Yi Hao CZ1016 Test Notes

Data Science Pipeline

Practical Motivation - Can you relate the problem for Data	**Sample Collection** - How to effectively sample real data
Problem Formulation - How to construct a data science problem	**Data Preparation** - How do I prepare raw data for analysis? - Data Cleaning
Statistical Description - How do I summarise/describe the data	**Exploratory Analysis** - EDA - Basic insights from the data
Pattern Recognition - Can I find insights and patterns	**Analytic Visualization** - How do I represent the data for reading
Machine Learning - How to learn from the data	**Algorithmic Optimization** - How to learn optimally from the data
Statistical Inference - How to confidently infer from the data	**Information Presentation** - How to communicate Data Analysis
Intelligent Decision - How to solve a real-life problem by data - Optimize the outcomes	**Ethical Considerations** - How to responsibly work in Data Science

Preparing Data

- **Feature Scaling:** Normalize the data if it is too skewed and scale them accordingly before building the model
- **Boxplot:** Boxplot is useful as it gives a statistical summary of the data (Quartile ranges, Outliers etc). 5-point statistic that breaks open the data in equal proportion (25%)
- **Pair plot:** Useful to express bivariate relationship when there are multiple features
- **Histogram:** Counts data in intervals, shows you the frequency. (Mean and Variance associated with it and KDE)

Analysing the Curve

- Normal Distribution: Mode = Mean = Median
- Negative Skew / Left Skew: Tail is longer to the left. Mean < Median < Mode
- Positive Skew / Right Skew: Tail is longer to the right. Mode < Median < Mean

Linear Regression

- Find the best fit line to the data points
- $y = \beta_0 + \beta_1 x + \varepsilon$ is a general formula. (β_0 is the intercept, β_{1-n} represents the coefficients of each feature, ε/J represents the cost function)
- J(a, b) is the cost function where a , b represent the parameters of the coefficients and intercepts. Gradient descent is performed to try and minimize the cost function.

- Mean Squared Error:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Drawbacks of Linear Regression

- Very sensitive to outliers in the datapoints
- Assumes Linearity relationship between the points

Measure of accuracy of model

R-Squared (Explained Var): Used to describe the proportion of variance that can be explained

$$R^2 = 1 - \frac{\text{MSE}}{\text{Var}(y)}$$

Lies between 0 < R^2 < 1. Having a **higher explained variance** implies a **stronger relationship** between the points and the line, as MSE measures the difference between our **predicted and actual value** while variance measures the difference between the **average and actual value**. Hence a better model ought to have a lower MSE which in turn leads to a higher Explained Variance.

Bias – Variance

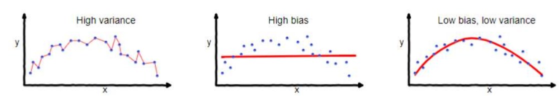
Overfitting

Generally, when the train set accuracy has an overly high R-Square, this implies **overfitting**. Our model has a **low bias but high variance** as it begins to curve too much (Linear Regression can include values of powers higher than 1).

Another way we can cause overfitting is by including too many features.

Underfitting

When we use too few features or a small train set, our model begins to have **a high bias but low variance.** This is also known as **underfitting.** The model is fails to capture and generalise the relationship of the data and performs poorly on both the train/test set.



overfitting underfitting Good balance

Classification

- Problem: Sometimes the dataset we are provided has imbalanced classes. Hence to do so we should consider down sampling or up sampling.

Decision Tree

- Tries to form partitions in the dataset based on **max depth** chosen
- At each node, the dataset is partitioned based on a certain numeric value
- Features that appear higher in the tree are implies to have greater feature importance as well
- The **response/class** is then decided based on the highest probability of the node you belong to

Measure of accuracy of model

Gini Index (Metric of Misclassification): Tries to find the probability of wrongly classifying

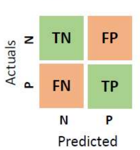
$$\text{Gini}(p_1, p_2, \ldots, p_k) = \sum_{i=1}^{k} p_i(1-p_i) = 1 - \sum_{i=1}^{k} p_i^2$$

Minimising the Gini Index implies a more accurate prediction is made.

