structure, resulting in instability.
When compared to other algorithms, a decision tree's calculation might get rather complicated at times.
The training period for a decision tree is often longer.
Because of the intricacy and time required, decision tree training is relatively costly.
When it comes to using regression and predicting continuous values, the Decision Tree method falls short.

Random Forest:

Pros :

Random Forest may be used to address issues in both classification and regression.
Both categorical and continuous variables function well with Random Forest.
Missing values may be handled automatically using Random Forest. Unlike curve-based algorithms, non linear parameters have no effect on the performance of a Random Forest. As a result, if the independent variables are very nonlinear, Random Forest may outperform conventional curve-based methods. Random Forest generates a large number of trees and aggregates their results.

Cons:

Complexity: Random Forest generates a large number of trees (as opposed to a single tree in a decision tree) and then mixes their results. In the Python sklearn package, it builds 100 trees by default. This approach necessitates a significant increase in processing power and resources. On the other hand, a decision tree is straightforward and does not need a large amount of computer power.

Longer Training Period: Random Forest takes significantly longer to train than decision trees since it creates several trees (instead of just one) and makes decisions based on the majority of votes.

Artificial Neural Networks :

Pros :

The output of ANNs can be discrete-valued, real-valued, or a vector of many real or discrete-valued characteristics, while the target function can be discrete-valued, real-valued, or a vector of numerous real or discrete-valued attributes.
Noise in the training data is not a problem for ANN learning algorithms. There may be faults in the training samples, but they will have no effect on the final result.
It's employed when a quick assessment of the learnt target function is necessary.

Cons :

Artificial Neural Networks are hardware-dependent due to their structure, which necessitates parallel processing power.
As a result, the equipment's manifestation is contingent.
The most serious issue with ANN is the network's unexplained behavior.
When ANN provides a probing answer, it does not explain why or how it was chosen.
The network's trust is eroded as a result of this.
Assurance of correct network structure: The construction of artificial neural networks is not determined by any precise rule.
Experience and trial and error are used to create the ideal network structure.

Thank you,

Anudeep Reddy Nare

Edited by Anudeep Nare (https://canvas.okstate.edu/courses/118118/users/188001) on May 1 at 11:54pm

○ (http

Bhanu Teja Pulipalupula (https://canvas.okstate.edu/courses/118118/users/159029)

Apr 29, 2022

Hello Anudeep, Thank you for sharing your thoughts about these machine learning techniques. Big-Tech companies like Walmart, amazon have streaming APIs that allow continuous data inflow to their databases. Read more about this here: **https://aws.amazon.com/streaming-data/** (https://aws.amazon.com/streaming-data/) . On your comments on the ml techniques, Yes, SVM takes forever if the dataset is medium to large-sized. I've experienced this even for the term project data with just over 40K records. Random forest is an enhanced version of decision trees and has always given higher accuracy, AUC for all the knime data mining assignments this semester. I completely agree that ANN handles noisy data very well and in fact, adding noise to the training process while using ANN can improve its robustness and reduce generalization error, read more on how to train neural networks to reduce overfitting here: **https://machinelearningmastery.com/train-neural-networks-with-noise-to-reduce-overfitting/** (https://machinelearningmastery.com/train-neural-networks-with-noise-to-reduce-overfitting/) .

○ (http **Rithik Ponugoti** (https://canvas.okstate.edu/courses/118118/users/189064)

Apr 29, 2022

Hi Anudeep,

Thank you for your detailed information. I appreciate your thorough understanding of the ML applications and their use cases. In our project, we have used various ML techniques like Random Forest, Decision Trees, SVM, etc. While using SVM, we had to filter out the Categorical Variables as it was not taking them as input. However, we used them because they have the ability to model non-linear decision boundaries, and there are several kernels from which to pick. They're also resistant to overfitting, particularly in high-dimensional space. I noticed that you have mentioned underperforming of SVM if the number of features for each data point exceeds the number of training data samples. That's a great point!

I hope you continue growing in the field and wish you the best in your future endeavors. See you around :)

Best Regards,

Rithik Sai Ponugoti.

○ (https:/ **Shirley She** (https://canvas.okstate.edu/courses/118118/users/133072)

Apr 13, 2022

SVM (Support Vector Machine)

Pros:1. Performs well in Higher dimension 2. Outliers have less impact.3.SVM is suited for extreme case binary classification.

Cons:1. Poor performance with Overlapped classes 2. Selecting the appropriate kernel function can be tricky.

Logistic Regression

Pros: Simple to implement

Cons:1, Poor performance on non-linear data 2. Poor performance with irrelevant and highly correlated features 3. Not a very powerful algorithm and can be easily outperformed by other algorithms.

Random Forest

Pros:1. Good Performance on Imbalanced datasets: 2. Handling of the huge amount of data: 3 Good handling of missing data: 4. Little impact of outliers.5. Useful to extract feature importance (we can use it for feature selection)

Cons: Appears as Black Box: It is tough to know what is happening. You can at best try different parameters and random seeds to change the outcomes and performance.

Decision Trees

Pros:

1. Normalization or scaling of data is not needed.
2. Handling missing values: No considerable impact of missing values.
3. Easy to explain to non-technical team members.
4. Easy visualization
5. AutomaticFeature selection: Irrelevant features won’t affect decision trees.

Cons:1.Prone to overfitting.2. Sensitive to data. If data changes slightly, the outcomes can change to a very large extent.

k-NN (K Nearest Neighbors)

Pros:1. Simple to understand and implement 2. No assumption about data.3. Constantly evolving model: 4. Multi-class problems can also be solved.

Cons:1. Slow for large datasets.2. Curse of dimensionality: This does not work very well on datasets with a large number of features.3. Does not work well on Imbalanced data.4. Sensitive to outliers.

(https:/

Paul Dreyer (https://canvas.okstate.edu/courses/118118/users/183234)

Apr 13, 2022

While I think there are multiple advantages to machine learning, I think the biggest advantage is the ability to look at and interpret data in whole new ways that were undetected before. In the healthcare industry, we are able to look at the billions of claim lines submitted for members and pull out patterns of billing by providers that show that they are clear outliers within the data. The biggest example we have of this now is what's called "episode grouping", which is exactly what it sounds like. A patient has a medical episode (such as a broken ankle) that marks the beginning of the episode. We then compile data based on how all broken ankles are treated (i.e. goes to surgery, therapy, length of therapy, number of therapy sessions, Medical equipment associated with the break, etc.) and we are then able to determine outliers within that data, like say a provider that is providing wheelchairs to all of his broken ankle patients, but other patients are getting crutches or walkers (wheelchairs are much more expensive in this case, which means the provider gets a higher reimbursement). Without machine learning, we would not be able to use this new technique, and we would still be sifting through the billions of claims lines trying to determine outliers in a new way.

A disadvantage to machine learning techniques is the lack of general understanding of such techniques outside of the data science community. While some are more self-explanatory and easier to understand than others (like decision trees and linear regression), models like SVM's require a much more extensive explanation for anyone outside of data science to use. Communicating what the models are really doing in layman's terms is one of the biggest obstacles I have had with ML. I know if our clients can't understand something, they will most likely not use that product or model because they would not be able to explain their own findings to someone else. Figuring out the best way to explain what a model does and what results it is producing has been my biggest challenge

(http

Paul Dreyer (https://canvas.okstate.edu/courses/118118/users/183234)

Apr 20, 2022

I'd like to supplement a little bit here to help relate it more to machine learning; we focus a lot on decision tree analysis; however we treat service and diagnosis codes as our "variables". So, an example would be (and keeping with the one I mentioned above), did the patient break an ankle? Yes or no? Then we move to the next level of "did they receive a cast?". Then we use an avg length between services to find how long the avg cast was on, and then use that as our "break point" (i.e. did the person have the cast on for a shorter or longer time?). We continue down this path throughout the entire "episode" and we are able to see the path that is taken the most by the highest number of providers. After performing that analysis, we are able to sift through and find those providers that "broke off" the path in terms of either the most claims, most patients, or highest $$ paid. This helps us find our outliers and begin our focus on investigating those providers further

○

(https:/

Ben Lewis (https://canvas.okstate.edu/courses/118118/users/211833)

Apr 13, 2022

Hi all,

Here are my pros and cons of different machine learning algorithms.

Artificial Neural Networks –

Pros - ANN can handle really complex problems! A lot of the leaders on machine learning competitions like Kaggle use neural networks to come up with their prize winning solutions. Neural Networks are great for complex problems, cutting through really noisy data, and areas where pattern identification can lead to a solved problem.

Cons -One of the big cons is the training time and resources needed, and how its sometimes hard to interpret. For example, you might get a solution, but not be able to explain how you got there!

K-nearest neighbors-

Pros – Unlike ANN, it’s pretty simple to understand the concepts underneath it. It can be used for both regression and classification, and is pretty fast to implement.

Cons – overfitting the data is a concern, and you have to make sure your dataset is balanced or else your prediction will not be accurate. Computation time might be a concern with really large datasets. Overall I think it’s a good algorithm, especially in classification problems, but is most effective when you have high quality data.

Support Vector Machines-

Pros – I’m still hazy on the mathematics of this one, but my understanding is that SVM is great for use cases where there are multiple dimensions to the data, or with a large amount of data set. It is effective at using a hyperplane to separate the data.

Cons – If a lot of the data is overlapped or not separated the SVM won’t perform as well. It also can be computationally heavy. Additionally, the math for me is harder to understand compared to other algorithms on this list.

Decision Trees

Pros – My favorite thing about decision trees is how easy it is to understand and interpret. I feel like it, along with linear regression, most closely mimics basic human deduction. The Decision tree is able to do it at a much more complicated level. Another really great pro is how it handles feature selection, and how it gives you a variable importance table.

Cons – Overfitting is again a concern. It can also take time to compute if there is a lot of features in the tree.

Random Forest –

Pros – Similar to the decision tree, it outputs the importance of features which is really powerful. One advantage it has over decision trees is that it is less likely to overfit data. It can also handle imbalanced data well.

Cons – Like ANN it is hard to fully understand what is being performed by the algorithm. There also needs to be no correlation in the prediction of the trees.

Gradient Boosting –

Pros – This is a newer one for me, but I know it is very popular, especially XGBoost. From my research, the biggest pro is that it curbs overfitting, reduces bias and variance.

Cons – From research, the biggest disadvantages include its scalability and it’s ability to handle outliers.

Linear Regression-

Pros– For me, the biggest pro is the ease of use for the algorithm. It was the first machine learning algorithm I studied, and I think would be simple enough a boardroom or stakeholders with a non-technical background would understand it’s function. It is also fast.

Cons – It is not as versatile as other algorithms on this list, and needs the data to be linear. Additionally, it needs strong data. Noisy data and outliers are both risks when using a Linear Regression.

Logistic Regression –

Pros – Similar to Linear Regression, the logistic regression is simple to understand, quick, and efficient. It is great with classification, and can give really accurate predictions despite being so simple.

Cons – It is sill based on linear problem solving, which not all data will fit. It also works better with a high number of observations, and struggles if there is not a lot of data for it to train on.

Deep Learning-

Pros – The model is learning from the data automatically, and then optimized. It can handle really large datasets and tackle very large problems.

Cons – A lot of the Deep Learning being done is experimental in nature. It doesn’t have strict guidelines. Additionally, it requires large amounts of data to be successful and is very demanding on resources.

○ (http **Fayaz Shaik (https://canvas.okstate.edu/courses/118118/users/190988)**

May 1, 2022

Thank you Ben for your informative post on all the ML techniques!

○ (http **Thirumala Krishna Kurakula (https://canvas.okstate.edu/courses/118118/users/161132)**

May 1, 2022

Hi Ben,

Thanks for sharing this information. This is the first time I've heard of gradient boosted trees. I looked through this method and discovered some useful information. Gradient boosting may easily overfit a training dataset. It may benefit from linearization approaches that punish different sections of the procedure and reduce the computational burden. Gradient boosting uses decision trees as a weak learner. Overfitting the training set might reduce the model's generalization capabilities.

○

(https:/

[Sumanjali Etlam (https://canvas.okstate.edu/courses/118118/users/165323)](https://canvas.okstate.edu/courses/118118/users/165323)

Apr 16, 2022

Hello Everyone,

The area of machine learning as a whole is incredibly broad and profound. I was fascinated with reading about theoretical limitations of machine learning throughout my first few years at university - error boundaries and computational complexity were endlessly intriguing to me. Machine learning is a data analytics approach that trains computers to learn from experience in the same way that people and animals do. Machine learning algorithms employ computer approaches to "learn" information directly from data rather than depending on a model based on a preconceived equation.

Some of the machine learning techniques advantages and disadvantages:

1.ANN: Artificial Neural Networks (ANNs) or neural networks are computational methods that aim to model the behavior of biological systems made up of neurons. ANNs are computer models based on the central nervous system of animals. It is capable of pattern recognition as well as machine learning techniques.

Pros:

- Attribute-value pairs are used to represent problems in ANN.
- The output of ANNs can be discrete-valued, real-valued, or a vector of many real or discrete-valued characteristics, while the target function can be discrete-valued, real-valued, or a vector of numerous real or discrete-valued attributes.
- Noise in the training data is not a problem for ANN learning algorithms. There may be faults in the training samples, but they will have no effect on the final result.
- The amount of weights in the network, the number of training instances evaluated, and the settings of different learning algorithm parameters can all contribute to extended training durations for ANNs.

