# Literature Review

3RD PERSON  
“A argues that”  
“B argues this”  
“Based on this it is clear that...”

[INTRODUCTION]

[MAIN BODY – TITLES OF RESEARCH ]

Definition and components of a data pipeline (DP)  
 Data Storage  
 SQLite3 Database  
 Data Cleaning  
 Handling missing data, outlier removal  
 Feature Engineering  
 What makes a good feature?  
 State of the art feature engineering techniques  
 Model building and evalutation  
 Best Performing Models in the industry (Strengths and weaknesses in regards to this task)  
 Methods of Evaluating Models (including developing an effective baseline to compare models against  
 Data Sources Available Online (That can be used in the project)

The evolution of a data pipeline

The value of a DP  
Benefits of a DP for predictive analysis

The use of a DP in real world scenarios

Challenges of using a DP

[CONCLUSION]

[REFERENCES]

# Summary

# Introduction

There are a number of studies of predictive analysis techniques available online, however the focus of this report is on english football prediction, so only relevant studies using Premier League data will be reviewed in any detail in the Literature Review section, however other studies may be referred to as and when appropriate.

This review summarises and critically analyses any arguments found during online research. This paper argues that classification algorithms are most appropriate as the variables we are predicting are discrete variables and not continuous.

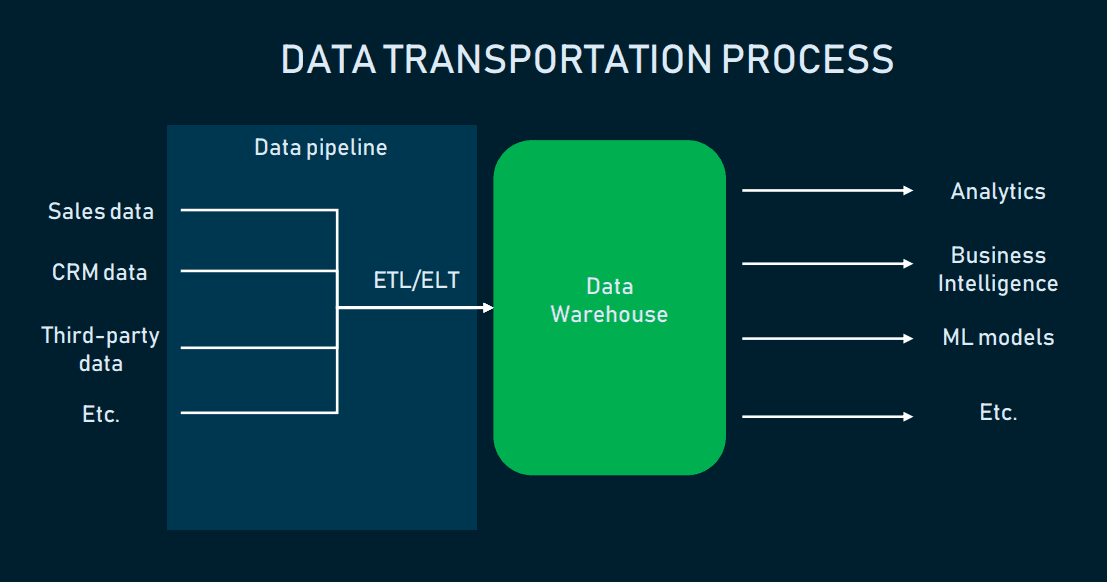
It also identifies what types of engineered features are best suited to various machine learning model types, and analyses research on machine learning toolkits that are currently available on Python.

All sources are compared and any constrasting viewpoints are highlighted in the literature.

