

1. Modeling

a. Bias and Variance

- i. Bias: how close the model's predicted value come to the true underlying $f(x)$ values, with smaller being better
- ii. Variance: the extent to which model prediction error changes based on training inputs with smaller being better
- iii. High bias but low variance:
 1. Oversimplified the situation
 2. The predicted values are frequently off from the true value
 3. Linear regression
- iv. Low bias but high variance
 1. Overfitting
 2. The predicted values closer to the true value
 3. The predictions varying wildly based on the input features

b. How to find outlier

i. Definition of outliers

1. Rare items, events or observation which raise suspicions by differing significantly from majority of the data

ii. Methods

1. Interquartile Range: $Q1-1.5IQR$
2. Visualization

c. How to handle missing value

i. Types of missing value

1. Missing Completely at Random (MCAR)

- a. When data are missing is not related to either the specific value which is supposed to be obtained or the set of observed responses.
- b. The probability of a missing data value is independent of any observation in the dataset
- c. Remains unbiased

2. Missing at Random (MAR)

- a. the missing and observed observations are no longer coming from the same distribution
- b. eg: men are less likely to fill in a depression survey, but this has nothing to do with their level of depression, after accounting for maleness.

ii. Method

1. Prevent the problem by well-planning the study and collecting data carefully
2. Deleting missing data if MCAR assumption is satisfied
3. Mean substitution
4. Regression imputation

- a. MICE (Multiple imputation by chained equation):
 - i. Steps1: start by filling in the missing data with plausible guesses at what the values might be (mean/median)
 - ii. Steps2: for each variable, predict the missing values by modeling the observed values as a function. At each step, update the predictions of the missing values.
 - d. How to handle imbalanced data
 - i. Definition: refers to those types of datasets where the target class has an uneven distribution of observations.
 - ii. Methods
 1. Design your own cost function that penalizes wrong classification of the rare class more than wrong classifications of the abundant class
 2. Resample classes by running ensemble models with different ratios of the classes, or by running an ensemble model using all samples of the rare class and a differing amount of the abundant class.
 3. Proper Evaluation Metric
 - a. Accuracy may be good enough for balance data but not imbalance one.
 - b. F1 is a more appropriate metric.
 - c.
$$\frac{2 * Recall * Precision}{Recall + Precision}$$
 4. Resampling
 - a. Oversampling: Oversample the minority class using replacement
 - b. Undersampling: randomly delete rows from the majority class to match them with the minority class
 5. Synthetic Minority Oversampling Technique (SMOTE)
 - a. Logics behind: Simply adding duplicate records of minority class often don't add any new information to the model
 - b. SMOTE looks into minority class instances and use k nearest neighbor to select a random nearest neighbor, and synthetic instance is created randomly in feature space.
 - c. Steps:
 - i. For every observation x in minority set, determine k neighbors around xi.
 - ii. Randomly select one from k neighbors
 - iii. Generate a random number between 0 and 1 and use the formula below.
 - d.
$$x_{new} = x + rand(0,1) * (x_{neighbor} - x)$$
 6. Threshold Moving

- a. We assign those prediction's probabilities to a certain class based on a threshold which is usually 0.5.
 - b. $\frac{y'}{1-y'} = \frac{y}{1-y} * \frac{m^-}{m^+}$
- e. Model evaluation to understand the characteristics and application of each metrics
 - i. Cross-Validation, stratified cross-validation
 1. Cross-Validation (k-fold):
 - a. Definition: Cross-validation is a resampling method that uses different portions of the data to test and train a model on different iterations.
 - b. Divide dataset into k exclusive sets and every time using k-1 as training set and the rest one as the testing set.
 2. stratified cross-validation
 - a. To ensure that each fold has the same proportion of observations with a given categorical value. Such as the class outcome value.
 3. Leave-one-out cross-validation (LOOCV)
 - a. K is equal to the size of the dataset (n). That is, it is where the model is testing on every single data point during the cross-validation.
 4. Standard k-fold CV can't not be applied to time-series data since the time-series data is not randomly distributed but instead is already in chronological order. You should use historical data up until a given point in time and vary that point in time from the beginning till the end.
 - ii. MSE, MAE, impurity function, cross-entropy, precision, recall, AUC, ROC, F1
 1. MSE: Mean Square Error
 2. MAE: Mean Absolute Error
 3. Impurity Function:
 - a. Gini Index:

$$Gini(D) = 1 - \sum_{k=1}^{|y|} p_k^2$$

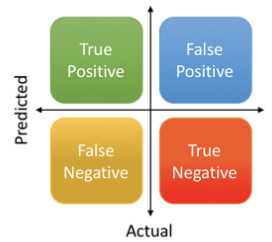
$$Gini_index(D, a) = \sum_{v=1}^V \frac{|D^v|}{|D|} Gini(D^v)$$
 4. Cross-entropy
 5. Precision (correctly predicted positive to total predicted positive observation)
 - a. $\frac{True\ Positive}{True\ Positive + False\ Positive}$
 6. Recall (correctly predicted positive to total actual positive observation)

$$a. \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

$$\text{Precision} = \frac{\text{True Positive}}{\text{Actual Results}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{Predicted Results}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{Total}}$$

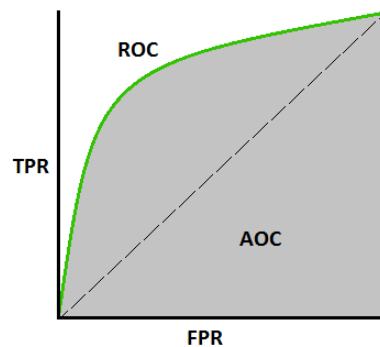


7. AUC

8. ROC

$$a. \text{TPR} = \frac{TP}{TP+FN} = \text{Recall}$$

$$b. \text{FPR} = \frac{FP}{TN+FP}$$



9. F1

$$a. \frac{2 * \text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}}$$

- f. False positive and false negative: give example where false positive is more important than false negative
 - i. The patient may be diagnosed with diabetes when they actually do not have the disease. This is a false positive. This can lead to unnecessary medical treatment.
- g. Hyperparameter Tuning
- h. How to choose a feature
 - i. PCA
- i. Overfitting, underfitting respective performance and solution
 - i. Overfitting
 - 1. model tries too hard to capture the noise in the training set
 - 2. High Variance
 - ii. Underfitting

1. Model is unable to capture the relationship between input and output accurately, generating a high error rate on both training set and unseen data.
2. High Bias
- j. Variance/bias trade-off
 - i. Variance:
 1. Variability of model prediction for a given value which tells us the spread of our data
 2. High variance -> pays a lot of attention to training data and does not generalize on the data
 - ii. Bias
 1. Definition: Difference between the average prediction of model and correct value
 2. High bias -> oversimplified model
- k. Explain gradient descent, stochastic gradient descent, mini-batch gradient descent
 - i. Gradient Descent
 1. Definition: an iterative first-order optimization algorithm used to find a local minimum of a given function
 - a. $x^{t+1} = x^t - \alpha \nabla f_i(x)$
 2. Limitation
 - a. Calculating derivatives for the entire dataset is time consuming
 - b. Memory required is proportional to the size of dataset
 - ii. Stochastic Gradient Descent
 1. Definition: the algorithm calculates the gradient for one observation picked at random, instead of calculating the gradient for the entire dataset.
 2. SGD can obtain an unbiased estimate of the true gradient without going through all data points by uniformly selecting a point at random and performing a gradient update then and there.
 - a. $\nabla f(x) = \frac{1}{n} \sum_{i=1}^n \nabla f_i(x)$
 3. SGD is useful when redundancy in the dataset is present.
 4. Pros:
 - a. Speed up
 5. Cons:
 - a. Never reach the local minimum but dance around it
 - iii. Mini-batch Gradient Descent
 1. We use a batch of a fixed number of training examples which is less than the actual dataset and call it a mini batch.
 2. Calculate the mean gradient of the mini-batch
 3. Still fluctuating
1. Difference between statistical learning and machine learning