Cons:

- The construction of Artificial Neural Networks necessitates parallel processing power. As a result, the equipment's manifestation is contingent.
- When the network's error on the sample is lowered to a specific value, the training is complete. The value does not provide us with the best outcomes.
- This is the most critical issue with ANN. When ANN provides a probing answer, it does not explain why or how it was chosen. The network's trust is eroded as a result of this.

2.KNN: The KNN method is a relatively basic classification technique. KNN is an abbreviation for K-Nearest Neighbors. In KNN, K is the number of neighbors. Let's look at some of the benefits and drawbacks of the KNN algorithm.

Pros:

- Because the KNN algorithm does not require any training before generating predictions, fresh data may be supplied without affecting the system's accuracy.
- KNN is a simple algorithm to use. KNN may be implemented with only two parameters: the value of K and the distance function (e.g. Euclidean or Manhattan etc.)
- Lazy Learner is the name given to KNN (Instance based learning). During the training phase, it does not learn anything. The training data isn't used to construct any discriminative functions. In other words, it does not require any training. It saves the training dataset and uses it only when generating real-time predictions to learn from it. This makes the KNN method significantly quicker than other training-based algorithms like SVM and Linear Regression.

Cons:

- The cost of computing the distance between the new point and each old point is enormous in big datasets, which lowers the algorithm's speed.
- The KNN algorithm does not operate well with high-dimensional data because calculating the distance in each dimension becomes challenging for the algorithm with a large number of dimensions.
- Before applying the KNN algorithm to any dataset, feature scaling (standardization and normalization) is required. If we don't, KNN may make incorrect predictions.
- The KNN algorithm is susceptible to dataset noise. Missing values must be manually imputed, and outliers must be removed.

3.SVM: The data is classified using SVM (Support Vector Machine) utilizing a hyperplane, which functions as a decision border between distinct classes. Support Vectors are extreme data points from each class. SVM seeks to identify the best and most optimum hyperplane for each Support Vector that has the most margin.

Pros:

- When there is a clear margin of distinction between classes, SVM performs effectively.
- In high-dimensional spaces, SVM is more effective.
- When the number of dimensions is more than the number of samples, SVM is successful.
- SVM uses a little amount of memory.

Cons:

- For big data sets, the SVM method is ineffective.
- When the data set contains additional noise, such as overlapping target classes, SVM does not perform well.

- The SVM will underperform if the number of features for each data point exceeds the number of training data samples.
- There is no probabilistic justification for the classification because the support vector classifier operates by placing data points above and below the classifying hyperplane.

4.DT: A prominent machine learning algorithm is Decision Tree. By translating data into a tree representation, Decision Tree tackles the challenge of machine learning. Each attribute is represented by an internal node in the tree representation, and each class label is represented by a leaf node.

Pros:

- Decision trees need less work for data preparation during pre-processing than other methods.
- A decision tree does not need data normalization.
- A decision tree does not need data scalability.
- In addition, missing values in the data have no significant impact on the decision tree-building process.
- A decision tree model is simple to understand and communicate to technical teams and stakeholders.

Cons:

- A slight change in the data can result in a substantial change in the decision tree's structure, resulting in instability.
- When compared to other algorithms, a decision tree's calculation might get rather complicated at times.
- The training period for a decision tree is often longer.
- Because of the intricacy and time required, decision tree training is relatively costly.
- When it comes to using regression and predicting continuous values, the Decision Tree method falls short.

5.DEEP LEARNING: Deep learning is a machine learning approach that allows computers to learn by example in the same way that people do. Deep learning is a critical component of self-driving automobiles, allowing them to detect a stop sign or discriminate between a pedestrian and a lamppost.

Pros:

- Inside a Deep Learning model, feature engineering can be done automatically.
- Ability to tackle difficult issues while being flexible enough to adapt to new challenges in the future (or transfer learning can be easily applied)
- High levels of automation. Users may design a deep learning model in seconds using a deep learning library (Tensor flow, Keras, or MATLAB.). (without the need of deep understanding)

Cons:

- A large volume of data is required.
- Overfitting is an expensive and intense training method that may be used to solve simple difficulties.
- There is no standard for model training and tweaking.
- Inside each layer, it's a black box that's difficult to comprehend.

I am looking forward to seeing your thoughts and interested to know your responses!!

○ (http [Paul Dreyer (https://canvas.okstate.edu/courses/118118/users/183234)](https://canvas.okstate.edu/courses/118118/users/183234)

Apr 18, 2022

Hi Sumanjali,

I really like how detailed your pros and cons are. I agree that ML is very broad and I do think it's very complex. Picking the right model isn't always easy, but I have learned that there are some neat additions to KNIME that really assist with a user's ability to select the right model (the H2O packages in particular). In my current profession, we have actually started utilizing Deep learning and we are finding the cons to be fixed, albeit with some hard work. We have billions of rows available, so the biggest obstacle (IMO) of the large volume of data is knocked out right away. After that, we are able to tweak our models significantly until we are able to see the results that we expect. We also use KNN models a lot, but instead of removing the outliers, we are able to batch them together because the outliers are exactly what we are looking for. The negatives we have found definitely coincide with what you mentioned, in that we have to cut down significantly on our dimensions within the data.

All in all, I think your post sums up these models very well, and is actually a great reference for anyone who is starting out in ML!

○ (https:/ [Anurag Budme (https://canvas.okstate.edu/courses/118118/users/155819)](https://canvas.okstate.edu/courses/118118/users/155819)

Apr 16, 2022

Pros and Cons of ML techniques

SVM

The unsupervised model is used for classification use cases.

Pros:

- Works well in the case of high-dimensional data.
- Less impact of outliers on classification data.

Cons:

- Large execution time.
- Poor performance when the data belongs to multiple classes.

Application: Classification of images (nonlinear data), medical analytics, speech recognition

Naïve Bayes

The supervised model used for classification

Pros:

- Fast and can be used in real-time.
- Works well with large datasets.
- Works well with multi-class prediction.

Cons:

- Data should not be imbalanced.
- The independence of features does not hold.

Decision Tree:
Used for regression and data classification
Pros:

- Less effort requires for data cleaning/pre-processing
- It does not require data scaling/normalization.
- Very Intuitive and easy to explain to non-tech stakeholders.
- Have ensemble methods (bagging & boosting)

Cons:

- Inadequate for applying regression and predicting continuous variables.
- Changes in data can change the structure of the decision tree creating instability in decision making.
- Time-Consuming in model building.

Logistic Regression

Pros:

- simple to understand, quick, and efficient.
- Works well on datasets that have linearity.
- Interpret model coefficients as indicators of feature importance.

Cons:

- Did not consider all columns, since it is based on linear problem solving, in which not all data will fit. It also works better with a high number of observations and struggles if there is not a lot of data for it to train on.
- Assumption of linearity between the dependent variable and the independent variables.

Deep Learning

Pros:

- Can be applied to complex non-linear problems.
- Works well on large input data.

Cons:

- It is not known to what extent each independent variable is affected by the dependent variable.
- Computation is difficult and time-consuming.
- Model functioning depends on the quality of the training.

Edited by Anurag Budme (https://canvas.okstate.edu/courses/118118/users/155819) on Apr 28 at 8:25pm

○ (https:/

Tejaswi Maruthi (https://canvas.okstate.edu/courses/118118/users/188246)

Apr 17, 2022

Advantages and disadvantages of the Machine Learning techniques

ANN:

Advantages: ANN's are powerful and good to classify images. Robust to noise in training. ANN's can handle long training times. Ann's can parallel process and can work with incomplete information. Information is stored on entire network not on a database.

Disadvantages: ANNs are limited by insufficient training data. Although parallel processing is an advantage of ANN, due to this very feature of parallel processing its highly hardware dependent. The main disadvantage of ANN's is we cannot explain the behaviour of network, the why & how of the solution.

KNN:

Advantages: Very easy to implement as we only need k and the distance function. KNN performs well when sample size is less than 100K records, for non-textual data. KNN doesn't require training (it's called Lazy Learner), so new data can be added not impacting the accuracy.

Disadvantages: Since entire data is processed for every prediction it doesn't work well for large datasets. Doesn't work well for high dimensions as calculating distance for each dimension becomes difficult. KNN is sensitive to noise, missing values and outliers. Scaling should be done properly.

SVM:

Advantages: Good when we have no idea on data, good for unstructured, semi- structured data especially text classification, works well for high dimensional and non-linear data, has good generalization capabilities so risk of overfitting is less, unlike neural nets SVM is not solved for local optima. SVM model is more stable and small change in data doesn't affect the hyperplane. SVM works better and gives better accuracy than KNN with small train sets and are easier to train than ANN's.

Disadvantages: Extensive memory requirement and high algorithmic complexity. Difficult to interpret final model. Long training time for huge datasets. In creating too many support vectors to handle high dimensional data, the training speed reduces.

Decision trees:

Advantages: Most useful in data exploration, good for categorical classification. DT's aren't effected much by outliers. Can be used to capture non-linear relations. Fast and efficient compared to other classification algorithms like KNN. No feature scaling, normalization is required. Overall easy to interpret and visualize.

Disadvantages: If we don't prune, it leads to overfitting. Bagging and boosting is required to handle the variance in decision trees. For large data sets the tree gets complex.

Random Forest:

Advantages: RF's are easy to implement and are relatively insensitive to training data size. More accurate than most non-linear classifiers. No overfitting issue since it takes average of all predictions. Versatile – can be used for both regression and classification. Stable and robust to outliers and missing values.

Disadvantages: Since it has to get predictions from multiple decision trees to arrive at a solution its very time consuming. A little bit complex compared to Decision trees since there are a lot of trees.

GBT:

Advantages: Gradient boosting trees and generally more accurate than Random Forests. Every successor tree in GBT takes into the error the previous tree made. It curbs overfitting easily.

Disadvantages: Gradient boosting trees cannot be trained in parallel like Random Forest as they have to be trained sequentially one tree at a time to correct each other errors. Slower to train, sensitive to outliers.

Logistic Regression:

Advantages: Simple and easy to implement and interpret the results since it gives probabilistic outputs. Good for simple datasets that have linearity. Can be used as a benchmark to compare other models' efficiency.

Disadvantages: If number of features are more than observations, leads to overfitting. Assumes linearity and multicollinearity which is quite rare in real worlds problems.

Deep Learning:

Advantages: Once trained with large amounts of data gives flawless quality results. Since the model learns itself, no need of feature engineering. Massive Parallel computation. High dimensionality – by adding more layers to the neural network.

Disadvantages: Requires a large amount of data to train and learn. Since its computationally complex, it requires hardware with good GPU. Hard to understand how the complex architecture actually works.

○

(https:/

Aislynn Pasierb (https://canvas.okstate.edu/courses/118118/users/173707)

Apr 17, 2022

The majority of my experience with machine learning comes from the classroom along with minimal experience with using ML techniques in the "real world". I am most experienced with implementing ensemble methods, particularly Random Forests. One advantage of Random Forest I have found in both in and out of the classroom include the ability to use the method on both classification and regression problems. Also, Random Forests do a good job at both linear and nonlinear problems and can handle both numerical and categorical data. One downfall of this method is that since Random Forests are built of Decision Trees which are prone to overfitting, hyperparameter tuning can be difficult as changing on hyperparameter can cause a Random Forest model to then overfit. In addition, these models can be hard to interpret.

○

(http

Ben Lewis (https://canvas.okstate.edu/courses/118118/users/211833)

Apr 20, 2022

Hi Aislynn,

I'm in the same boat on 'real world' experience. I keep trying to find ways to incorporate into my work, but have struggled so far with finding a place where I can use my knowledge. Either the data is too messy, the need isn't one I've encountered in my schooling, or the timeline is too short. I just got access to my companies data lake. While not part of my job description, I think im going to try to find a fun project and work on it in between projects. I enjoyed your post and description of ensemble methods!

-Ben

○ (http [Moises Marin Martinez (https://canvas.okstate.edu/courses/118118/users/198182)](https://canvas.okstate.edu/courses/118118/users/198182)

Apr 28, 2022

Hi Aislynn,

one more advantage of random forest, specially when working with a large number of features, is feature selection. By the way they work with subsets of features, Random forests rank the features by how well they improve the nodes in the tree.

○ (http [Wes Maxwell (https://canvas.okstate.edu/courses/118118/users/165015)](https://canvas.okstate.edu/courses/118118/users/165015)

Apr 30, 2022

Hi Aislynn,

I'm in the same boat as you and Ben, most of my experience with machine learning comes from this class. I'd say I'm most experienced with ensemble/tree methods as well. In my undergrad studies we often used tree-based sorting algorithms to sort large data sets before running them through our application. Thanks for your insight into Random Forest as well, it was very informative!

Best wishes,

-Wes

○ (https:/ [Nithya Satheneni (https://canvas.okstate.edu/courses/118118/users/187996)](https://canvas.okstate.edu/courses/118118/users/187996)

Apr 22, 2022

Hi All,

I have observed few pros and cons in machine learning techniques and listed them below.

Machine Learning Techniques:

Decision Tree:

Advantages:

Supervised learning algorithm used for both classification and regression problems.