# Main Body

## Definition and Components of a Data Pipeline (DP)

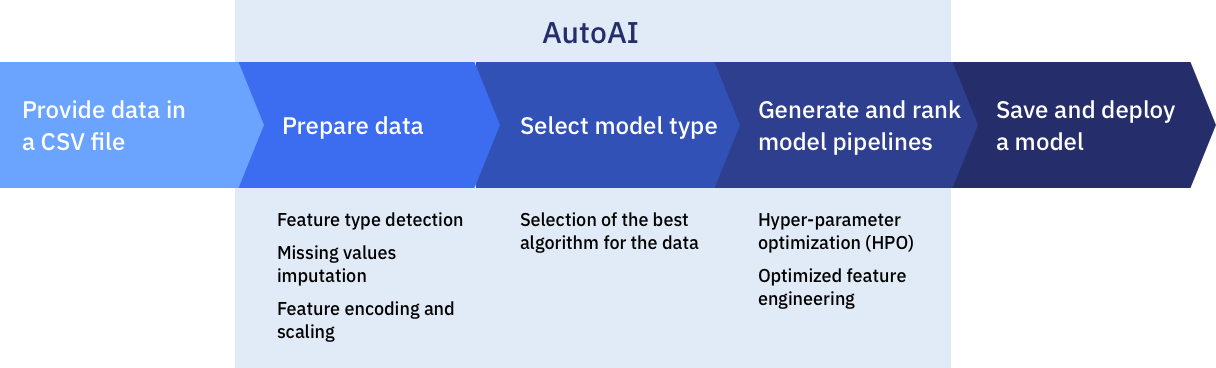
A Data Pipeline (DP) is a combination of tools, techniques and processes used to integrate data and discover new information and patterns. The purpose of a DP is to transfer raw data into an ideal environment for it to be analysed.

  
*[ALTEXSOFT] The process of transporting raw data into a Data Warehouse (DW)*

The data requires seperation from its initial storage for a number of reasons, as stated below:

* To save computational power - By using optimal computation power soley for one task, it therefore increasing performance power
* To store and structure the data in a way that makes sense for analysis
* To increase security and privacy of data, restricting who can access certain data
* To reduce the risk of losing data, as it's backed up in its previous storage location [DREMIO]

In a DP there are multiple tasks that are usually carried out in a specific order, similar to a waterfall methodology. Once one task is done, we go onto the next.

  
*[AUTOAI] An illustration showing the process of a data pipeline.*

### Data Storage

Numeric data is often saved as .CSV files as this allows them to easily be imported to or exported from a program that uses tables to store the data, such as on Microsoft Excel [COMPUTERHOPE]. This data type is great for storing numeric data as the data fields are delimited and seperated with a comma. This makes it easier to interact with the data as it presents it in a structured and easier to read manner.

Once we have the data source stored in a csv file, we can then use a program such as Python to feed the data into the pipeline and store it in a database where it can be queried.

#### SQLite3 Database

The standard python library features a module called "sqlite3" which is intended to be used when working with an SQLite database. SQLite is a lightweight database that uses C library. The advantage of using this database is that it doesn't need a server to access the database - you can simply access the database using a non-standard variant of SQL [PYTHON]. There is also a free-to-use, open source tool called DB Browser that is compatible with SQLite which allows you to make it easier to create, search and edit a database [DBBROWSER].

For the purpose of this project, an SQLite3 database is used to store the data.

### Data Cleaning

Data cleaning refers to the finding and correcting of errors in the data. Dirty data, or data that has not been cleaned, leads to innacurate data and therefore incorrect decision making [DATACLEANING]. It is for this reason that data cleaning is so important.

By addressing errors in the data, we can in turn make the data more accurate, readable and make better decisions. Errors in the data refer to missing values, random erroneous data, spelling mistakes, different formats, replicated entries and any violations of data integrity [DATACLEANING].

The majority of scientific studies and complex models use and rely upon assumptions to ensure the validity of the results and to avoid undesirable outcomes [JOSBOURNE]. According to Jason Osborne, author of Best practices in Data Cleaning, "cleaning data and addressing assumptions can have important benefits on the power, effect size and accuracy of population estimates" [JOSBORNE].

Data cleaning usually consists of two phases;

* Error detection (analysing the database for errors)
* Error repair (updating the database)   
  [DATACLEANING]

We can either use quantitative or qualitative error detection techniques. Quantative techniques use statistical methods to identify errors, whereas qualitative techniques identifies errors by specifying patterns or contraints and highlighting data that violates them [DATACLEANING].

#### Handling missing data, outlier removal

By knowing the cause of the missing data, we can apply the most appropriate method for analysis. [PIGOTT] There are a number of methods that can be used for handling missing data.

We could use **complete-case analysis**, which only uses cases that are not missing variables. This is appropriate if there is only a few observations, as we can assume the missing data is **MCAR** (Missing Completely At Random) in accordance to Rubin's terminology [PIGOTT]. The researcher can use standard methods to compute estimates making it easy to implement, however if there are large amounts of missing data, this method is not appropriate as there may not an be an adequate amount of data remaining. [PIGOTT]

Another method we could use is **available-case analysis**, which utilises all available data to estimate the parameters of the model. We can use different cases of data to estimate parameters of interest within the data set. When variables are moderatetly correlated within regression models this method can be applied to provide consistent estimates. [PIGOTT] The more correlated the variables, the more inadequate the results become.