- i. Statistical Learning is math intensive which is based on the coefficient estimator and requires a good understanding of your data. On the other hand, Machine Learning identifies patterns from your dataset through the iterations which require a way less of human effort.
 - ii. Machine learning models are designed to make the most accurate predictions possible. Statistical models are designed for inference about the relationships between variables.
- m. Spherical hashing
 - i. Parametric/ Non-parametric model
 - 1. Parametric
 - a. Algorithms that simplify the function to a known form are called parametric
 - b. Steps
 - i. Select a form for the function
 - ii. Learn the coefficients for the function from the training data
 - c. Examples
 - i. Logistic Regression
 - ii. Linear Discriminant Analysis
 - iii. Perceptron
 - iv. Naïve Bayes
 - v. Simple Neural Networks
 - d. Pros
 - i. Simpler, Speed, Less Data
 - e. Cons
 - i. Constrained, Limited complexity, Poor Fit
 - 2. Non-parametric
 - a. Algorithms that do not make strong assumptions about the form of the mapping function are called nonparametric ML algo
 - b. Examples
 - i. K-Nearest Neighbors
 - ii. Decision Trees
 - iii. SVM
 - c. Pros
 - i. Flexibility, Power (No assumptions), Performance
 - d. Cons
 - i. More Data, Slower, Overfitting
 - ii. Generative/ Discriminant model
 - 1. Generative Model
 - a. Definition: models the actual distribution of each class.
 - b. Learns the joint probability distribution $p(x,y)$. It predicts the conditional probability with the help of Bayes Theorem.

c. Steps

- i. Assume some functional form for $P(Y)$, $P(X|Y)$
- ii. Estimate parameter of $P(X|Y)$, $P(Y)$ directly from training data
- iii. Use Bayes rule to calculate $P(Y|X)$

d. Example

- i. Naïve Bayes
- ii. Bayesian Networks
- iii. Markov Random Fields
- iv. HMM

2. Discriminant model

a. Definition: models the decision boundary between the classes.

b. Learns the conditional probability distribution $p(y|x)$

c. Steps

- i. Assume some functional form for $P(Y|X)$
- ii. Estimate parameters of $P(Y|X)$ directly from training data

d. Example

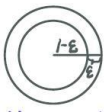
- i. Logistic Regression
- ii. SVM
- iii. NNW
- iv. Nearest Neighbor

iii. Curse of dimension

1. Definition: refers to a set of problems that arise when working with high dimensional data.

2. Sparsity

- a. As the number of attributes or the dimensions increases the number of training samples required to generalize a model also increases phenomenally.
- b. The available training samples may not have observed targets for all combinations of the attributes. This is because some combination occurs more often than others. Due to this, the training samples available for building the model may not capture all possible combinations.



$r=1$,
 $V_{\text{shell}} = k \cdot 1^D = k$
 $V_{\text{shell}} = V_{\text{out}} - V_{\text{in}} = k - k \cdot \epsilon^D$
 $\frac{V_{\text{shell}}}{V_{\text{out}}} = \frac{k - k \cdot \epsilon^D}{k} = 1 - \epsilon^D$
 $0 < \epsilon < 1$ $\lim_{D \rightarrow \infty} (1 - \epsilon^D) = 1$
 $\lim_{D \rightarrow \infty} \frac{V_{\text{shell}}}{V_{\text{out}}} = 1$

大多数的数据点

- c. Training a model with sparse data could lead to high-variance or overfitting conditions.