No need to bother about feature scaling as it follows rule based approach

Disadvantages:

Due to supervised learning, it takes more time during training the model

After tree construction is done, a small change in data can result in a drastic change in the decision tree

It has high probability of overfitting which leads to wrong predictions of the data

Support Vector Machines:

Adv:

This model is best for classification problems

This is more suitable for smaller datasets

It best in segregating the classes

Disadvantages:

This model does not perform well in large data sets and also data having more noise

This also not perform well when classes are overlapping

Random Forest:

Adv:

Random forest is collection of decision trees

Random forest can be used while needing better accuracy irrespective of the model consideration

It reduces variance error

Disadvantages:

This model creates lot of trees unlike one tree in decision tree

Due to more number of trees it takes more time to train the model compared to decision tree

Neural Networks:

Advantages:

Neural Networks are the best models to train all types of datasets

It automatically assigns weights and baises during training the model

It ignores the noise data or any errors present in the data

It majorly works better in image classification

Disadvantages:

Model should be well defined by user or else user ability can impact the model

This model need high hardware dependency

It does not have specific structure to build the model

Naïve bayes:

Advantages:

This model is fast and easy to implement

This model can be used when planning to train the dataset faster

This model depends on conditional probability which is easy to evaluate

Disadvantages:

It does not provide good performance when attribute values are not trained during training

It may lead to Zero observation probability

Edited by **Nithya Satheneni (https://canvas.okstate.edu/courses/118118/users/187996)** on Apr 22 at 6:56am

○ (https:/

Paul Davis (https://canvas.okstate.edu/courses/118118/users/194957) ⋮

Apr 23, 2022

I've really enjoyed getting the opportunity to play with and test machine learning techniques. I think what has been hard is how some models are much more transparent than others in how they work. For instance, Random Forest is incredibly transparent in how it is implemented. It is very simple for me to understand conceptually and I can see each individual tree that it builds and can see the results of it. I really like being able to see the variable importance for all variables as well. It not only helps with understanding what the model is saying, but also helps with troubleshooting problems in the data and modeling. Also the iterative approach really seems to be effective in increasing the models accuracy. In all of the modeling I've done Random Forest has been either the most accurate, or very close to the most accurate model.

Decision tree models are also very simple and usually very effective in my experience as well. They are very transparent, very fast to execute and I see how easy they would be to deploy into a real world context.

The SVM model seems like it would be highly effective, but it take a lot of time to run and execute the model making it very ineffective to implement and utilize. There are a lot of instance where I think this model would achieve accurate results, but because of the way it is built out in KNIME it just isn’t realistic to utilize.

The neural network models I think are really good at picking up patterns and showing broader patterns in the data. However they are less transparent than the decision tree or random forest models and ultimately that makes them harder to deploy and iterate upon.

I will also say that in this class I really have grown comfortable with the classification models and feel pretty confident about my abilities to implement those, but the mathematical models I feel less confident about both my understanding and my ability to implement them.

○

(https:/

Fayaz Shaik (https://canvas.okstate.edu/courses/118118/users/190988)

Apr 25, 2022

Hello class,

Today, let me share about another day of ML in my life. I have always been into gaming a lot. I am going to share my experience with PC gaming with X-box controllers. I have been playing the game "FIFA 2021" for quite a while now. Playing it often made me realise that whenever I have selected my favourite team to play and my opponent is the AI, I was fascinated by how well the machine learning and AI algorithms works. I have seen the following develop features develop and the reason behind it is ML:

- Game Settings: Whenever I select my go-to team, I used to change my team formation style, controller setting style with respect to different buttons I used, and camera-focus setting style before I start the match. I did this about 3-5 times in the start (training dataset) and later on, whenever I selected the same team, automatically to this date it has been adapted to my settings (trained dataset).
- Player Settings: This was another shocking moment for me, whenever I used my to play with my favourite team, as I said I placed the players in my formation of attacking and defending. Even the players in specific positions like right-wing, left-wing, lower back, center attacking mid, lower back, and even the goalkeeper got used to my playing style and perform better than the usual with respect to stamina, the accuracy of shoots, dribbling, and interception.

These are some of the things I have observed while playing, in my next post I will tell you in detail about what ML algorithms and AI techniques FIFA 2022 uses to make its users blow their minds.

Best Regards,

Shaik Fayaz

MSIS Fall 2021

Edited by **Fayaz Shaik (https://canvas.okstate.edu/courses/118118/users/190988)** on May 1 at 2:39pm

○ (http **Fayaz Shaik (https://canvas.okstate.edu/courses/118118/users/190988)**

May 1, 2022

Hello Class,

Hope you enjoy reading!

I've spent a dozen years playing sports video games, and I've found their near-annual claims of new animations, frequently a record-breaking, more-than-ever amount, to be a marketing cliché. Regular players may not be able to witness all of the flashy acrobatics that is recorded. NBA Live designers said almost a decade ago that they went through the ruins of the abandoned NBA Elite 11 and unearthed a slew of incredible blocks and dunks that the game's engine would never allow. The moral of the story is that recording it is wonderful, but delivering it is where the rubber meets the road.
In the instance of FIFA 22, the motion capture crew may have captured the genuine agony of a goalkeeper who had been defeated by the deciding score in a 4-2 match between two real-life rivals from Spain's Primera División RFEF (the country's third tier). But neither it, nor CD Gerena's desperation play leading to Atlético Sanluqueo's devastating counterattack should be the reason why soccer fans at home think they're viewing a more realistic version of the game, or that it's more fluid and responsive when they play it.
The player can have it both ways when it comes to player movement and engagement, thanks to HyperMotion, which is named by the fact that marketers will promote these animations as well. Animators stated, "In the past, we used to prioritize brief animations so the game is responsive." "The game looks good if you're in a long animation, right?" But if the circumstance changes [in the middle of the animation], and a defender approaches, you're stranded, and you're likely to be tackled, which isn't fun.
The algorithm (which was done by an advanced-works section at EA Vancouver) and the motion capture did not use HyperMotion. Rivera and his team were developing pipelines and processes to receive, interpret, and apply anything their machine spit forth. These little algorithm updates are definitely making a difference in the gaming experience.

Best Regards,

Shaik Fayaz

MSIS Fall 2021

Edited by **Fayaz Shaik (https://canvas.okstate.edu/courses/118118/users/190988)** on May 1 at 4:46pm

○

(https:/

Grant Lackey (https://canvas.okstate.edu/courses/118118/users/87846)

Apr 27, 2022

There are plenty of advantages and disadvantages to Machine Learning techniques. No matter which technique you use, you will find yourself facing similar mistakes and helpful insights.

Advantages: Assuming Machine Learning works correctly;

1) Easily identifiable patterns.

I love finding new patterns and reasoning for specific results without searching for them. It is a great conversational point to address and always fun to display during presentations as your, "ah-ha!" moment!

2) Always room for improvement to find more accurate answers.

I have never created a model to find 100% accuracy and I never will... but that does not stop me from trying to get as close to 100% as possible. I continually adjust, change, and sometimes completely restart projects to simply have a higher accuracy rate.

3) Wide Application usage, multiple analytical tools to use.

I did not know how many analytical tools there were until I joined the graduate program. Yes, most analytical tools do the same thing with their own creative nodes here and there, but it is important to know which one is best for your specific problem. Although it can be stressful at times, it is interesting how I can have a hard time with a certain dataset in one program and then completely understand the dataset from another program. It is amazing to have diversity within analytical tools.

Disadvantages: Common mistakes when using Machine Learning;

1) High-error susceptibility

It is common to run into or create errors throughout your process whether it is from the user or the tool. I have never run through an entire model-building process without error once, and honestly, I don't know if I ever will. That is what users will have to expect when using these tools, but the reward is worth it.

2) Time and resources

Although there is always room for improvement, this takes time to work on these tools and critical thinking. Especially in graduate school, there are simply not enough hours in the day to continually improve a model to address your problem.

3) Results can have varying meanings

After all your hard work, nothing hurts more than an inconclusive result. I have seen this plenty of times and it has not improved my feelings of sadness every time I have to adjust my whole modeling process to find a more accurate conclusion.

Link to kickstart my ideas:

https://data-flair.training/blogs/advantages-and-disadvantages-of-machine-learning/ (https://data-flair.training/blogs/advantages-and-disadvantages-of-machine-learning/)

○ (https:/

Wes Maxwell (https://canvas.okstate.edu/courses/118118/users/165015)

Apr 27, 2022

Hello everyone,

My experience with machine learning/data science has mostly come from what we've learned in this class. Here's some of what I gathered:

Each of the different machine learning techniques have their own advantages and disadvantages, and are designed for different types of problems. For example, the Decision Tree and Random Forest models do not require the input data to be normalized. This can make the pre-processing phase quite easy, allowing to quickly move on to the modeling phase. Comparatively, neural network-based algorithms (such as the Artificial Neural Network model) require normalizing the input data which can lead to a prolonged data understanding or preparation phase. The Decision Tree and Random Forest models are also extremely efficient at handling missing values; which do not have a large effect on the results these models generate. However, small changes in the data fed into these algorithms can have drastic effects on the results. Also, these algorithms can become computationally expensive with large data sets. On the other hand, artificial neural networks are able to work with an incomplete data sets and are robust to errors. One possible drawback to this model is it can only operate on numerical problems, which would require translating non-numerical problems.

A real world example where machine learning has been quintessential is in the fields of biology and bioinformatics. By using machine learning, scientists were able to break world records for predicting how proteins would fold. This will have major impacts on countless studies and initiatives, such as developing targeted medicine.

Thanks everyone!

Best,

Wes Maxwell

Edited by **Wes Maxwell (https://canvas.okstate.edu/courses/118118/users/165015)** on Apr 27 at 8:53pm

○

(https:/

[Rohan Yadav (https://canvas.okstate.edu/courses/118118/users/213842)](https://canvas.okstate.edu/courses/118118/users/213842)

Apr 27, 2022

I really enjoyed this class learning different machine learning techniques. I had already taken another class(machine learning) similar to this when I did master's in computer science. I studied about Artificial Neural Network, k Nearest Neighbor, Logistic Regression, Decision Tree and Random Forest in that class too. I especially liked working on KNIME, SAS and Tableau for data visualization.

Some of the advantages of machine learning I think are:

1. It is difficult to deal with huge amount of data by human since we cannot memorize everything but machines can. Thus they can see the patterns and trends in huge amount of data.

2. Machine learning techniques once programmed they can keep running forever or as long as desired without our intervention and they are finding huge applications in different industry.

Some of the disadvantages of machine learning are:

1. Sometimes machine learning will give us the results but we will be unable to interpret it if not knowledgeable about its working.

2. Data preparation takes huge amount of time and resources and sometimes can be cumbersome.

3. Acquiring data is also getting slightly challenging for ML due to different privacy acts involved and ML are very susceptible to error.

○

(https:/

[Moises Marin Martinez (https://canvas.okstate.edu/courses/118118/users/198182)](https://canvas.okstate.edu/courses/118118/users/198182)

Apr 28, 2022

Advantages and disadvantages of artificial neural networks.

I first came into contact with artificial neural networks in college a good number of years ago, in that class we programmed a perceptron to detect letters from a noisy input. I remembered I put a plus sign where a minus sign was expected and my program was not detecting letters correctly, I then worked with the professor to understand where was the error and he found it. Incorrect sign in the implementation, after that the perceptron was able to predict letters.

I think an advantage of neural networks is the association that we can do between the network and the neurons, understanding how a neuron works is simple, at least conceptually, there are input signals that are added, if they result above a threshold then it sends a signal to its outputs.

A disadvantage is the compute intensive calculations that are executed by large clusters, my mac took 20+ minutes to run 20 epochs in a network, 20 minutes felt like an eternity, because i had to go do something else while the network was executing, then adjust and repeat. More powerful computers can process the equivalent of hundreds of years of calculations in few minutes, that's an impressive power.

Another advantage is the ability to create models that can then be reused without repeating the training expense, for example training with software simulation a robot arm, and then using that model with the real physical device, at a lower cost than training the model with the physical device. Or deploying a model on a mobile phone for quick face or object detection, without training the model on the mobile phone itself.

One more disadvantage is the complexity of some networks, to the point where some patterns can be identified and then repeated multiple times, like blocks used to build large networks, sometimes even recursively. This complexity turns the network into a black box, and when transparency is required to understand the reasons behind decisions of the model, using such a model is not acceptable.

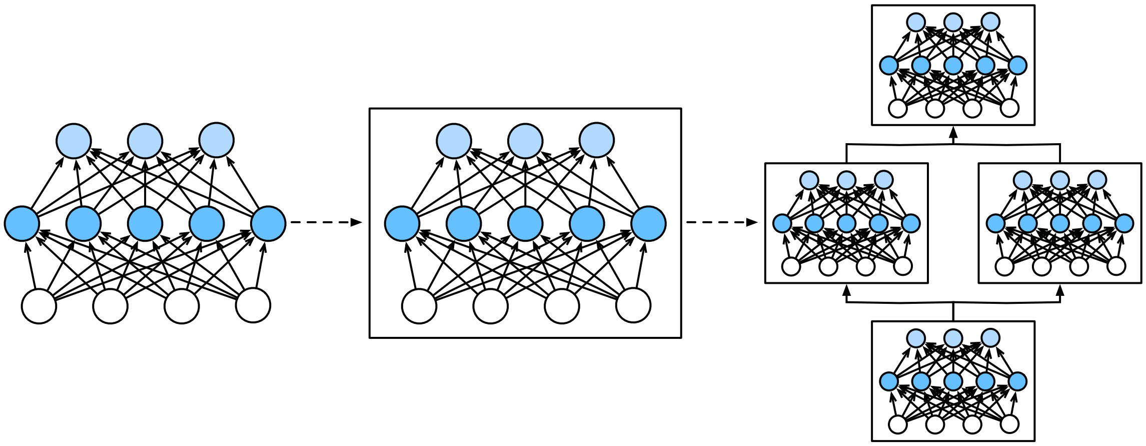


Image taken from [https://d2l.ai/chapter_deep-learning-computation/model-construction.html (https://d2l.ai/chapter_deep-learning-computation/model-construction.html)](https://d2l.ai/chapter_deep-learning-computation/model-construction.html)

Advantages and disadvantages of decision trees.