If the previous two methods are not appropriate, we could use Single-Value Imputation. This method involves the researcher filling the missing value with a more plausible value, such as the mean of all cases that observe the variable. This is referred to as mean imputation. [PIGOTT] The problem with this method is it will likely lead to variables being underestimated, as the true value is likely not the same as the mean value, and the results will therefore be biased. [PIGOTT]

To remove outliers from the data, we must inspect the data and use our judgement to determine outliers. We can determine outliers by identifying data that is believed to be erroneous and contributing to the larger range of data. Once the outliers are identified we can set their value to 0 [GREENWOOD].

### Feature Engineering

Feature Engineering (FE) refers to the process of using domain knowledge to extract features from raw data in order to improve model accuracy. The workflow is an iterative process in which we create new features from old features to improve model accuracy and performance. FE plays a key role in successfully predicting important process information when developing a data-driven machine learning model. [SHAH] These features are then used to improve the machine learning algorithms for better model performance.

The process of Feature Engineering is as follows:

1. **Brainstorm ideas for features** – Research other projects and discuss feature ideas to base the model on.
2. **Create features** – You can automatically extract features, manually construct features or use a combination of both depending on your needs and constraints.
3. **Select features** – List and score the features and determine the best features to base the model on.
4. **Evaluate model performance** – Using the chosen features, estimate the accuracy of the model. This can be done on unseen data to test its performance.  
   [Brownlee]

Feature selection is focused on removing non-informative and un-needed predictors from the model. Non-informative variables will reduce the accuracy of the model and make the predictions unreliable. [Brownlee]

There are two main methods for feature selection; **supervised** and **unsupervised**. To help decide which is the most appropriate method, we must determine whether the features are selected based on the target variable, or not. If they are then it is a supervised method, if not then it is unsupervised. [Brownlee] Examples of supervised selection methods can be split into wrapper and filter methods.

Alternatively, we can use **machine learning algorithms** to select features. These algorithms can perform automatically during the learning process of the model, and are referred to as intrinsic feature selection methods. Examples of these algorithms include penalized regression models like decision-trees and random forest. [Brownlee]

#### What makes a good feature?

**Repeated Discrete Values**

A good feature value will appear multiple times within the data set. This allows the model to learn exactly how this specific feature value relates to the label. By having multiple examples it enables the model to see the feature in more than one setting and improves its judgement for when its a good predictor for the label. [GOOGLEDEV]

**Easily distinguishable**

All features should be clearly defined with labels that make sense of the value. If the value can't be made sense of then it can lead to unclear values. Noisy data can be another cause for unclear values. The value should always be in an appropriate format in correlation to its label name. [GOOGLEDEV]

**A combination of available data**

Instead of focusing on available data for a feature, the requirements of the model should be the main priority. Often the variable should be a combination of available data to produce a variable that will make a good predictor. [TBOCK]

#### State of the art feature engineering techniques

**Pandas library**  
As the model is created using Python, the **Pandas** library is suitable for data manipulation and visualization.

**SciPy Library**

This library provides an implementation of useful statistical measures, such as:

* **SelectKBest** - This selects the top K varaibles
* **SelectPercentile** - This selects the top perciential variables  
  [Brownlee (used in Feature Engineering)]

These techniques can be used during feature selection to help select the best features.

**Binning**

This is a technique applied to prevent overfitting and make the model more robust [TDS]. It involves grouping continuous values into smaller groups referred to as bins. By grouping the continuous data into smaller groups, we can minimise the number of categories which can be useful for data visualisation. The disadvantage of this technique is its negative impact on model performance, as information is lost and the data becomes regularized [tibco][TDS].

**Filter methods**

Filter selection methods work by using statistical techniques that evaluate the relationship between all input variables and the target variable [Brownlee]. The scores obtained are used to filter the input variables to choose the variables most appropriate for the model.