Accuracy (Confusion Matrix)

A confusion matrix is used to express the **True Negatives, True Positives, False Negatives, False Positives** of our predicted results. TPR (Sensitivity), TNR (Specificity)

After the Model is Trained/Fit on Train Data



Classification Accuracy $acc = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}$

True Negative Rate $tnr = \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}}$

False Positive Rate $fpr = \frac{\mathrm{FP}}{\mathrm{FP}+\mathrm{TN}}$

False Negative Rate $fnr = \frac{\mathrm{FN}}{\mathrm{FN}+\mathrm{TP}}$

True Positive Rate $tpr = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$

Additionally, we should also take into account the **precision and recall**,

$$\text{Precision} = \frac{tp}{tp+fp}$$
$$\text{Recall} = \frac{tp}{tp+fn}$$

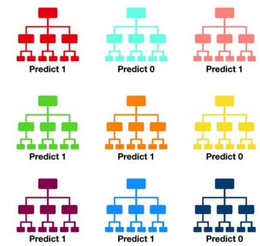
F1 Score should be used to create a **harmonic mean** as it takes both the **accuracy of precision and recall**. In reality, a large amount of our data is affected by True Negatives (TN). However, what we want to consider are the False Negatives (FN) and False Positives (FP).

Note: Having a high number of True Positives and True Negative is good, but we need to consider our False Negatives (e.g., Identifying Covid Patients as Non-Covid) as it can have serious implications

Random Forest

- Ensemble learning method that builds multiple decision trees
- Each tree uses **some randomly chosen** features and is trained on **randomly chosen** data points
- Trees are trained **parallel** to one another with no interdependence
- Final results are then collated to obtain a good decision tree

Why: Having multiple trees allow the trees to learn from each other. Additionally, this helps to reduce **overfitting (low bias, high variance).** However, higher bias can occur as each model is simpler and shallower. Furthermore, the trees are trained on fewer points. (**High bias due to part of the training data and features being used**).



Cross Validation

Split the dataset into **k partitions** of which 1 partition is used as the test set. It is a useful technique for assessing the effectiveness of your model, particularly in cases where you need to **mitigate overfitting.** This is because your model gets trained on differently on all parts of the dataset.

Example: K-Folds, Leave one out (High Variance due to high intersection of dataset)

Clustering

K-Means

- Choose K Clusters
- K-Means randomly chooses the K-Clusters (K-Means++ chooses the furthest points with equal probability to the centroid)
- For each point in the dataset, assign it to the nearest centroid
- For each cluster, compute the Within Sum of Squares (WSS)
- Recompute the new centroid (Iterate the process again until the centroids do not update)

Within Cluster Sum of Squares (WSS)

$$WSS = \sum (dist.\,frm.\,centroid)^2$$
$$= \sum\left[(x_i - \bar{x})^2 + (y_i - \bar{y})^2\right] = n\sigma^2(x) + n\sigma^2(y)$$

There are two kinds of WSS to consider, **Total WSS** and **Average WSS**. Average WSS divides the WSS by the number of points in the cluster.

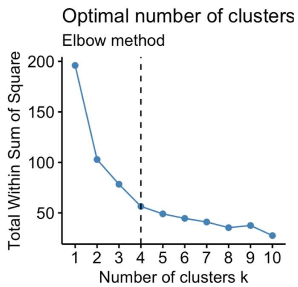
Choosing the optimal clusters:

Elbow Method: Plot the Total WSS against K-Clusters. Try to identify the point where increasing the number of clusters does not influence the Total WSS much. We will denote this K0 as the optimal number of clusters.

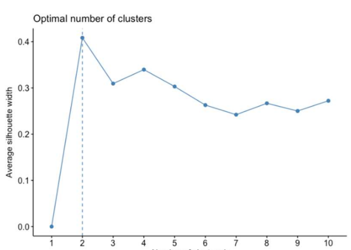
K < K0: Increasing K will change the Total WSS significantly

K > K0: Increasing K will not have much change on the Total WSS

K = K0: The point where WSS has decreased sufficiently and is at its saturation point



Average Silhouette Method: Measures the quality of a clustering. It determines how well each object lies within its cluster. A high average silhouette width indicates a good clustering.