Decision trees are a way to classify and understand the rules applied in the classification, this transparency is one of the advantages of using this technique. We can even visualize the decision tree if it's not too big.

A disadvantage is that we can get to an overfitting situation where additional settings need to be tweaked, like pruning or setting maximum depth to escape that scenario.

Another disadvantage is that biased trees can be formed if some of the features are more predominant than others.

An advantage of decision trees is that they can be used as a group, in the random forest technique to increase the precision of the result or to reduce overfitting.

Edited by **Moises Marin Martinez (https://canvas.okstate.edu/courses/118118/users/198182)** on Apr 28 at 10:34pm

○

(http

Jacob Wood (https://canvas.okstate.edu/courses/118118/users/214790)

Apr 30, 2022

Moises - I completely agree with your frustration about the large computational power needed for some ANNs. Also, regarding the DT bias, I also wonder about this. I've haven't had time yet, but would like to try different years of accident data on the DT (and all our predictive models) from the project to see how predictive they are on different sets of the same data.

Best,

Jake

○

(https:/

Rithik Ponugoti (https://canvas.okstate.edu/courses/118118/users/189064)

Apr 29, 2022

Hello all,

This course was nothing but amazing. I was given an opportunity to dive deep into the applications of ML techniques. I will be grateful for Dr.Delen and his class!

Coming to the advantages and disadvantages of machine learning techniques:

1. **Regression** is a supervised learning activity that is used to model and predict continuous numeric variables. Predicting real-estate values, stock price changes, or student test scores are some examples.

Strengths: Linear regression is simple to comprehend and explain, and it can be regularized to prevent overfitting. Furthermore, using stochastic gradient descent, linear models may be quickly updated with fresh data.

Weakness: When there are nonlinear connections, linear regression works poorly. They aren't designed to capture more complicated patterns, and adding the necessary interaction terms or polynomials can be difficult and time-consuming.

2. **Decision Trees** learn in a stepwise form by regularly splitting your dataset into independent branches with the best information gained for each split. Regression trees may spontaneously learn non-linear correlations because of their branching structure.

Strengths: Decision trees are capable of learning non-linear correlations and are somewhat resistant to outliers. In reality, ensembles perform well, winning a slew of traditional (i.e. non-deep-learning) machine learning contests.

Weakness: Individual trees that are not limited are prone to overfitting because they can branch until they memorize the training data. However, by using ensembles, this may be avoided.

3. **Deep Learning:** Multi-layer neural networks that can learn incredibly complicated patterns are referred to as deep learning. They describe intermediary data representations that other algorithms cannot readily learn by using "hidden layers" between inputs and outputs.

Strengths: For some applications, like computer vision and speech recognition, deep learning is the current state-of-the-art. Deep neural networks perform exceptionally well on the picture, audio, and text data, and batch propagation makes it simple to update them with fresh data. Their designs (number and structure of layers) may be tailored to a wide range of situations, and their hidden layers eliminate the need for feature engineering.

Weaknesses: Because deep learning methods require a considerable quantity of data, they are typically not suited as general-purpose algorithms. In fact, for traditional machine learning issues, tree ensembles frequently outperform them. In addition, they take a lot of time to train and demand a lot more experience to tune (i.e. set the architecture and hyperparameters).

4. **Linear Regression:** The classification analog of linear regression is logistic regression. The logistic function maps predictions to a range of values between 0 and 1, allowing them to be understood as class probabilities.

Strengths: The method may be regularized to minimize overfitting, and the outputs have a suitable probabilistic meaning. Using stochastic gradient descent, it is simple to update logistic models with fresh data.

Weaknesses: When there are several or non-linear decision boundaries, logistic regression tends to underperform. They aren't adaptable enough to capture more complicated interactions organically.

5. **SVM:** Support vector machines (SVM) use a mechanism called kernels, which essentially calculate the distance between two observations. The SVM algorithm then finds a decision boundary that maximizes the distance between the closest members of separate classes.

Strengths: SVMs have the ability to model non-linear decision boundaries, and there are several kernels from which to pick. They're also resistant to overfitting, particularly in high-dimensional space.

Weakness: SVMs, on the other hand, are memory heavy, more difficult to adjust due to the necessity of choosing the proper kernel, and do not scale well to bigger datasets. Random forests are now favored over SVMs in the business.

Best,

Rithik Sai Ponugoti

○ (http [Navya Mynedi (https://canvas.okstate.edu/courses/118118/users/157687)](https://canvas.okstate.edu/courses/118118/users/157687)

May 1, 2022

Hi Rithik,

Thank you for sharing the insights about each model. You explained effectively about strengths and weaknesses of each model. Yes, I agree with you decision trees are not constrained and are prone to overfitting because they can branch until the training data is memorized. And coming to the SVM they don't function well for large data sets one more disadvantage of SVM is It doesn't perform well when we have overlapped classes. if you find time, look into this link to get some good insights about SVM and decision tree algorithms.

https://data-flair.training/blogs/svm-kernel-functions/ (https://data-flair.training/blogs/svm-kernel-functions/)

○ (http [Thirumala Krishna Kurakula (https://canvas.okstate.edu/courses/118118/users/161132)](https://canvas.okstate.edu/courses/118118/users/161132)

May 1, 2022

Hi Rithik,

Thanks for sharing this information. I liked the way you explained the advantages and disadvantages of ML techniques. You mentioned that deep learning techniques demand a large amount of data. I feel that large-scale DL algorithms require more development. But I would also say that the DL model performs best at learning complicated statistical properties using large amounts of data.

○ (https:/ [Jeya Subburaj (https://canvas.okstate.edu/courses/118118/users/167277)](https://canvas.okstate.edu/courses/118118/users/167277)

Apr 29, 2022

ML Models Pros & Cons

- Multiple Linear Regression
- Logistic Regression
- k-Nearest Neighbors (KNN)
- k-Means Clustering
- Decision Trees/Random Forest
- Support Vector Machine (SVM)
- Naive Bayes

Multiple Linear Regression

Pros

- Easy to implement, the theory is not complex, **low computational power** compared to other algorithms.
- Easy to **interpret coefficients** for analysis.
- Perfect for **linearly separable datasets**.
- Susceptible to **overfitting**, but can avoid using dimensionality reduction techniques, cross-validation, and regularization methods.

Cons

- Unlikely in the real world to have perfectly linearly separable datasets, the model often **suffers from under-fitting** in real-world scenarios or is outperformed by other ML and Deep Learning algorithms.
- **Parametric**, has a lot of assumptions that need to be met for its data in regard to its distribution. **Assumes a linear relationship** between the dependent and independent variables.
- **Examples of assumptions: There is a linear relationship between the dependent variable and the independent variables. The independent variables aren't too highly correlated with each other. Your observations for the dependent variable are selected independently and at random. Regression residuals are normally distributed.**

2. **2. Logistic Regression**

Pros

- **A simple algorithm** that is easy to implement, does **not require high computation** power.
- Performs extremely well when the data/response variable is **linearly separable**.

- Less prone to over-fitting, with **low-dimensional data**.
- Very easy to **interpret**, can give a measure of how **relevant a predictor** is and the **association** (positive or negative impact on response variable).

Cons

- Logistic regression has a **linear decision surface** that separates its classes in its predictions, in the real world it is extremely rare that you will have linearly separable data.
- Need to perform careful data exploration, logistic regression suffers from datasets with **high multicollinearity between their variables**, repetition of information can lead to **wrong training of parameters**.
- Requires that **independent variables are linearly related to the log odds (log(p/(1-p))**.
- The algorithm is **sensitive to outliers**.
- Hard to **capture complex relationships**, deep learning, and classifiers such as Random Forest can outperform with more realistic datasets.

3. **k-Nearest Neighbors (KNN)**

Pros

- A **lazy-learning algorithm**, **no actual training step**, new data is simply tagged to a majority class, based on historical data. Very easy to understand and implement.
- Can be used for **both** classification and regression.
- Only **one hyper-parameter**: k value.
- **Non-parametric** makes no assumptions about the data/parameters

Cons

- Struggles with a large **number of dimensions**, the greater the number of dimensions the harder for the algorithm to efficiently calculate the distance (Curse of Dimensionality). Often **need to use dimensionality reduction** techniques, especially in regression tasks with noisy data.
- Very **sensitive to outliers & noise**.
- Can become very **computationally expensive** as the **dataset grows**, need a lot of memory, and can become very slow with a large-sized dataset.
- **K value selection**, often need to estimate ranges or combine with cross-validation techniques to obtain the optimal k value selection. A frequent technique is to **plot an elbow graph** to find the optimal k value

4. **k-Means Clustering**

Pros

- Very easy to **interpret** the results and highlight conclusions in a visual manner.
- Very flexible and fast, also **scalable** for large datasets.
- Always yields a result.

Cons

- Struggles with a **high number of dimensions**, need to use PCA or spectral clustering to help fix the issue.
- Choosing **K value** manually/based on your domain knowledge of the problem. Need to use the **elbow method** to assess the best K value.
- Sensitive to **outliers**.
- Sensitive to **initialization**, if the initial centroids you pick are inaccurate this can cause problems with later points.

5. **Decision Trees/Random Forest**

Decision Tree Pros

- Do **not** need to **scale and normalize data**.
- Handles **missing values** very well.
- Less effort in regard to **preprocessing**.

Decision Tree Cons

- Very prone to **overfitting**.
- Sensitive to **outliers** and changes in the data.
- Takes a **long time to train** and is expensive complexity wise
- **Weak** in terms of **regression**.

Random Forest Pros

- Performs well on **imbalanced data**.
- Works well with **high dimensionality** and handles a **large amount of data**.
- **De-correlates trees** can deal with the problem of variance.
- Solves **classification and regression** issues.

Random Forest Cons

- Need to have **predictive power** with the **features** otherwise, it is inefficient.
- Can be a **black box**, hard to interpret.
- Code Demo Link Random Forest

6. **Support Vector Machine (SVM)**

Pros

- Very effective with **high-dimensional data**.
- Works extremely well when there is a **clear margin of separation**.
- Effective when there are **more dimensions than a number of samples**.
- **Outliers have less of an impact** as the hyperplane is affected only by the support vectors.

Cons

- **Selecting an appropriate kernel** can be computationally expensive/need to know the dataset very well to be able to pick the right kernel.
- Can take a large amount of time with a large dataset.
- Struggles with performance when there is a lot of **overlap between the target classes** or noise in classification problems.

7. **Naive Bayes**

Pros

- **Speed**, and assumptions of feature independence allow the algorithm to be very fast. If this assumption holds true, performs exceptionally well.
- Performs well with **multi-class prediction**.
- Works well with **high dimensions**, and works well with problems such as **text classification** (spam detection in the code demo).

Cons

- Assumes **all features are independent**, this is rarely accurate in real life.
- **Zero Frequency**: If the categorical variable has a category in the test data set, which was not observed in the training data set, the model assigns a zero probability to this category and fails at making a prediction. Use **smoothing** to deal with this issue.
- Smoothing is a technique for **detecting trends with noisy data** for cases where the shape of the trend is unknown. **Laplace Smoothing** is common with Naive Bayes, it is used with categorical data and meant to alleviate the problem of zero probability.

(https:/

Neeraj Kankani (https://canvas.okstate.edu/courses/118118/users/190006)

Apr 29, 2022

Artificial Neural Networks

Pros

1. Neural networks are flexible and can be used for both regression and classification problems. Any data which can be made numeric can be used in the model, as neural network is a mathematical model with approximation functions.
2. Neural networks are good to model with nonlinear data with large number of inputs; for example, images. It is reliable in an approach of tasks involving many features. It works by splitting the problem of classification into a layered network of simpler elements.
3. Once trained, the predictions are pretty fast.
4. Neural networks can be trained with any number of inputs and layers.
5. Neural networks work best with more data points.

Cons

1. Neural networks are black boxes, meaning we cannot know how much each independent variable is influencing the dependent variables.
2. It is computationally very expensive and time-consuming to train with traditional CPUs. Artificial Neural Networks require processors with parallel processing power, by their structure.
3. Neural networks depend a lot on training data. This leads to the problem of over-fitting and generalization. The mode relies more on the training data and may be tuned to the data.
4. Further, there is no assurance of proper network structure for a neural network. The appropriate network structure is achieved through experience and trial and error.

SVM (Support Vector Machine)

Pros

1. **Performs well in Higher dimension.** In real world there are infinite dimensions (and not just 2D and 3D). For instance image data, gene data, medical data etc. has higher dimensions and SVM is useful in that. Basically when the number of features/columns are higher, SVM does well

2. **Best algorithm when classes are separable.** (when instances of both the classes can be easily separated either by a straight line or non-linearly). To depict separable classes, lets take an example(here taking an example of linear separation, classes can also be non-linearly separable, by drawing a parabola for e.g. etc). In first graph you cannot tell easily whether X will be in class 1 or 2, but in case 2 you can easily tell that X is in class 2. Hence in second case classes are linearly separable.

3. **Outliers** have less impact.

4. SVM is suited for extreme case binary classification.

Cons:

1. **Slow:** For larger dataset, it requires a large amount of time to process.

2. **Poor performance with Overlapped classes** : Does not perform well in case of overlapped classes.

3. **Selecting appropriate hyperparameters is important:** That will allow for sufficient generalization performance.

4. **Selecting the appropriate kernel** function can be tricky.

Applications:

Bag of words application(many features and columns), speech recognition data, classification of images(non-linear data), medical analytics(non linear data), text classification(many features)

Naive Bayes

Pros

1. **Real time** predictions: It is very fast and can be used in real time.