**Wrapper methods**

Wrapper selection methods work by creating a number of models using different subsets of input features. The features that create the best performing model possible are chosen according to a performance metric. [Brownlee]

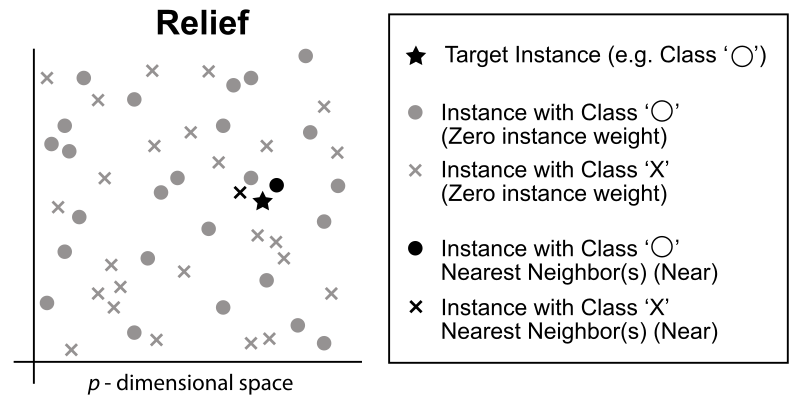
**Embedded methods**

This method works by learning which features contribute the most to the model accuracy whilst the model is beingcreated. An example of an embedded feature selection method is regularization, also known as a penalized method. This method works by introducing added constraints to the algorithm that make the model less complex. Examples of penalized methods include lasso and ridge regression. [VAISH]

**Recursive Feature Elimination (RFE)**

The RFE method can be used to select the best features. This method recursively removing attributes and building the model on the remaining attributes. By using the model accuracy, it is able to identify which particular attributes (or combination of attributes) contribute the highest when predicting the target attribute. [VAISH]

**RReliefF algorithm**The original relief algorithm was designed by Kira and Rendall in 1992 and is used for feature selection. The RReliefF algorithm is an extension inspired by the relief algorithm. It is known as a relief-based algorithm (RBA) and has been adapted to address problems with the original algorithm, such as performing more reliably in noisy environments and generalizing to solve multi-class and regression problems. [Myopia]

*An illustration of Relief neighbor selection for scoring. [Reliefwiki].*

In this algorithm, the parameters are set to 100 points which are evaluated with 10 neighbours with exponential rank correction [Bocca]. The algorithm is iterated 10 times before the results are clustered. Any features that have a negative importance are dropped and the remaining features are scaled to provide a total importance value. The total importance is regarded as the percent of the change, and features are then added until the value reaches 90% [Bocca].

### 

### Model Building and Evaluation

#### Predictive Models, Algorithms & Methods

Choosing the best model usually depends on type of data you’re using.  
Here is a number of commonly used predictive models and methods:

* **Classification model**

This is widely regarded as the simplest of predictive analyics models. It works by categorizing data based on what it learns from historical data. It provides a broad analysis which aids in decision making [LOGI].

* **Clustering Model**

This model orders the data into nested smart groups based on similar attributes. This model is popular in marketing as it can easily and quickly seperate customers into similar groups based on their characteristics, which allows them to devise a strategy for each group. This saves time as rather than going through each customer seperately, they can cluster customers into groups [LOGI].

* **Forecasting model**

This is the most widely used model for predictive analysis. This model learns from historic data to predict the values for the new data. It can be applied to any historical numerical data sets. Overall this model provides an accurate insight into how the data will look in the future, which will help decision makers make the right choices going forward. [LOGI]

* **Logistic regression**

This is used for multiclass classification. If the target labels are strings, then Scikit-learn will automatically encode the target labels. In a test for accuracy the majority class baseline was tested to be at around .54, with .73 for train and test data. [KSMITH]

* **Random Forest**

This is a classification algorithm that performs classification or regression. Like logistic regression, it also uses strings as the target label. The algorithm is a combination of decision trees, hence the name Random Forest. It uses bagging to achieve minimum errors, which means it creates subsets of data from training samples and is chosen randomly with a replacement [LOGI].  
In a test for accuracy compared to logistic regression, it achieved a majority class baseline of .79, and a .97 on train and test data, making it better than Logistic Regression. [KSMITH]

* **Ridge Regression**

This is a technique for analzying multiple regression data sets which suffer from multicollinearity. When this occurs it means the least squares estimates are not biased, however the variances are large so are likely not the true value. By adding bias, ridge regression minimises errors. It hopes to give more reliable estimates. [NCSS] Ridge regression creates a parsimonious model - this means the number of predictor variables in a given set is higher than the number of observations. It belongs in a class of regression tools that use L2 regularization. [STATISTICSHOWTO]

* **K-nearest Neighbour (K-NN)**

For each target data set, the training set predicts the variable value of interest for each member of the target group data set. It locates the closest k members for each row in the data set. It then determines the weighted sum value of the variable of interest for the closest k members. This process is then repeated for all remaining rows in the target data set. XLMiner can then build parellel models on all values of k and score the best of the models. [KNN]