3. Solution

- a. Dimension Reduction: PCA

2. Regression

- a. The basic assumptions of linear regression, what to do when the basic assumptions are violated
 - i. Linearity: The relationship between X and the mean of Y is linear
 - 1. If not linear: apply a nonlinear transformation to independent and dependent variable. Like taking the log, sqrt, reciprocal.
 - ii. Homoscedasticity: the variance of residual is the same for any value of X
 - 1. How to test: plot the fitted value vs. residual plot
 - 2. Simply take the log of the dependent variable
 - 3. Use weighted regression
 - iii. Independence: Observation are independent of each other
 - 1. Mostly relevant when working with time series data
 - 2. How to test: plot the residuals vs. time
 - 3. Adding lags, adding seasonal dummy variables
 - iv. Normality: The residuals of the model is normally distributed
 - 1. How to test: QQ plots, Shapiro-Wilk test
 - 2. Verify any outliers aren't having a huge impact on the distribution
 - v. If one or more of these assumptions are violated, then the results of our linear regression may be unreliable or misleading
- b. How to measure covariance, VIF
 - i. Covariance: a measure of how changes in one variable are associated with changes in a second variable
 - ii.
$$cov(X, Y) = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{n}$$
- c. Comparison of correlation and causation, how to measure each
 - i. Correlation
 - 1. The statistical indicator of the relationship between variables
 - 2.
$$r = \frac{cov(X, Y)}{S_x S_y}$$
 - ii. Causation
 - 1. means that changes in one variable brings about changes in the other; there is a cause-and-effect relationship between variables.
 - 2. Hypothesis testing and A/B testing
- d. Linear regression, how to change the model when performing various linear transformation on the data, how to change the predictive value, R-squared, coefficients, etc.
- e. Why the sum of residuals is zero under OLS

- i. Ordinary Least Square estimator minimizes the sum of squared residuals.

$$\frac{\partial}{\partial a} \sum_{i=1}^n (y_i - a - bx_i)^2 \Big|_{(\hat{a}, \hat{b})} = 0 \Rightarrow -2 \sum_{i=1}^n (y_i - \hat{a} - \hat{b}x_i) = 0$$

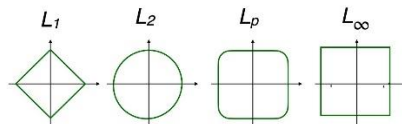
ii.

- f. How to determine how well the model fits based on residual plot and QQ-plot
- g. The potential points that I can think of that have not been tested
 - i. How to estimate the parameters of logistic regression
 - ii. The form of the LOSS function of logistic regression
 - iii. Why OLS estimation is used in linear regression, and some properties of OLS estimators

3. Regularization

- a. Definition and main function
 - i. Avoid overfitting by constraining or shrinking the coefficient estimates towards zero (penalize the flexibility of our model)
- b. Comparing Lasso and Ridge
 - i. Lasso is more likely to set parameter lamda to 0, which means we can filter out some unnecessary features, so we can get a sparse model.
 - ii. While Ridge will be more likely to yield a smoothing model.
 - iii. Why? On L1, w^* is more likely to be tangent on the vertex of the square, which means that some of the w is 0.

The L_p Norm



Approximation Theory 3

- c. Are the results of Lasso the same for different programming languages?
 - i. No, because the grid is not the same
- d. L1 norm and L2 norm
 - i. $L_1: \sum_{r=1}^n |x_r|$
 - ii. $L_2: \sqrt{\sum_{r=1}^n |x_r|^2}$
 - iii. $L_p: (\sum_{r=1}^n |x_r|^p)^{1/p}$
- e. Are the estimated coefficients of Regularization unbiased?
 - i. ??

4. Tree & Ensemble

a. Explain the tree model

- i. It is a supervised machine learning which performs classification and regression tasks by building tree-like structure for deciding the target variable class or value according to the features

- ii. Entropy:

$$Ent(D) = - \sum_{k=1}^{|y|} p_k \log_2 p_k$$

- iii. ID3 Tree

1. Gain:

$$Gain(D, a) = Ent(D) - \sum_{v=1}^V \frac{|D^v|}{|D|} Ent(D^v)$$

- a. V: 对 attribute a 进行划分后有 V 个分支
 - b. Choose the a with the greatest Gain

- iv. C4.5 Tree

1. Gain Ratio:

$$Gain_{ratio}(D, a) = \frac{Gain(D, a)}{IV(a)}$$

2. IV(a):

$$IV(a) = - \sum_{v=1}^V \frac{|D^v|}{|D|} \log_2 \frac{|D^v|}{|D|}$$

3. Using Gain as measure will tend to choose an attribute with large V, for example ID.
 4. Still when using C4.5 tree, choose attributes with Gain higher than average Gain then choose the one with the highest Gain Ratio.