2. **Scalable** with Large datasets

3. **Insensitive to irrelevant features.**

4. **Multi class prediction** is effectively done in Naive Bayes

5. **Good performance with high dimensional data**(no. of features is large)

Cons

1. **Independence of features does not hold:** The fundamental Naive Bayes assumption is that each feature makes an independent and equal contribution to the outcome. However this condition is not met most of the times.

2. **Bad estimator:** Probability outputs from predict_proba are not to be taken too seriously.

3. **Training data should represent population well:** If you have no occurrences of a class label and a certain attribute value together (e.g. class="No", shape="Overcast ") then the posterior probability will be zero. So if the training data is not representative of the population, Naive bayes does not work well.(This problem is removed by smoothening techniques).

Applications:

Naive Bayes is used in Text classification/ Spam Filtering/ Sentiment Analysis. It is used in text classification (it can predict on multiple classes and doesn't mind dealing with irrelevant features), Spam filtering (identify spam e-mail) and Sentiment Analysis (in social media analysis, to identify positive and negative sentiments), recommendation systems (what will the user buy next)

Logistic Regression

Pros

1. **Simple** to implement

2. **Effective**

3. **Feature scaling not needed:** Does not require input features to be scaled (can work with scaled features too, but doesn't require scaling)

3. **Tuning of hyperparameters not needed.**

Cons

1. **Poor performance on non-linear data**(image data for e.g)

2. **Poor performance with irrelevant and highly correlated features** (use Boruta plot for removing similar or correlated features and irrelevant features).

3. **Not very powerful** algorithm and can be easily outperformed by other algorithms.

4. **High reliance on proper presentation of data**. All the important variables / features should be identified for it to work well.

Applications:

Any classification problem that is preferably binary (it can also perform multi class classification, but binary is preferred). For example you can use it if your output class has 2 outcomes; cancer detection problems, whether a student will pass/fail, default/no default in case of customer taking loan, whether a customer will churn or not, email is spam or not etc.

Random Forest

Pros:

1. Random forest can **decorrelate trees**. It picks the training sample and gives each tree a subset of the features(suppose training data was [1,2,3,4,5,6], so one tree will get subset of training data [1,2,3,2,6,6]. Note that size of training data remains same, both datas have length 6 and that feature ‘2’ and feature ‘6’ are repeated in the randomly sampled training data given to one tree. Each tree predicts according to the features it has. In this case tree 1 only has access to features 1,2,3 and 6 so it can predict based on these features. Some other tree will have access to features 1,4,5 say so it will predict according to those features. If features are highly correlated then that problem can be tackled in random forest.

2. **Reduced error:** Random forest is an ensemble of decision trees. For predicting the outcome of a particular row, random forest takes inputs from all the trees and then predicts the outcome. This ensures that the individual errors of trees are minimized and overall variance and error is reduced.

3. **Good Performance on Imbalanced datasets** : It can also handle errors in imbalanced data (one class is majority and other class is minority)

4. **Handling of huge amount of data:** It can handle huge amount of data with higher dimensionality of variables.

5. **Good handling of missing data:** It can handle missing data very well. So if there is large amount of missing data in your model, it will give good results.

6. **Little impact of outliers:** As the final outcome is taken by consulting many decision trees so certain data points which are outliers will not have a very big impact on Random Forest.

7. **No problem of overfitting:** In Random forest considers only a subset of features, and the final outcome depends on all the trees. So there is more generalization and less overfitting.

8. **Useful to extract feature importance** (we can use it for feature selection)

Cons:

1. **Features** need to have **some predictive power** else they won't work.

2. **Predictions of the trees need to be uncorrelated**.

3. **Appears as Black Box:** It is tough to know what is happening. You can at best try different parameters and random seeds to change the outcomes and performance.

Applications:

Credit card default, fraud customer/not, easy to identify patient's disease or not, recommendation system for ecommerce sites.

Decision Trees

Pros

1. **Normalization or scaling of data not needed**.

2. **Handling missing values**: No considerable impact of missing values.

3. **Easy to explain** to non-technical team members.

4. **Easy visualization**

5. **Automatic Feature selection** : Irrelevant features won't affect decision trees.

Cons

1. **Prone to overfitting.**

2. **Sensitive to data.** If data changes slightly, the outcomes can change to a very large extent.

3. **Higher time required to train** decision trees.

Applications:

Identifying buyers for products, prediction of likelihood of default, which strategy can maximize profit, finding strategy for cost minimization, which features are most important to attract and retain customers (is it the frequency of shopping, is it the frequent discounts, is it the product mix etc), fault diagnosis in machines(keep measuring pressure, vibrations and other measures and predict before a fault occurs) etc.

XGBoost

Pros

1. **Less feature engineering required** (No need for scaling, normalizing data, can also handle missing values well)

2. **Feature importance** can be found out(it output importance of each feature, can be used for feature selection)

3. **Fast** to interpret

4. **Outliers** have minimal impact.

5. **Handles large sized datasets** well.

6. **Good Execution** speed

7. **Good model performance** (wins most of the Kaggle competitions)

8. **Less prone to overfitting**

Cons

1. **Difficult interpretation** , visualization tough

2. **Overfitting** possible if parameters not tuned properly.

3. **Harder to tune** as there are too many hyperparameters.

Applications

Any classification problem. Specially useful if you have too many features and too large datasets, outliers are present, there are many missing values and you don’t want to do much feature engineering. It wins almost all competitions so this is an algo you must definitely have in mind while solving any classification problem.

k-NN (K Nearest Neighbors)

Pros

1. **Simple** to understand and impelment

2. **No assumption about data** (for e.g. in case of linear regression we assume dependent variable and independent variables are linearly related, in Naïve Bayes we assume features are independent of each other etc., but k-NN makes no assumptions about data)

3. **Constantly evolving** model: When it is exposed to new data, it changes to accommodate the new data points.

4. **Multi-class** problems can also be solved.

5. **One Hyper Parameter:** K-NN might take some time while selecting the first hyper parameter but after that rest of the parameters are aligned to it.

Cons

1. **Slow** for large datasets.

2. **Curse of dimensionality**: Does not work very well on datasets with large number of features.

3. **Scaling** of data absolute must.

4. **Does not work well on Imbalanced data.** So before using k-NN either undersamplemajority class or oversample minority class and have a balanced dataset.

5. Sensitive to **outliers**.

6. Can't deal well with **missing values**

Applications:

You can use it for any classification problem when dataset is smaller, and has lesser number of features so that computation time taken by k-NN is less. If you do not know the shape of the data and the way output and inputs are related (whether classes can be separated by a line or ellipse or parabola etc.), then you can use k-NN.

Edited by **Neeraj Kankani (https://canvas.okstate.edu/courses/118118/users/190006)** on Apr 29 at 7:46pm

(https:/

Pranjali Pingale (https://canvas.okstate.edu/courses/118118/users/190864)

Apr 29, 2022

Here are my thoughts on the most popular machine learning algorithms, their pros and cons and where it makes sense to use them.

Naive Bayes

Pros

- super simple(just doing some counts) yet performing well in practice.
- compute the multiplication of independent distributions
- require less training data
- no distribution requirements
- converge quicker than discriminative models(e.g. logistic regression) under conditional independence assumption
- good for few categories variables

Cons

- suffer multicollinearity

Logistic Regression

Logistic regression is probably the most widely used classification algorithm which is based out of statistical modelling.

- easy to interpret. **the output can be interpreted as a probability: you can use it for ranking instead of classification.**
- good for cases where features are expected to be roughly linear, and the problem to be linearly separable.
- can easily "feature engineering" most non-linear features into linear ones.
- robust to noise
- can use l2 or l1 regularization to avoid overfitting(and for feature selection)
- efficient, and can be distributed(ADMM)
- no distribution requirement
- compute the logistic distribution
- cannot handle categorical(binary) variables well
- compute Confidence Interval
- suffer multicollinearity
- no need to worry about features being correlated, like in Naive Bayes.
- easily update the model to take in new data (using an online gradient descent method)
- use it if you want a probabilistic framework (e.g., to easily adjust classification thresholds, to say when you're unsure, or to get confidence intervals)
- use it if you expect to receive more training data in the future and want to quickly be incorporate into the model.

Lasso(L1)

- no distribution requirement
- compute L1 loss
- variable selection
- suffer multicollinearity

Ridge(L2)

- no distribution requirement
- compute L2 loss
- no variable selection
- not suffer multicollinearity

When NOT to use

- if the variables are normally distributed and the categorical variables all have 5+ categories: use Linear discriminant analysis
- if the correlations are mostly nonlinear: use SVM
- if sparsity and multicollinearity are a concern: Adaptive Lasso with Ridge(weights) + Lasso

Linear Discriminant Analysis

LDA: Linear discriminant analysis, not latent Dirichlet allocation

- require normal distribution
- not good for few categories variables
- compute the addition of Multivariate distribution
- compute Confidence Interval
- suffer multicollinearity

Support Vector Machines (SVM)

- Support Vector Machines (SVMs) use a different loss function (Hinge) from LR.
- they are also interpreted differently (maximum-margin).
- SVM with a linear kernel is similar to a Logistic Regression in practice
- if the problem is not linearly separable, use an SVM with a non linear kernel (e.g. RBF). (Logistic Regression can also be used with a different kernel)
- good in a high-dimensional space (e.g. text classification).
- high accuracy
- good theoretical guarantees regarding overfitting
- no distribution requirement
- compute hinge loss
- flexible selection of kernels for nonlinear correlation
- not suffer multicollinearity
- hard to interpret

Cons:

- can be inefficient to train, memory-intensive and annoying to run and tune
- not for problems with many training examples.
- not for most "industry scale" applications (anything beyond a toy or lab problem)

Decision Tree

- Easy to interpret and explain
- Non-parametric, no need to worry about outliers or whether the data is linearly separable.
- no distribution requirement
- heuristic
- good for few categories variables
- not suffer multicollinearity (by choosing one of them)
- can easily overfit,
- tree ensembles
    - e.g. Random Forests and Gradient Boosted Trees, using bagging or boosting
    - generally outperform single decision tree.
    - handle very well high dimensional spaces as well as large number of training examples.

Random Forest

- train each tree independently, using a random sample of the data, so the trained model is more robust than a single decision tree, and less likely to overfit
- 2 parameters: number of trees and number of features to be selected at each node.
- good for parallel or distributed computing.
- lower classification error and better f-scores than decision trees.
- perform as well as or better than SVMs, but far easier for humans to understand.
- good with uneven data sets with missing variables.
- calculates feature importance
- train faster than SVMs

Gradient Boosted Trees

- build trees one at a time, each new tree corrects some errors made by the previous trees, the model becomes even more expressive.
- 3 parameters - number of trees, depth of trees, and learning rate; trees are generally shallow.
- usually perform better than Random Forests, but harder to get right. The hyper-parameters are harder to tune and more prone to overfitting. RFs can almost work "out of the box".
- training takes longer since trees are built sequentially

Artificial Neural Networks

- good to model the non-linear data with large number of input features
- widely used in industry
- many open source implementations
- only for numerical inputs, vectors with constant number of values, and datasets with non-missing data.
- "black box-y", the classification boundaries are hard to understand intuitively("like trying interrogate the human unconscious for the reasons behind our conscious actions.")
- computationally expensive.
- the trained model depends crucially on initial parameters
- difficult to troubleshoot when they don't work as expect
- not sure if they will generalize well to data not in training set
- multi-layer neural networks are usually hard to train, and require tuning lots of parameters
- not probabilistic, unlike their more statistical or Bayesian counterparts. The continuous number output (e.g. a score) can be difficult to translate that into a probability.

Deep Learning

- not a general-purpose technique for classification.
- good in image classification, video, audio, text.

This is based on my experience so far into machine learning. If you have any comments or feedback, you're welcome to write it down below!

Edited by **Pranjali Pingale (https://canvas.okstate.edu/courses/118118/users/190864)** on Apr 29 at 8pm

○ (https:/

[Josh Basquez (https://canvas.okstate.edu/courses/118118/users/158078)](https://canvas.okstate.edu/courses/118118/users/158078)

Apr 29, 2022

These machine learning techniques are new to me this semester howeveri have had some experience in discussions of the k nearest neighbor knn technique in relation to the classification of new data points. I think knn is a simplification of the learning methods of small children and makes sense that it would be useful in machine systems as well.

○ (https:/

[Jacob Wood (https://canvas.okstate.edu/courses/118118/users/214790)](https://canvas.okstate.edu/courses/118118/users/214790)

Apr 30, 2022

Hello all,

From lectures, working in KNIME, and extra reading on ML, I've included some Pros and Cons of various methods of Machine Learning below.

Support Vector Machines –

- Pro – SVMs have really great predictive performance. In our first homework assignment using KNIME, SVMs were the best at predicting churn. Also, in the paper from Dr. Delen titled "Investigating injury severity risk factors in automobile crashes with predictive analytics and sensitivity analysis methods" the SVM performed best. Clearly, predictive performance is a strength of SVM models.
- Pro – These models are flexible and deployable in many instances as they can be used for both classification and prediction models.
- Con – This was a very computationally expensive algorithm. With larger data sets processing speed can become a limiting factor. This became especially challenging if an SVM is paired with 10-fold cross validation.