* **XGBoost**

This is an open-source, optimized distributed gradient boosting software library. It implements machine learning algorithms by using a gradient boosting framework. It also provides parellel tree boosting which can be used to solve many data science problems fast and accurately. [XGBoost]

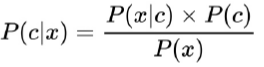
* **Sigmoid Function**

#### Machine learning models

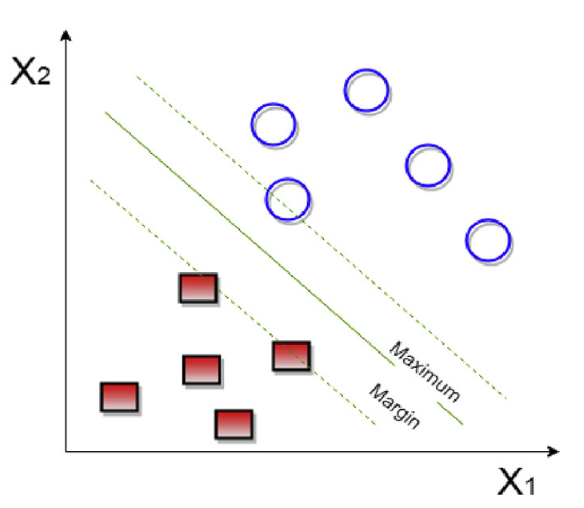
A model is fed with data which is split up into four categories: supervised learning, semi-supervised learning, unsupervised learning, and reinforcement. [AKHTAR] We are going to be evaluating each and acknowledging each.

**Supervised**

When we have both input variables and output variables, we can conduct supervised machine learning. The function (or pattern) is mapped between two parties. [AKHTAR] Here are some widely used supervised machine learning algorithms:

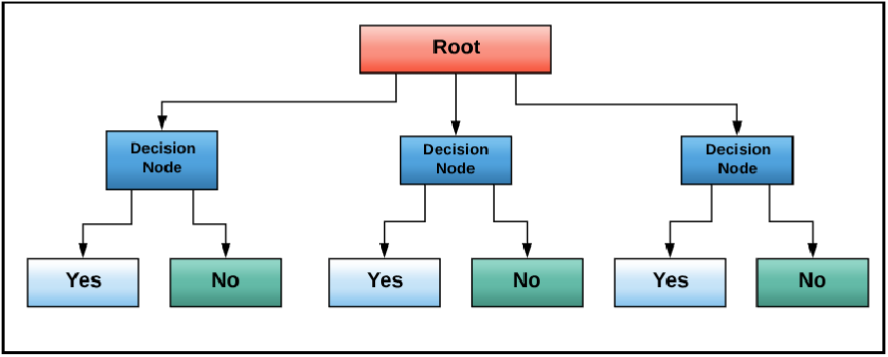
* Bayesian classifiers*[AKHTAR] Bayes Formula.*

This is mathematically based on the Bayes formula, and allows personal opinions to influence judgement, and then updating it later by using data. Naive bayes is particularly useful when using a small amount of data.

* Support Vector Machines  
    
  *[SUPPORTVECTORMACHINES] Support vector machine.*

This is a supervised machine learning model that is widely used in classification models due to it being represented in a multidimensional space. It works by finding a hyperplane between the data. The hyperplane which is most seperated from both classes will be chosen. Identifying the correct class is performed using a margin (nearest distance between hyperplanes) and the hyperplane with the largest gap is chosen. Regularization and gamma play an important role in determining the correct hyperplane. [AKHTAR]

* Decision tree

  
*[CHEBBI] Decision Tree Machine Learning.*

This model was introduced by Leo Breiman. It is a supervised machine learning algorithm that is used in decision making. It represents data as upside-down trees, with the root being at the top. Data is shown using the Iterative Dichotomiser 3 algorithm. [CHEBBI]

**Semi-supervised**

When combining a small amount of labeled and unlabeled data, this is called semi-supervised learning. This type of learning is widely used in the real-world. Some examples of its use is in protein sequence classification and web-content classification. Some semi-supervision methods include generative models, low-denisity deparation and graph-based models. [AKHTAR]

**Unsupervised Machine Learning**

This is when a machine learning algorithm detects patterns within a database where the data is unlabeled. The outcome is unknown. Whilst this type of learning cannot be used with a regression or classification problem (as the outcome values are unknown), it can be used to discover the underlying structure within the data [DATAROBOT], meaning it can find previously unknown patterns within the data.