Drawbacks of K-Means

- K-Means is highly dependent on the choice of initial clusters (Improved via K-Means++)
- Favours spherical shaped clusters due to the nature of computing centroid via Euclidean Distance

Proposed Alternatives

- DBSCAN: Density Based clustering that tries to include a point in the cluster if it is within the distance/epsilon(eps) value chosen. DBSCAN is great at **separating high density clusters from low density clusters**, however, it **struggles with clusters of similar density**.
- Hierarchical Clustering/Dendrogram: Used to join the points closer to one another in a **bottom-up approach**. Helps you to analyse how the dataset would be partitioned further and further. (Its like a cluster within a cluster)
- Expectation Maximization Algorithm (Gaussian): A form of **soft clustering** that tries to use a normal distribution to identify the clusters. Points are not fixed to one cluster but rather given a probability of being in a cluster. The mean and covariance are used to iteratively recompute the centroid. It assumes that for each data point, there is a hidden latent variable.

Anomaly Detection

Local Outlier Factor

- Choose k, number of neighbours and d, fraction of anomalies
- For each point, compute the K-nearest neighbours
- Find the reachability distance to the Kth point
- Compute the Local Reachability Distance for each point (Lower implies anomaly)
- Calculate the LOF (LOF > 1 imply an outlier, LOF < 1 imply an inlier)

Drawbacks of LOF

- LOF goes by density and hence sparse areas of points are automatically considered as clusters
- Outliers may not appear in 1-d axis
- It is not easy to decide which specific threshold determines if a point is an outlier. Only LOF < 1 is clear implication of inlier

Proposed Alternatives

Isolation Forest: Using decision trees to continually partition the dataset until an anomaly is found

Recommenders

User-Item Matrix (Each row / column can be treated as a vector)

Types of similarity: Item-Similarity, User-Similarity, Global Trends

Recommenders can be done through 2 ways, content filtering (Using similar features of items) and collaborative filtering (Using preference and tastes to find similarities).

Euclidean Distance:

- Useful for computations where magnitude matters (Movie ratings)
- Place large emphasis on distance

Jaccard Similarity:

- Useful for computations of binary variables (Either bought or never bought the product)
- Does not place any emphasis on magnitude

Cosine Similarity:

- Useful for computation of similarity where magnitude is not important (Person 1 watches the exact same movie as Person 2 but twice as much. However, they are treated to have the exact same taste)
- Place large emphasis on the angle

Must Know!

Mean: Represents the average (affected by outliers)

Median: The middle number after ordering the points (ignores outliers)

Quartiles: Measures the spread of values above and below the means by dividing the data into three points – lower quartile (Bottom 25%) – Median – upper quartile (Upper 25%).

Standard deviation: Looks at how spread out the data is from the mean

Variance: Average degree to which each point differs from the mean

Skewness: Measure of symmetry of the probability distribution of data

Pearson Correlation: Used to measure the relationship between 2 continuous variables (-1 to 1)

Boxplot: Median, Inter-quartile range, Upper quartile, Lower quartile, Whiskers, Outliers. Boxplot is useful as it gives a statistical summary of the data (Quartile ranges, Outliers etc). 5-point statistic that breaks open the data in equal proportion (25%)

Histogram: Provides a visual representation of the distribution of data. Can be used to show the skewness of data as well. (Used with **KDE Plot:** Provides a smoother estimate of the data, has greater flexibility)

Jointplot: Helps us to better visualize the correlation between 2 numeric variables

Heatmap: It can express correlation (-1 to 1) and gives colours to extreme value which make it visually easier to interpret

Structured Data: Organized, Easy to Read, Quantitative (Numeric, Classes)

Unstructured Data: Unorganized, Qualitative (Audio, Visual, Texts)

Supervised Learning (Regression, Classification):

The dataset is labelled. Outputs and inputs are known to the user. Hence, the model can measure its accuracy and try to adjust in order to iteratively improve

Unsupervised Learning (Clustering, Anomaly Detection):

Output is not known. Input data is not known. Generally, less accurate.