Decision Trees –

- Pro – Can be utilized to understand key variables, based on the variables used for the first few splits. I was able to use this method in the Customer Churn Homework to get a quick understanding of key variable. This was a great advantage to Decision Trees.
- Pro – Can be easily represented visually so that it can be communicated easily to end-users. This can be very important depending on the business use, and stakeholders.
- Pro – Can be used for both regression and classification type problems.
- Con – Generally these models have high variance. As decision trees split until each row of data has been fit, small changes in the underlying data can impact the model. Due to this,

Decision Trees may have problems making predictions on new datasets.

Artificial Neural Networks –

- Pro – In my experience ANNs have been some of the best predictive algorithms to employ. In the county voting homework, ANNs gave a great result.
- Pro – Using the MLP Learner in KNIME is great as you can increase the complexity of the model by adding or subtracting additional layers to the algorithm. This has the ability to increase accuracy.
- Con – A large disadvantage to ANNs is that the algorithm is difficult to understand. Using these algorithms to make real world decisions may be difficult if total trust by stakeholder is needed and those stakeholders cannot easily have their questions answered.

Logistic Regression –

- Pro – Logistic regressions have been used in various fields and have been around for a long time. They are simple to develop, and easy to communicate and explain to stakeholders.
- Con – In my experience, Logistic Regressions have not created strong predictive models. In each of the homeworks and in the project, they have been outperformed by ANNs, DTs, Tree Ensembles, RFs, etc. For complex data sets, I would not expect LRs to be the most predictive.
- Con – Logistic regressions require linear relationships, and as such, they are not well-suited to interpret complex relationships.

I've really enjoyed the hands-on nature of our KNIME homeworks and the team project. I generally learn from doing, so the insights developed from this coursework was really reenforced through these assignments.

(http

Chitra Boorla Boorla (https://canvas.okstate.edu/courses/118118/users/193742)

May 1, 2022

Hello Jacob,

Thank you for sharing your valuable insights. I really like how you used real life situations related to us to explain the pro's and con's for the model's. It was pretty easy to understand each model especially after reading many theoretical explanations of these models. Initially I found it hard to work on knime but once I started to read more about the different models and how it can used it real life to predict and solve problems i started loving it.

(https:/

Navya Mynedi (https://canvas.okstate.edu/courses/118118/users/157687)

May 1, 2022

Based on the experience I got during working on the project I have learned many things about different machine learning models.

SVM(support vector machine):

Advantages:

1. we can use SVM in higher dimension data for example image data that has more number of dimensions. Basically, it works well when we have more features in the data.

2. Outliers have less impact on SVM

3. It is more suitable for binary classification.

Disadvantages:

1. It takes more time while work on a large dataset.

2. It doesn't perform well when we have overlapped classes

3. Selecting the appropriate functions is more important for better performance.

Naïve Bayes:

Advantages:

1. It is more useful when we have multi-class predictions.

2. when we have large data sets compared to other models naïve Bayes is more useful

3. One of the important features is it is insensitive to irrelevant features.

Disadvantages:

1. The fundamental assumption is that each attribute contributes equally and independently to the result in most of the conditions this assumption doesn't meet.

2. The 'zero-frequency issue' occurs when an algorithm assigns zero probability to a categorical variable whose category in the test data set was not present in the training dataset.

KNN:

Advantages:

1. No assumptions were made in this model like linear regression and naïve Bayes

2. We can solve multi-class problems easily in this model

3. It is a constantly evolving model when we apply new data it will create new data points according to the new data.

Disadvantages:

1. On huge datasets, it is slower.

2. On datasets with a huge number of features, it won't operate well.

Logistic Regression

Advantages:

1.Logistic regression is more straightforward to apply, analyze, and train.

2. It doesn't make any assumptions about class distributions in feature space.

3. It classifies unknown records very quickly.

Disadvantages:

1. The assumption of linearity between the dependent and independent variables is a major limitation of Logistic Regression.

2. Only discrete functions may be predicted using it. As a result, the discrete number set is bound to the dependent variable of Logistic Regression.

Decision Tree:

Advantages:

1. It doesn't require normalized data

2. missing values in the data have no significant impact on the decision tree-building process.

3. And also it doesn't require any scaling of data

Disadvantages:

1. When compared to other algorithms, a decision tree's calculation might get rather complicated at times.

2. slight change in the data can result in a substantial change in the decision tree's structure, resulting in instability.

Random Forest

Advantages:

1. It works well with both categorical and continuous values.

2. It can automatically handle the missing values.

3. Unlike curve-based algorithms, nonlinear parameters have no effect on the performance of a Random Forest. As a result, if the independent variables are very non-linear, Random Forest may outperform conventional curve-based methods.

Disadvantages:

1. It takes much time to train the data

2. Sometimes it takes more time because it has to create a lot of decision trees.

ANN:

Advantages:

1. A neural network is capable of completing tasks that a linear program is incapable of.
2. When a component of the neural network deteriorates, its parallel properties allow it to continue without causing problems.
3. A neural network determines without the need for reprogramming.

Disadvantages:

1. The construction of Artificial Neural Networks necessitates parallel processing power. As a result, the equipment's manifestation is dependent.

2. When ANN provides a probing answer, it does not explain why or how it was chosen. The network's trust is lost as a result of this.

Edited by **Navya Mynedi (https://canvas.okstate.edu/courses/118118/users/157687)** on May 1 at 2:08pm

○ (https:/

Ashish Kumar Pampana (https://canvas.okstate.edu/courses/118118/users/24369)

May 1, 2022

Hello! I'm Ashish. This is my personal experience, please pardon me if I'm not accurate with my comments.

ML Pros:

1) With all the ML techniques available, I have a lot of options to choose from while doing any analysis.

2) These techniques give a lot of flexibility while working with supervised or unsupervised learning.

3) With platforms like KNIME, SAS, etc., I can choose the best classification model fit for my analysis based on the accuracy and error matrix.

4) I can have a deeper understanding of the variables I'm working with.

ML Cons:

1) As an engineer, I can't always think from the business perspective for a problem statement. Rather I need to go deep into the analysis and do it from the scratch and write my own codes. I feel the automations available for doing all kinds of analysis makes us lazy and run towards the results rather than understanding the process of any technique.

2) Having platforms like KNIME and SAS always an advantage when we need quick results but this makes us limited in creativeness and develop a new kind of analysis where we can tweak one or two things in the methodology of the techniques.

Thank you!

○

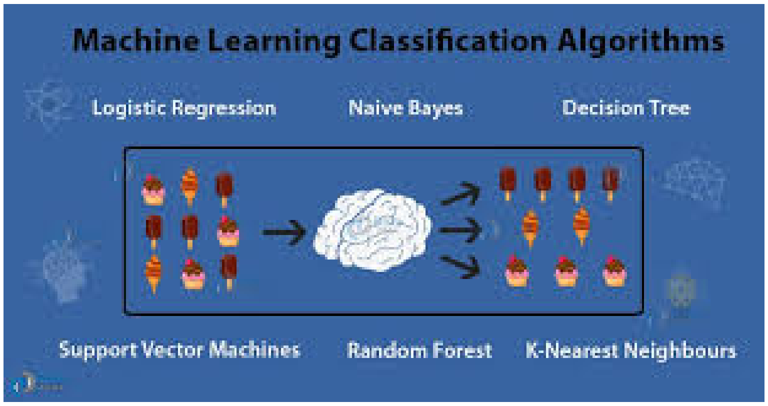
(https:/

[Fayaz Shaik (https://canvas.okstate.edu/courses/118118/users/190988)](https://canvas.okstate.edu/courses/118118/users/190988)

May 1, 2022

Hello Class!

Hope you are holding on to these tough times of Finals Week. It has been a pleasure reading all of your insightful posts and information on distinct ML techniques. Thank you class for your valuable posts. Here is my take on it (other than my above-referred ML in my daily life posts):



Machine Learning algorithms look through a set of different prediction models to find the one that best reflects the relationship between the descriptive and target properties.

Deep Learning has the advantage of being able to deal with tensors and requiring a high-end machine configuration. The remaining variables can be utilized for classification or regression. Working on data features is far more significant than choosing a specific ML model to use.

Though it may appear that creating an ML model is simple, it requires a significant amount of time and effort to:

- Gather data
- Clean data
- Understand distributions
- Wrangle data

Because a training dataset is typically always a tiny sample of the world/actual problem that we wish to model/solve in real life, machine learning is ill-posed. If the data isn't clean, datasets may deceive us and offer an entirely other picture. Deep Learning assists us in solving a variety of difficulties that Machine Learning is unable to address.

Machine learning is a well-posed problem that refers to whether or not the problem is stable, as evaluated by the following criteria:

There is a solution: for all d, s exists (for every d relevant to the problem).

A unique solution: s is unique for all d; there is only one value of s for each data point d.

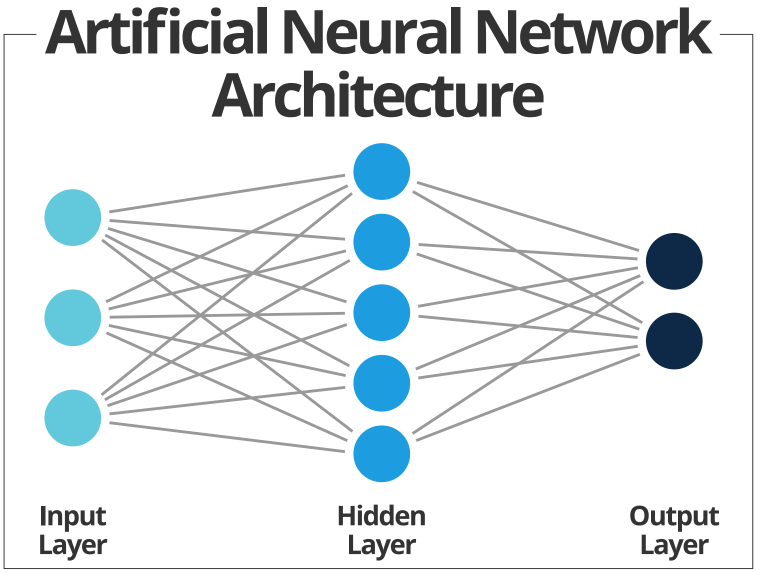
A stable solution: s is always dependent on d.

Machine learning has the potential to go wrong in a number of ways.

1. If an algorithm performs well on a subset of issues, it must compensate for this by degrading performance on the rest of the problems.
2. If the incorrect inductive bias is chosen, underfitting or overfitting may occur.

Now, let us look into the following ML techniques:

- Artificial neural networks
- K nearest neighbors
- Support vector machines
- Decision Trees
- Random Forest
- Gradient boost techniques
- Linear regression
- Deep learning



ARTIFICIAL NEURAL NETWORKS

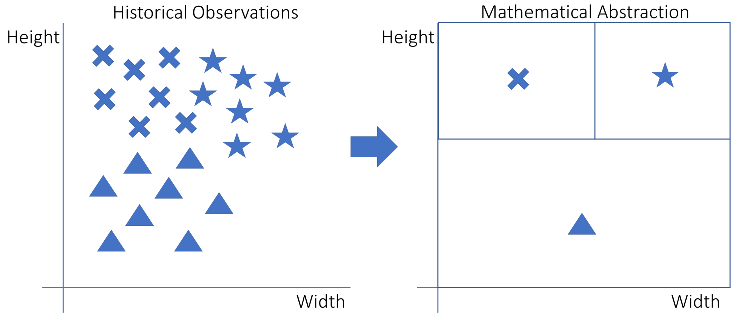
Pros:

- The entire network can be used to store data.

- Neural networks have the ability to cope with ambiguous data.
- They are tolerant of mistakes.
- Parallel processing is possible.
- They may be taught based on previous occurrences, and the ANNs can make decisions.

Cons:

- When dealing with categorical variables, random forests are found to be biased.
- Training at a Slow Pace
- Linear approaches with a lot of sparse features aren't recommended.



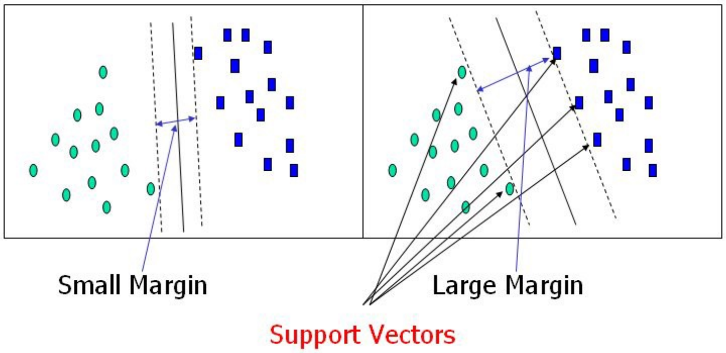
K-NEAREST NEIGHBOURS

Pros:

- For multi-class problems, it is simple to implement.
- Both classification and regression can be done with it.
- It's quite straightforward and straightforward.
- There is no need for training, and it is always evolving as a result of its learning.
- There are several distances to choose from, including Euclidean, Hamming, Manhattan, and Minkowski distances.

Cons:

- As efficiency improves, speed decreases.
- Dimensionality's curse.
- It can only work with features that are all the same.
- Outliers have a big impact on it.
- When classifying a new data entry, determining the optimal number of neighbors is a critical issue.