#### Best performing models in the industry

Model 1 – Predicting English football results using machine learning classifiers

This model is capable of using machine learning algorithms such as Support Vector Machines, XGBoost and Logistic Regression to predict the match outcomes of the English Premier League. It performs a detailed study of past football matches and observes the most important attributes likely to devide the conclusion. The best algorithm is selected to obtain the target label. This is an example of a model that is applied on real team data and fixture results. [HMURUGAN]

<https://www.researchgate.net/publication/342231340_ENGLISH_FOOTBALL_PREDICTION_USING_MACHINE_LEARNING_CLASSIFIERS>

Model 2 - Predicting Football Results Using Machine Learning Techniques

This model uses machine learning techniques to predict the score and outcome of English football matches. The five main components of this model are as follows:

- Shot xG generation - this generates an expected goal value per shot and represents the probabilit of the shot resulting in a goal.

- Match xG generation - this generates a shot-based expected goals value per match.

- ELO calculation - this generates an offensive and defensive ELO rating per match, using expects goals values and the performance. The ELO ratings are recalculated at the end of each match and overall team ratings are stored ready to be used by the classification and regression models.

Once these values are calculated, it then uses classification and regression model training to train and test the model and make its predictions. [CHERBINET]

<https://www.imperial.ac.uk/media/imperial-college/faculty-of-engineering/computing/public/1718-ug-projects/Corentin-Herbinet-Using-Machine-Learning-techniques-to-predict-the-outcome-of-profressional-football-matches.pdf>

#### Methods of Evaluating Models

We can evaluate a model by comparing its performance against professional odds-makers and benchmark methods that have previously been used in past research. We can then assess the results of the classification model to get a predictive accuracy which will tell us how accurate the model is.

### Data sources available online

1. **www.football-data.co.uk**

This domain provides all the latest Premier League match data in a .CSV file type to the public for free.

As there are no pay walls to access features, so this is the best data source I have found.

Link to data: <https://www.football-data.co.uk/englandm.php?fbclid=IwAR0dXb-VFaJJjlit5MIq8oPhPOnWVQabuoxAfGJB41oted_PizNJps5afE8>

1. **www.football-data.org**

This website is good for Premier League match data and results, however some features are behind a pay wall, i.e. to access more in-depth statistics. It is subscription-based.

Link to data: <https://www.football-data.org/pricing>

1. **https://datahub.io/sports-data/english-premier-league**

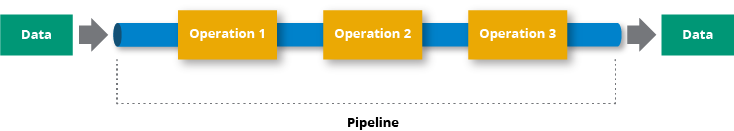
This is a good data source for historic Premier League data. It has CSV data files from Season 09/10 up until 18/19. This dataset does not include data for last season (19/20) or the current season (20/21), so it is not the most ideal option for my project.

Link to data: <https://datahub.io/sports-data/english-premier-league>

## The evolution of a DP

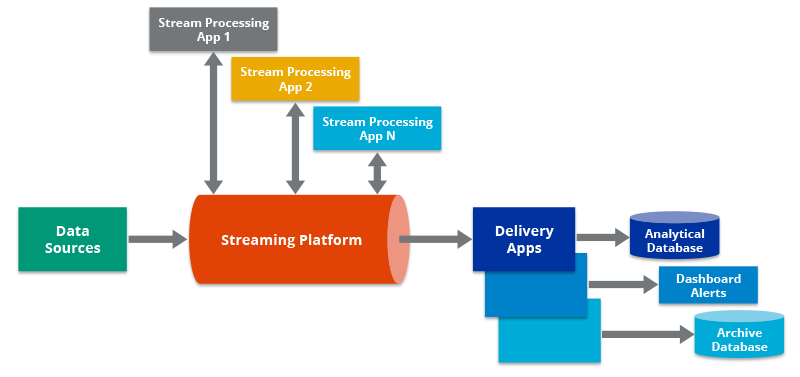
Data pipelines have evolved over time to accomodate for new needs.

To begin with, employees would have to manually update data inside tables using data entry operators [DPEVO]. As you would expect, this resulted in a lot of man-made mistakes and errors. There was also a security risk as sensitive data was being exposed to those who were updating it. The time taken to upload data was usually an overnight task, which resulted in a time lag whilst users were waiting for data to be uploaded.