- v. CART Tree

1. Gini Index:

$$Gini(D) = 1 - \sum_{k=1}^{|y|} p_k^2$$

$$Gini_index(D, a) = \sum_{v=1}^V \frac{|D^v|}{|D|} Gini(D^v)$$

2. We choose the attribute which yields the least Gini Index after splitting.

b. Explain the random forest model and compare it with the boosting model (GBT is more commonly tested)

- i. Bagging (Random Forest)

1. Definition:

- a. Models run in parallel and are independent of each other.
 - b. Training set is generating with selecting observations with replacement.

- c. Every model has the same weight.
 - ii. Boosting
 - 1. Definition:
 - a. Models have different weight based on their performance.
 - 2. Gradient Boosting Decision Tree
 - a. Building tree based on the residual calculated by subtracting predicted value from true value in the last decision tree.
- c. The adjustable parameters of the random forest and GBT models in the programming language
 - i. Random Forest
 - 1. n_estimators: number of trees
 - 2. criterion: splitting criterion (Gini, Entropy, log_loss)
 - 3. max_depth: maximum depth of the tree
 - 4. bootstrap: If set to False, the whole dataset is used to build each tree
 - 5. min_samples_leaf: The minimum number of samples required to be a leaf node.
 - ii. GBT (Gradient Boosting Regressor)
 - 1. Loss: loss function (squared_error, absolute_error)
 - 2. Learning_rate:
 - 3. n_estimators: number of boosting stages
 - 4. criterion: splitting criterion (friedman_mse, squared_error, mse)
 - 5. max_depth: maximum depth of the individual regression estimators
- d. You should know that each tree of random forest is better to make deeper because random forest is more suitable for low bias high variance; each tree of boosting model should not be too deep
- e. What is the most preferred model? Why?
- f. Know the advantages and disadvantages of each model, what is applicable, what data, complexity and computational effort related to what.
 - i. Adaboost
 - 1. Definition:
 - a. Every Learner uses the same dataset
 - b. Learners are connected sequentially and are dependent on each other (same algo mostly)
 - c. Focus on decreasing the Bias but sensitive variance (数据扰动造成的影响)
 - 2. Pros:
 - a. Good at solving hard problems (high ceilings)
 - 3. Cons:
 - a. Overfitting
 - b. Too sensitive to outliers

- c. Slow
- ii. Random Forest
 - 1. Definition
 - a. Using Bootstrap Sampling (取出又放回) to get the training set for each learner. The size of training set is equal to that of dataset.
 - b. Randomly choose k attributes from all the attributes and choose the best one (Gain/ Gain Ratio/ Gini Index) for the root node.
 - 2. Pros:
 - a. Not likely to overfit because every learner uses different training set (not sensitive to outliers)
 - b. Quicker when dealing with high dimensional data because only using some attributes
 - c. Tree-structure is more easily to explain
 - 3. Cons:
 - a. Too general (Low ceiling)
 - 4. Out-of-bag Score (OOB score)
 - a. Out-of-bag sample
 - i. While making the samples, data points were chosen randomly and with replacement, and the data points which fail to be a part of that particular sample known as Out-of-bag points
 - b. OOB score
 - i. OOB score is a way of validating the RF model. Find all the tree models using without data point p1 and calculate accuracy rate which p1 is correctly classified.
 - ii. Pros:
 - 1. Compared to using validation set, OOB only uses the data points in training set.
 - iii. Cons:
 - 1. Time-consuming

5. KNN

- a. Please explain KNN and then write out the code for its implementation
 - i. The k-nearest neighbors (KNN) algorithm is a data classification method for estimating the likelihood that a data point will become a member of one group, or another based on what group the data points nearest to it belong to.

ii. K:

1. Small k will result the noise will have a higher influence on the result.
2. A large value makes it computationally expensive.
3. Simple approach $k = n^{1/2}$

6. K-MEANS

a. Please explain K-means and then write out the code for its implementation

i. K-means clustering is one of the simplest unsupervised machine learning algorithms

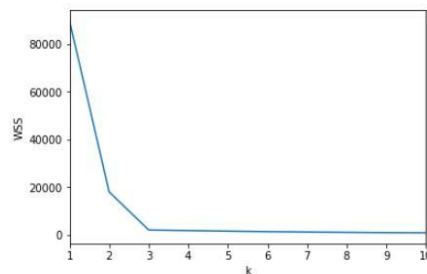
ii. Steps:

1. Choose K
2. Select K random points from dataset as centroids
3. Assign each observation to the cluster with nearest mean (least squared Euclidean Distance).
4. Update: Recalculate means for observations assigned to each cluster.

b. How to choose k

i. The Elbow Method

1. Calculate the Within-Cluster-Sum of Squared Errors (WSS) for different values of k, and choose the k for which WSS becomes first starts to diminish. In the plot of WSS-versus-k, this is visible as an elbow.



2. The Silhouette Method

a. The silhouette value is a measure of how similar an object is to its own cluster (cohesion) compared to other clusters (separation).

$$b. s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \text{ if } |C_I| > 1$$

$$i. s(i) = 0, \text{ if } |C_I| = 1$$

$$c. a(i) = \frac{1}{|C_I| - 1} \sum_{j \in C_I, i \neq j} d(i, j)$$

- i. The average distance between I and all the other data points in the cluster to which o belongs. (Cohesion)
- d. $b(i) = \frac{1}{|C_J|} \sum_{j \in C_I} d(i, j) = \min(a(i))$
 - i. The minimum average distance from i to all clusters to which i does not belong. (Separation)
- e. A high Silhouette Score is desirable and it reaches its global maximum at the optimal k
- c. How to measure the results (unsupervised learning, I guess interviews tend to want to hear some collaboration with domain people)
 - i. Drawbacks
 - 1. Can't handle complicated geometric shapes.

7. SVM

- a. Please explain SVM, (it seems that any model may "explain the model")
 - i. The objective of SVM is to find a hyperplane that distinctly classifies the data points
 - ii. To find a plane that has the maximum margin
- b. What is support vector
 - i. Data points that are closer to the hyperplane and influence the position and orientation of the hyperplane
- c. Explain the kernel trick, why it kernels matrix is positive definite
 - i. Not all the data are linearly separable. Higher dimensional transformations can allow us to separate data in order to make classification predictions.
 - ii. However, when there are more and more dimensions, computations within the space become more and more expensive.
 - iii. Kernel Tricks: it allows us to operate in the original feature space without computing the coordinates of the data higher dimensional space. In essence, what the kernel trick does for us is to offer a more efficient and less expensive way to transform data into higher dimensions.
- d. What does the complexity of the SVM depend on, the sample size or the number of variables?
 - i. Complexity: number of examples, number of features, type of kernel function and the regularization parameter.
- e. Explain some important parameters of SVM model
 - i. C: Regularization parameter
 - ii. Kernel: Kernel method
 - iii. Tol: stopping criterion

8. ML related algorithm implementation

- a. Write a KNN algorithm
- b. Write a KMeans algorithm
- c. Write a mini-batch gradient descent function

9. NLP related

- a. According to the different interview positions, for some positions ML model is built on text data, so NLP can be added points
- b. Some basic concepts such as
 - i. BOW model, N-gram model
 - ii. Term matrix, TFIDF
 - iii. Stemming, part-of-speech, NET
 - iv. Word2Vec, Topic modeling
- c. Computational linguistic, just ask how to design rules to extract company names given a dirty data

10. Summary

- a. Practice manual implementation of ML algorithms, KNN, K-Means better write and often test; Naive Byes can also write a logistic regression/ linear regression can also use gradient descent write, a friend was tested EM algorithm implementation.