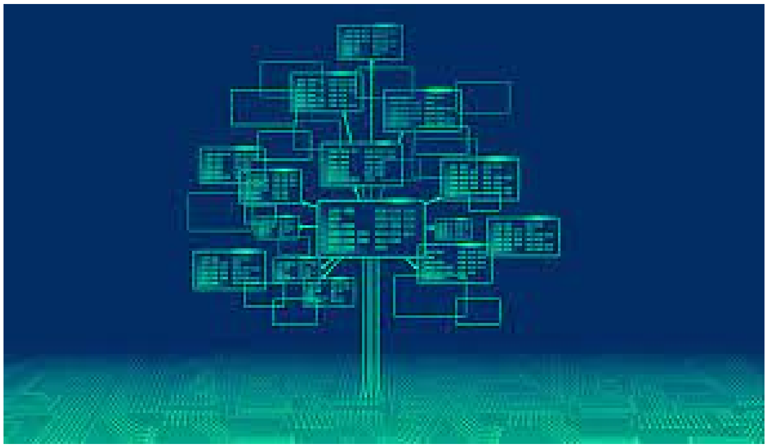
SUPPORT VECTOR MACHINES

Pros:

- In high-dimensional spaces, it's more effective.
- It is memory-friendly.

Cons:

- Large data sets are not recommended.
- When there is noise in the dataset, it does not perform well.



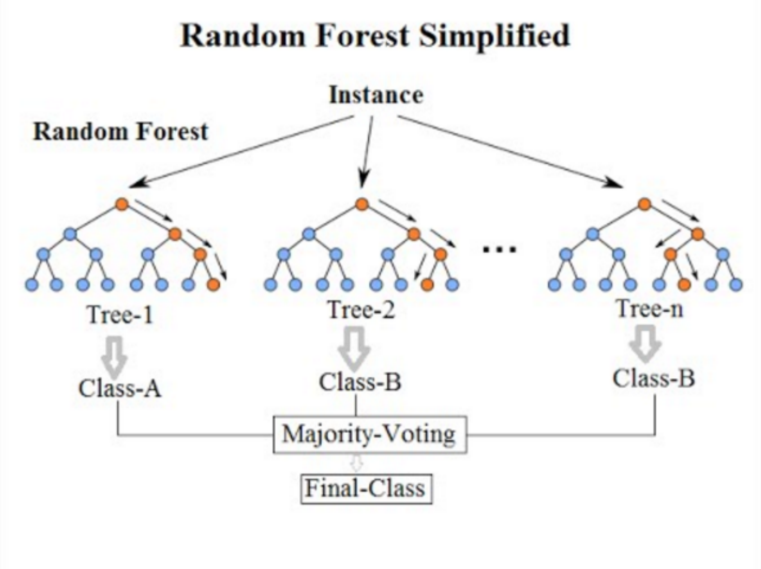
DECISION TREES

Pros:

- During pre-processing, decision trees take less effort for data preparation.
- The data does not need to be normalized.
- A decision tree is very straightforward and simple to comprehend.
- The process of creating a decision tree is unaffected by missing values.

Cons:

- A small modification in the data can completely alter the decision tree's structure.
- The training of a decision tree frequently takes a long period.
- It is a costly procedure due to its complexity and length of time.



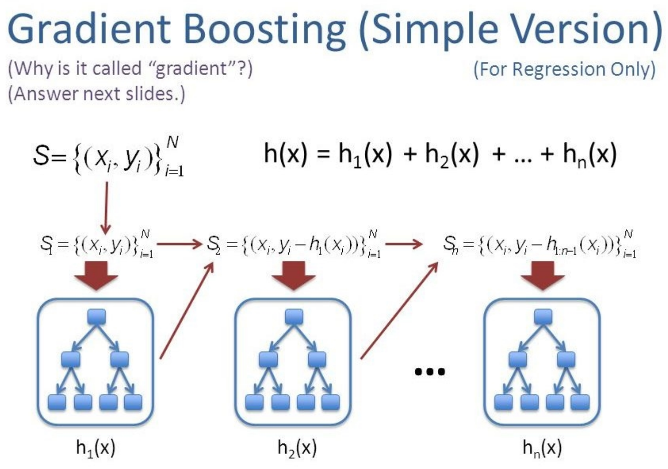
RANDOM FOREST

Pros:

- Outlier-resistant.
- Works well with data that isn't linear.
- Overfitting is less likely.
- On a large dataset, it performs well.
- Other classification methods have a worse accuracy than this one.

Cons:

- When dealing with categorical variables, random forests are found to be biased.
- Training at a Slow Pace.
- Linear approaches with a lot of sparse features are not acceptable.


Gradient Boosting (Simple Version)

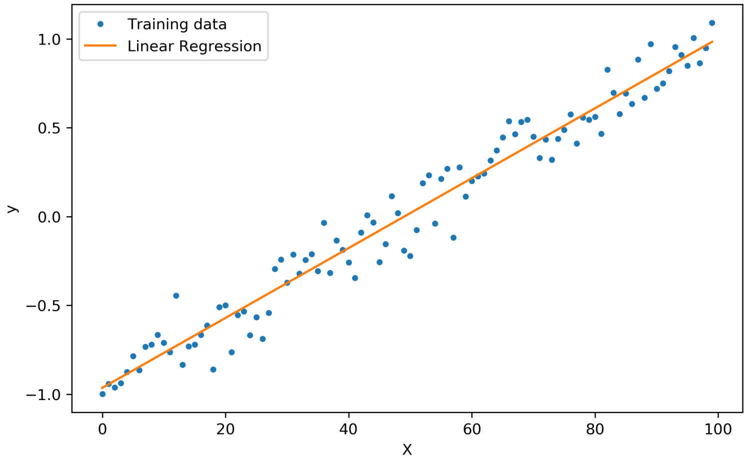
(Why is it called "gradient"?)
(Answer next slides.)

(For Regression Only)

$S=\{(x_i, y_i)\}_{i=1}^{N}$
$h(x) = h_1(x) + h_2(x) + \ldots + h_n(x)$
$S_1=\{(x_i, y_i)\}_{i=1}^{N}$
$S_2=\{(x_i, y_i - h_1(x_i))\}_{i=1}^{N}$
$S_n=\{(x_i, y_i - h_{1:n-1}(x_i))\}_{i=1}^{N}$
...
$h_1(x)$
$h_2(x)$
$h_n(x)$

GRADIENT BOOSTING TECHNIQUES

Boosting, as an ensemble model, has an easy-to-read and interpret algorithm, making prediction interpretations simple. Through the employment of clone methods like as bagging, random forest, and decision trees, the prediction capability is effective. Boosting is a durable approach for reducing over-fitting. Because every classifier is required to rectify the flaws in the predecessors, boosting is sensitive to outliers. As a result, the technique is overly reliant on outliers. Another downside is that scaling up the process is nearly impossible. This is due to the fact that each estimator is based on the accuracy of preceding predictors, making the operation difficult to simplify.



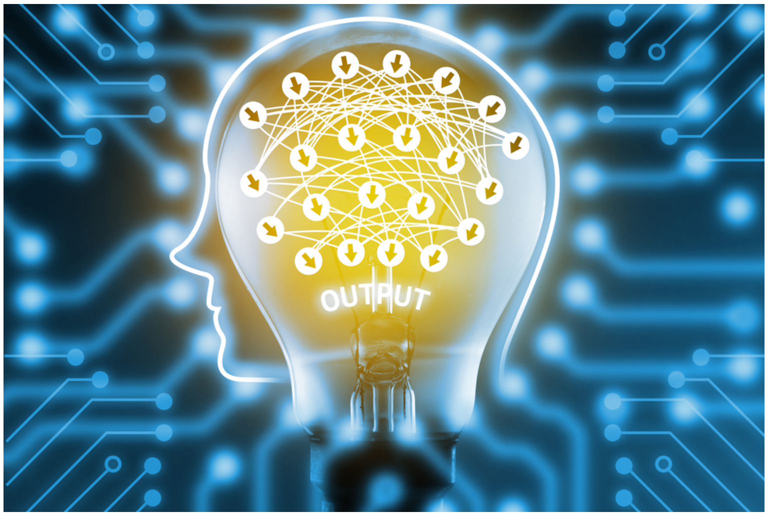
LINEAR REGRESSION

Advantages:

- Linear regression performs exceptionally well for linearly separable data.
- It is easier to implement, interpret, and train.
- It effectively handles overfitting using dimensionally reduction techniques, regularization, and cross-validation; and it has the ability to extrapolate beyond a specific data set.

Disadvantages:

- The dependent and independent variables are assumed to be linear.
- Linear regression is prone to noise and overfitting.
- It is sensitive to outliers.
- It is prone to multicollinearity.



DEEP LEARNING

Advantages:

- Feature engineering can be carried out automatically within the Deep Learning model.
- It can tackle difficult problems and is adaptable to new challenges in the future (or transfer learning can be easily applied).
- High levels of automation. Users can design a deep learning model in seconds using a deep learning library (Tensorflow, Keras, or MATLAB..), (without the need of deep understanding).

Disadvantages:

- Requires a large quantity of data.
- Expensive and rigorous training.
- Overfitting when applied to simple situations.
- There is no standard for training and tuning the model.
- It's a blackbox, difficult to understand inside each layer.

Thank you for reading!

All The Best everyone on your finals!

Best Regards,

Shaik Fayaz

MSIS Fall 2021

Edited by Fayaz Shaik (https://canvas.okstate.edu/courses/118118/users/190988) on May 1 at 4:47pm

○ (http

Fayaz Shaik (https://canvas.okstate.edu/courses/118118/users/190988)

May 1, 2022

Adding on to my above understandings, here is another a blog where all the ML techniques are described in a nutshell:

https://towardsdatascience.com/11-most-common-machine-learning-algorithms-explained-in-a-nutshell-cc6e98df93be

○ (http **Seonwoo Ko (https://canvas.okstate.edu/courses/118118/users/194258)**

May 1, 2022

Hello Shaik,

Thank you for your information! I was looking for the deep learning technique but I could not find some good resources. Yours is really helpful for me to understand not only deep learning but also other techniques with your additional photos. Thank you again.

Thanks,

Seonwoo

○ (http **Thirumala Krishna Kurakula (https://canvas.okstate.edu/courses/118118/users/161132)**

May 1, 2022

Hi Fayaz,

Thanks for sharing this information. I'd like to add a few points to the advantages of decision trees. The decision tree algorithm finds the fastest ways to detect important connections. Though this algorithm needs more time to train the data, this algorithm gives efficient results.

○ (https:/ **Thirumala Krishna Kurakula (https://canvas.okstate.edu/courses/118118/users/161132)**

May 1, 2022

Advantages of the ANN model:

1. Structured programming data is stored on the network, not in a repository.
2. It is known for its rapid analysis report.
3. ANNs adapt from comparable occurrences and conclude.

Disadvantages of the ANN model:

1. A network ages and degrades.
2. The architecture of ANN is not predetermined. Only Brute Force leads to proper network composition.
3. The functionality of the ANN is unknown.

Advantages of the KNN model:

1. When designing a KNN model, the user may specify the range.
2. computes both classification and regression issues.
3. This algorithm is simple to use and requires no training.

Disadvantages of the KNN model:

1. Not suitable for huge datasets.
2. Performance deteriorates due to distortion, erroneous information, and abnormalities.
3. Needs more time and space for execution.

Advantages of the SVM model:

1. Quick and accurate predictions.
2. Minor data changes have little influence on the higher dimensional space.
3. Selecting the proper kernel with the configuration tool may give a lot of freedom.

Disadvantages of the SVM model:

1. The SVM shifts over from parameterization to feature extraction.
2. Takes more time to get trained.
3. Requires a lot of memory storage.

Advantages of the Decision Tree model:

1. The decision tree paradigm is simple to analyze, grasp, and depict.
2. fastest approaches to find meaningful correlations among attributes.

Disadvantages of the decision tree model:

1. Regularization is a problem with DT models.
2. The learning algorithm keeps creating assumptions that lower training data set error but increase test set error. Scaling and constraining model variables may help fix this problem,
3. Unlike KNN, this algorithm needs more time to train the data.

Advantages of the Random Forest model:

1. Effective on unbalanced datasets
2. We can get attribute statistics and with that, we can calculate variable importance.

3. It does not affect performance because of distortion, erroneous information, and abnormalities

Disadvantages of the Random Forest model:

1. it is not suitable for regression since it does not provide exact indefinite aspect projection.
2. The fundamental drawback of RF is that it is sluggish and useless for meaningful forecasts.

Advantages of the Gradient Boosted Trees model:

1. Incorporating the trees individually is a cyclical and progressive strategy. Each repetition should minimize our loss function.
2. More accurate than other models learn quicker over bigger datasets, some inherently handle incomplete data.

Disadvantages of the Gradient Boosted Trees model:

1. Identifies our data badly, possibly as well as arbitrary speculation. It is prone to mistakes.
2. This model is practically infeasible to train

Advantages of the Logistic regression model:

1. Uses less computing power, making it appropriate for machine learning categorization.
2. The probability measurements might be quite beneficial if we combine this application with some other framework based on probabilistic estimates.

Disadvantages of the logistic regression model:

1. Converting the non-linear situations to streamlining may be difficult and complicated,
2. struggles to capture complicated correlations.

Advantages of the Deep Learning:

1. Instantaneously learns resilience through natural fluctuations in data.
2. Attributes are dynamically determined and optimized. No prior feature engineering is necessary.
3. This framework is adaptable to potential changes.

Disadvantages of the Deep Learning:

1. It takes a lot of information to outperform conventional strategies.
2. Choosing the correct deep learning resources needs an understanding of architecture, coaching techniques, and many other characteristics.

(https:/

[Srikanth Daruru (https://canvas.okstate.edu/courses/118118/users/193729)](https://canvas.okstate.edu/courses/118118/users/193729)

May 1, 2022

Artificial Neural Networks:

Artificial neural networks are the modeling of the human brain with the simplest definition and building blocks are neurons. There are about 100 billion neurons in the human brain. In multi-layer artificial neural networks, there are also neurons placed in a similar manner to the human brain. Each neuron is connected to other neurons with certain coefficients. During training, information is distributed to these connection points so that the network is learned.