  
*Illustration of a basic data pipepline [BASIC].*

This led to a new demand of needing frequent automated data uploads which removed human intervention from the process of updating data.

Modern day needs require the data pipeline to provide real-time data availability, which is appropriate for the turbulent business environment of the modern day. This has led to data pipelines uploading data to cloud services. By using the cloud environment, the data is more secure and accessible, and users don't have to view sensitive data that is not relevant to them. [DPEVO] This type of data pipeline is referred to as a **streaming data pipeline**.

  
*Illustration of a streaming data pipeline [MODEL].*

## The Benefits of a DP for Predictive Analysis

## The Use of a DP in Real World Scenarios

# Conclusion

# References

Introduction  
[McKinsey] J. Manyika et al., "Big data: the next frontier for innovation, competition, and productivity", McKinsey Global Institute, 2011.

Definition and components of a DP  
[DREMIO] "What is a Data Pipeline? - Dremio", *Dremio.com*. [Online]. Available: https://www.dremio.com/data-lake/data-pipeline/. [Accessed: 11- Feb- 2021].  
*[AUTOAI] Illustration of how AutoAI builds a machine learning model*. 2020.  
[ALTEXSOFT] Altexsoft, *The process of transporting data from sources into a warehouse*. 2019.  
[COMPUTERHOPE] "How to create a CSV file", *Computerhope.com*, 2021. [Online]. Available: https://www.computerhope.com/issues/ch001356.htm#:~:text=CSV%20is%20a%20simple%20file,Microsoft%20Excel%20or%20OpenOffice%20Calc.&text=Its%20data%20fields%20are%20most,or%20delimited%2C%20by%20a%20comma. [Accessed: 12- Feb- 2021].

SQLite3 Database

[PYTHON] "sqlite3 — DB-API 2.0 interface for SQLite databases — Python 3.9.1 documentation", *Docs.python.org*. [Online]. Available: https://docs.python.org/3/library/sqlite3.html. [Accessed: 15- Feb- 2021].  
[DBBROWSER] "DB Browser for SQLite", *Sqlitebrowser.org*. [Online]. Available: https://sqlitebrowser.org/. [Accessed: 15- Feb- 2021].

Data Cleaning

[JORBORNE] J. Osborne, *Best practices in data cleaning*. Thousand Oaks, Calif.: Sage, 2013, pp. 4-8.  
[DATACLEANING] I. Ilyas and X. Chu, *Data cleaning*, 1st ed. Association of Computing Machinery and Morgan & Claypool Publishers, 2019, pp. 1-2.

Handling Missing Data, Outlier Removal

[PIGOTT] T. Pigott, "A Review of Methods for Missing Data", *Educational Research and Evaluation*, vol. 7, no. 4, pp. 362-363, 2001. Available: https://galton.uchicago.edu/~eichler/stat24600/Admin/MissingDataReview.pdf. [Accessed 15 February 2021].  
[GREENWOOD] M. Greenwood-Nimmo and K. Shields, "An Introduction to Data Cleaning Using Internet Search Data", *Australian Economic Review*, vol. 50, no. 3, pp. 363-372, 2017. Available: 10.1111/1467-8462.12235.

Feature Engineering

[SHAH] D. Shah, J. Wang and Q. He, "Feature engineering in big data analytics for IoT-enabled smart manufacturing – Comparison between deep learning and statistical learning", *Computers & Chemical Engineering*, vol. 141, p. 106970, 2020. Available:   
<https://www.sciencedirect.com/science/article/pii/S0098135420300363#sec0016>.  
[Brownlee] J. Brownlee, "Discover Feature Engineering, How to Engineer Features and How to Get Good at It", *Machine Learning Mastery*, 2020. [Online]. Available: https://machinelearningmastery.com/discover-feature-engineering-how-to-engineer-features-and-how-to-get-good-at-it/. [Accessed: 16- Feb- 2021].  
[TBOCK] T. Bock, "What is Feature Engineering?", *ADVANCED ANALYSIS | MACHINE LEARNING | USING DISPLAYR | WHAT IS...*. .  
[GOOGLEDEV] "Representation: Qualities of Good Features", *Google Developers*. [Online]. Available: https://developers.google.com/machine-learning/crash-course/representation/qualities-of-good-features. [Accessed: 16- Feb- 2021].

State of the art feature engineering techniques

[tibco]"What is Binning?", *Docs.tibco.com*. [Online]. Available: https://docs.tibco.com/pub/spotfire/7.0.1/doc/html/bin/bin\_what\_is\_binning.htm. [Accessed: 16- Feb- 2021].  
[VAISH] P. Vaish, "The Comprehensive Guide for Feature Engineering", *A Data Analyst*. .  
[Myopia] I. Kononenko, E. Simec and M. Robnik-sikonja, "Overcoming the Myopia of Inductive Learning Algorithms with RELIEFF", *The International Journal of Research on Intelligent Systems for Real Life Complex Problems*, vol. 7, pp. 39–55, 1997. Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.56.4740&rep=rep1&type=pdf. [Accessed 16 February 2021].  
[Bocca] F. Bocca and L. Rodrigues, "The effect of tuning, feature engineering, and feature selection in data mining applied to rainfed sugarcane yield modelling", *Computers and Electronics in Agriculture*, vol. 128, pp. 67-76, 2016. Available: https://www.sciencedirect.com/science/article/pii/S0168169916306391#b0150.  
[Reliefwiki] *Relief Wiki*. 2018.

Model building and evaluation

[LOGI] S. PARTHASARATHY, "Top 5 Predictive Analytics Models and Algorithms | Logi Analytics Blog", *Logi Analytics*. [Online]. Available: https://www.logianalytics.com/predictive-analytics/predictive-algorithms-and-models/. [Accessed: 10- Feb- 2021].  
[KSMITH] K. Smith, "Predictive Modeling: Picking the best model", *Medium*, 2019. [Online]. Available: https://towardsdatascience.com/predictive-modeling-picking-the-best-model-69ad407e1ee7. [Accessed: 10- Feb- 2021]. *[NCSS] NCSS Statistical Software*. NCSS.com, pp. 335-21.  
[STATISTICSHOWTO] "Ridge Regression: Simple Definition - Statistics How To", *Statistics How To*. [Online]. Available: https://www.statisticshowto.com/ridge-regression/#:~:text=Ridge%20regression%20is%20a%20way,(correlations%20between%20predictor%20variables). [Accessed: 10- Feb- 2021].  
[KNN] "k-Nearest Neighbors (k-NN) Prediction", *solver*. [Online]. Available: https://www.solver.com/k-nearest-neighbors-k-nn-prediction#:~:text=In%20the%20k%2DNearest%20Neighbor,of%20a%20target%20data%20set.&text=For%20each%20row%20(case)%20in,neighbors)%20of%20the%20Training%20Set. [Accessed: 10- Feb- 2021].  
[XGBOOST] "XGBoost Documentation", *Xgboost.readthedocs.io*. [Online]. Available: https://xgboost.readthedocs.io/en/latest/. [Accessed: 10- Feb- 2021].  
[AKHTAR] AKHTAR, F., 2018. *BIG DATA ARCHITECT'S HANDBOOK*. PACKT Publishing Limited, p.8.  
[SUPPORTVECTORMACHINES] A. Stetco, *Support Vector Machines*. 2018.  
[CHEBBI] C. Chebbi, *Mastering Machine Learning for Penetration*. Birmingham, Mumbai: Packt Publishing, Limited, 2018, pp. 8-11.  
[DataRobot] "What Is Unsupervised Machine Learning?", *DataRobot*. [Online]. Available: https://www.datarobot.com/wiki/unsupervised-machine-learning/. [Accessed: 10- Feb- 2021].

Best Performing models in the industry

[HMURUGAN] H. Murugan, "June 2020International Journal of Pure and Applied Mathematics 118(22):533-536 Project: ENGLISH FOOTBALL PREDICTION USING MACHINE LEARNING CLASSIFIERS Authors: Anand Ganesan Harini Murugan SRMIST", *International Journal of Pure and Applied Mathematics*, vol. 118, no. 22, pp. 533-535, 2020. Available: https://www.researchgate.net/publication/342231340\_ENGLISH\_FOOTBALL\_PREDICTION\_USING\_MACHINE\_LEARNING\_CLASSIFIERS. [Accessed 13 February 2021].

[CHERBINET] C. HERBINET, "Predicting Football Results Using Machine Learning Techniques", Imperial College London, London, 2018.

Evolution of a data pipeline

[DPEVO]"Data Pipeline And Its Evolution", *Data Pipeline*, 2020. .  
[MODEL]*Model of a streaming data pipeline*. .  
[BASIC] *A basic example of a data pipeline*. .