Advantages:

1. Information such as in traditional programming is stored on the entire network, not on a database. The disappearance of a few pieces of information in one place does not prevent the network from functioning.
2. Artificial neural networks learn events and make decisions by commenting on similar events
3. After ANN training, the data may produce output even with incomplete information. The loss of performance here depends on the importance of the missing information.

Disadvantages

1. After ANN training, the data may produce output even with incomplete information. The loss of performance here depends on the importance of the missing information.
2. When ANN produces a probing solution, it does not give a clue as to why and how. This reduces trust in the network.
3. There is no specific rule for determining the structure of artificial neural networks. Appropriate network structure is achieved through experience and trial and error.

KNN:

K- Nearest Neighbors or also known as K-NN belong to the family of supervised machine learning algorithms which means we use labeled (Target Variable) dataset to predict the class of new data point.

Advantages

1. K-NN algorithm is very simple to understand and equally easy to implement. To classify the new data point K-NN algorithm reads through whole dataset to find out K nearest neighbors
2. One of the biggest advantages of K-NN is that K-NN can be used both for classification and regression problems
3. K-NN might take some time while selecting the first hyper parameter but after that rest of the parameters are aligned to it.

Disadvantages

1. K-NN might be very easy to implement but as dataset grows efficiency or speed of algorithm declines very fast.
2. KNN works well with small number of input variables but as the numbers of variables grow K-NN algorithm struggles to predict the output of new data point.
3. K-NN inherently has no capability of dealing with missing value problem.

SVM:

SVM (Support Vector Machine) is used to classify the data, with a hyperplane serving as a judgment border between different classes. Extreme data points from each class are used to create Support Vectors. The goal of SVM is to find the best and most ideal hyperplane for each Support Vector with the highest margin.

Advantages

1. SVM works relatively well when there is a clear margin of separation between classes.
2. SVM is more effective in high dimensional spaces.
3. SVM is effective in cases where the number of dimensions is greater than the number of samples.

Disadvantages

1. SVM algorithm is not suitable for large data sets.
2. SVM does not perform very well when the data set has more noise i.e. target classes are overlapping.
3. In cases where the number of features for each data point exceeds the number of training data samples, the SVM will underperform.

Logistic Regression:

Logistic regression is also known as **Binomial logistics regression**. It is based on sigmoid function where output is probability and input can be from -infinity to +infinity. Let’s discuss some advantages and disadvantages of Linear Regression

Advantages:

1. Logistic regression is easier to implement, interpret, and very efficient to train.
2. It not only provides a measure of how appropriate a predictor(coefficient size)is, but also its direction of association (positive or negative).
3. Good accuracy for many simple data sets and it performs well when the dataset is linearly separable.

Disadvantages:

1. If the number of observations is lesser then the number of features, Logistic Regression should not be used, otherwise, it may lead to overfitting.
2. It can only be used to predict discrete functions. Hence, the dependent variable of Logistic Regression is bound to the discrete number set
3. Logistic Regression requires average or no multicollinearity between independent variables.

Decision Tree:

Decision Trees are a type of Supervised Machine Learning (that is you explain what the input is and what the corresponding output is in the training data) where the data is continuously split according to a certain parameter. The tree can be explained by two entities, namely decision nodes and leaves.

Advantages

1. Compared to other algorithms decision trees requires less effort for data preparation during pre-processing.
2. A decision tree does not require normalization of data.
3. A decision tree does not require scaling of data as well.
4. Missing values in the data also do NOT affect the process of building a decision tree to any considerable extent.

Disadvantages

1. A small change in the data can cause a large change in the structure of the decision tree causing instability.

2. For a Decision tree sometimes calculation can go far more complex compared to other algorithms.

3. Decision tree often involves higher time to train the model.

4. Decision tree training is relatively expensive as the complexity and time has taken are more.

Deep Learning:

Multi-layer neural networks that can learn incredibly complicated patterns are referred to as deep learning. They describe intermediary data representations that other algorithms cannot readily learn by using "hidden layers" between inputs and outputs.

Advantages

1. Features are automatically deduced and optimally tuned for desired outcome.
2. The same neural network based approach can be applied to many different applications and data types.
3. The deep learning architecture is flexible to be adapted to new problems in the future.

Disadvantages

1. It requires very large amount of data in order to perform better than other techniques
2. It is extremely expensive to train due to complex data models.
3. There is no standard theory to guide you in selecting right deep learning tools as it requires knowledge of topology, training method and other parameters.

↰ (https://canvas.okstate.edu/courses/118118/discussion_topics/782449/entry-3684810#)

Edited by **Srikanth Daruru** (https://canvas.okstate.edu/courses/118118/users/193729) on May 1 at 10:24pm

○ (http **Seonwoo Ko** (https://canvas.okstate.edu/courses/118118/users/194258) ⋮

May 1, 2022

Hello Srikanth,

Thank you for sharing your perspective and insights! Yes, I agree with you that big data analytics can be helpful for solving problems and developing businesses in diverse industries. For example, in the hospitality and tourism industry, such as hotels, resorts, and restaurants, the power of big data has been getting an increase to make better decisions in advance. I also think that big data analytics will be more popular in the near future, so we have to know various machine learning algorithm techniques as many as we can!

Thanks,

Seonwoo

○

(https:/

[Seonwoo Ko (https://canvas.okstate.edu/courses/118118/users/194258)](https://canvas.okstate.edu/courses/118118/users/194258)

May 1, 2022

I haven't used any Machine Learning techniques before this class, it has been challenging for me to differentiate and think about the advantages and disadvantages. Thanks to this course, I was able to think about which model is optimized or beneficial for certain datasets. I will share my own thoughts and knowledge about Machine Learning Techniques! I hope this helps.

1. ANN (Artificial Neural Networks)

1) Pros

- It is easier to handle more than one variable at the same time.
- ANN can be trained and learned from the past one, so it is easier to make decisions.

2) Cons

- It could take a long time to figure out what is the most appropriate structure for ANN. We can know the most proper one with trial and error.
- There is no way why ANN gives that solution, such as why and how.

2. Logistic Regression

1) Pros

- It is easy to train and interpret the data.
- It can handle a large dataset efficiently and rapidly.

2) Cons

- It is hard to process missing value data as compared to other machine learning techniques.
- It is possible to overfit the dataset, so a larger dataset is needed.

3. SVM (Support Vector Machine)

1) Pros

- It is capable of regularization, so it can prevent overfitting issues.
- SVM can handle both classification and regression models.

2) Cons

- SVM is not efficient and cannot handle larger datasets.
- It is hard for SVM to process some overlaps and noises within the dataset.

4. GBT (Gradient Boosting Trees)

1) Pros

- It is easy to interpret and predict the algorithm.
- It can be a comparably accurate model compared to others.

2) Cons

- It can make an issue if outliers are included in the dataset, causing too much dependent on outliers.
- It can take a long time to train the algorithms.

5. K-NN (K-Nearest Neighbors)

1) Pros

- It is easy to implement and understand the algorithms.
- K-NN can handle both classification and regression models.

2) Cons

- The features are needed to be on the same scale. Otherwise, it is hard for K-NN to interpret and implement algorithms.
- It can cause an issue if there are outliers and missing values in the dataset.

○ (http

Chitra Boorla Boorla (https://canvas.okstate.edu/courses/118118/users/193742)

May 1, 2022

Hello seonwoo,

Its my first time too studying about predictive models and different ML techniques. Thank you for sharing your insights. It is pretty crisp and hence very easy to understand. My personal favorites are random forest and SVM. I'm intrigued by all these different models and looking forward to working with these models in the future.

○ (https:/

Chitra Boorla Boorla (https://canvas.okstate.edu/courses/118118/users/193742)

May 1, 2022

Machine learning is becoming more pervasive in recent years as a result of increased demand and technological advancements. Machine learning's ability to extract value from data has made it appealing to companies across a wide range of industries. The majority of machine learning products are created and implemented using off-the-shelf machine learning algorithms with minor tweaks.

There are three major categories of machine learning algorithms:

Given a set of observations, **supervised learning algorithms** model the relationship between features (independent variables) and a label (target). Using the features, the model is then used to predict the label of new observations. It can be a classification (discrete target variable) or a regression (continuous target variable) task, depending on the characteristics of the target variable.

Unsupervised learning algorithms look for structure in data that hasn't been labeled.

The action-reward principle underpins **reinforcement learning**. An agent learns to achieve a goal by calculating the reward of its actions iteratively.

Linear regression is a supervised learning algorithm that tries to fit a linear equation to the data to model the relationship between a continuous target variable and one or more independent variables.

A linear relationship between the independent variable(s) and the target variable is required for a linear regression to be a good choice. Scatter plots and correlation matrices are two tools that can be used to investigate the relationship between variables.

Advantages of linear Regression

- Linear regression performs exceptionally well for linearly separable data
- It is easier to implement, interpret and efficient to train
- It handles overfitting pretty well using dimensionally reduction techniques, regularization, and cross-validation
- One more advantage is the extrapolation beyond a specific data set

Disadvantages:

- It is often quite prone to noise and overfitting
- Linear regression is quite sensitive to outliers
- It is prone to multicollinearity

The **Support Vector Machine** (SVM) is a supervised learning algorithm that is commonly used for classification but can also be used for regression.

By drawing a decision boundary, SVM distinguishes classes. The most important aspect of SVM algorithms is how to draw or determine the decision boundary. Each observation (or data point) is plotted in n-dimensional space before the decision boundary is created. The number of features used is indicated by the letter "n." If we use "length" and "width" to classify different "cells," for example, observations are plotted in a 2-dimensional space with a line as the decision boundary. The decision boundary is a plane in three-dimensional space if we use three features.

When more than three features are used, the decision boundary becomes a hyperplane, which is difficult to visualize.

Advantages of SVM

- It performs well in higher dimension, i.e., when the number of features or columns are more svm does really well
- It is a best algorithm when the classes are separable i.e., either by a straight line or non-linearity
- SVM is well suited to binary classification in extreme cases.

Disadvantages:

- It is quite slow when there is a huge data set and hence requires more time
- It doesn't perform well in case of overlapped classes
- The SVM will underperform if the number of features for each data point exceeds the number of training data samples.
- There is no probabilistic explanation for the classification because the support vector classifier works by placing data points above and below the classifying hyperplane.

The **supervised learning algorithm Naive Bayes** is used for classification tasks. As a result, it's also known as the Naive Bayes Classifier.

The assumption in naive bayes is that features are independent of one another and that there is no correlation between them. The term "naive" refers to the algorithm's naive assumption that features are uncorrelated.

Advantages:

- This algorithm works quickly and can save a lot of time
- Naive Bayes is suitable for solving multi-class prediction problems.
- If its assumption of the independence of features holds true, it can perform better than other models and requires much less training data.
- Naive Bayes is better suited for categorical input variables than numerical variables.

Disadvantages

- In Naive Bayes, all predictors (or features) are assumed to be independent, which is rarely the case in real life. This limits the algorithm's applicability in real-world scenarios.
- This algorithm faces the 'zero-frequency problem' where it assigns zero probability to a categorical variable whose category in the test data set wasn't available in the training dataset. It would be best if you used a smoothing technique to overcome this issue.
- Its estimations can be wrong in some cases, so you shouldn't take its probability outputs very seriously.

The **supervised learning algorithm logistic regression** is most commonly used for binary classification problems. Although "regression" and "classification" are incompatible terms, the emphasis here is on the word "logistic," which refers to the logistic function that performs the classification task in this algorithm. Because logistic regression is a simple yet powerful classification algorithm, it is frequently used for binary classification tasks.

Advantages

- It is effective and simple to implement
- It makes no assumptions about distributions of classes in feature space.
- It not only provides a measure of how appropriate a predictor (coefficient size)is, but also its direction of association (positive or negative).

Disadvantages

- If the number of observations is lesser than the number of features, Logistic Regression should not be used, otherwise, it may lead to overfitting.
- The major limitation of Logistic Regression is the assumption of linearity between the dependent variable and the independent variables.

The supervised learning algorithm **K-nearest neighbors (kNN)** can be used to solve both classification and regression problems. The main idea behind kNN is that a data point's value or class is determined by the data points surrounding it.

Advantages:

- Simple to understand and implement and Multi-class problems can also be solved.
- When it is exposed to new data, it changes to accommodate the new data points.
- K-NN might take some time while selecting the first hyper parameter but after that rest of the parameters are aligned to it.

Disadvantages

- Does not work very well on datasets with large number of features. Hence is time taking.
- It is sensitive to outliners and cant deal with missing values

A **random fores**t is a collection of multiple decision trees. Bagging is a method for creating random forests that uses decision trees as parallel estimators. When used to solve a classification problem, the outcome is determined by a majority vote of the results from each decision tree. The mean value of the target values in a leaf node is the prediction of that leaf in regression. The mean value of the decision tree results is used in random forest regression.

Advantages:

- It can handle both large data and missing values very well
- It works well with both categorical and continuous variables
- Random Forest is usually robust to outliers and can handle them automatically.

Disadvantages:

- Random Forest takes a lot longer to train than decision trees because it generates a lot of trees (instead of just one) and makes decisions based on the majority of votes.
- Features need to have some predictive power else they won’t work, and also predictions need to be uncorrelated

Edited by **Chitra Boorla Boorla** **(https://canvas.okstate.edu/courses/118118/users/193742)** on May 1 at 10:52pm