**INTRODUCTION**

**Types of Learning**

* **Supervised Learning**
  + For each observation of predictor measurement Xi, there is a response measurement Yi
  + We wish to fit a model that relates Xi and Yi with the aim of accurately predicting response for the future (prediction) or better understand their relationship (inference)
  + Examples - Linear Regression, GAM, SVM
* **Unsupervised Learning**
  + For each observation i, we have input measurement Xi but no associated response Yi
  + We seek to understand the relationship between the variables or observations
  + Example -Clustering
* **Semi-Supervised Learning**
  + Consider n observations out of which, for m observations we have Xi and Yi. For (n-m) , we only have Xi
  + m observations 🡪 Xi ~ Yi 🡪 Supervised
  + (n-m) observations 🡪 Xi 🡪 Unsupervised

**Statistical Learning**

e 🡪 random error independent of X

🡪 Variance = σ2

🡪 Mean (e) = 0 : If mean is not zero, then it can be broken into e’ = e + c where c is a

constant. This constant will then be absorbed into the intercept.

Essentially, the model shifts while keeping the slope / R2 constant.

**Data Generating Process**

* Data Generating Process (DGP) is the mechanism/process at work in the real-world giving rise to the sample. It is the true description of Y=f(X) + e.
* A model is a set of all possible DGPs. The data is used to help pick the correct/ relevant DGP by estimating f().

**Prediction**

* In many situations, a set of inputs X are readily available, but output Y cannot be easily obtained. In this setting, we estimate f() as f’()
* This is used as a black box as we are concerned with its form as long as it fields accurate predictions for Y.
* As error term averages to zero,



* f’() will not be the perfect estimate of f(). This inaccuracy will introduce some error. This s the reducible error as it can reduced by better estimating f().
* As Y is also a function of e, there will still be some error that cannot be reduced. This is the irreducible error. Hence, the variability of e introduces the irreducible error.

Text, letter

Description automatically generated

**Inference**

* We are often interested in understanding the way Y is affected as X1, X2, X3 … Xp change.
* We wish to estimate f() but the goal is not to predict Y but to understand the relationship between and Y. Hence, f() can no longer be used as a black box.
* What predictors are associated with the response? 🡪 Feature Selection
* What is the relationship between Y and X? 🡪 Y = k.Xp , Y = k.log(Xp) , etc.
* Is the relationship linear or non-linear?

**Parametric Method to estimate f()**

**STEP 1** : Make an assumption about the functional form of f(). One simple assumption would be that f() is linear in Y.

**STEP 2** : Use training data to fit the model. In case of Linear Regression, we estimate coefficients β1 , β2 …. Βp using OLS.

* This model-based approach reduces the problem of estimating f() down to one that of estimating a set of parameters.
* Disadvantages :
  + The model we choose may not match the true unknown form of f()
  + Usage of a highly flexible model with more parameters leads to overfitting as the model follows the noise too closely.

**Non-Parametric Approach to estimate f()**

* Non-Parametric methods do not make an explicit assumption about the functional form of f().
* Instead, they seek an estimate of f() that gets close to the data points without getting too rough or too wiggly.
* Advantage :
  + By avoiding the assumption of a particular form of f(), they have a potential to accurately fit a wider range of possible shapes of f().
* Disadvantage :
  + As they do not reduce the problem of estimating f() to a small set of parameters, a very large number of observations are required to estimate f().

**Model Interpretability**

* Some models are less flexible (restrictive) as they produce relatively small number of shapes to estimate f().
* Inference 🡪 Restrictive models are much more interpretable
* Prediction 🡪 Interpretability is not an issue and hence more flexible models can be used

Graphical user interface, text, application, email

Description automatically generated

**ASSESSING MODEL ACCURACY**

**Measuring the Quality of Fit**

* A picture containing watch, clock

  Description automatically generatedIn a regression setting, the most commonly used measure of quality of fit is the Mean Squared Error (MSE).
* MSE is small if the predicted responses are very close to the true responses.
* Suppose we fit a model on training observations [ (x1,y1) , (x2,y2) , …… (xn,yn) ] and obtain and estimate f’(). We can then compute f’(x1) , f’(x2) … f’(xn). The Mean Squared Error associated with this will be the Train MSE.
* However, we are not interested with the Train MSE. Instead, we want to know whether f’(x0) is approximately equal to y0 where (x0,y0) is a previously unseen test observation. This gives the Test MSE.
* We want a model that gives the lowest Test MSE as opposed to the lowest Train MSE.
* If we have a large number of observations in the data, we select the model with the lowest average squared prediction error for the test observations which is given by:



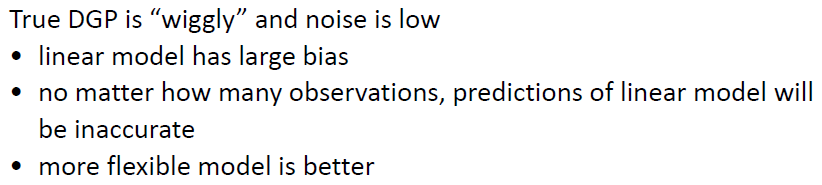
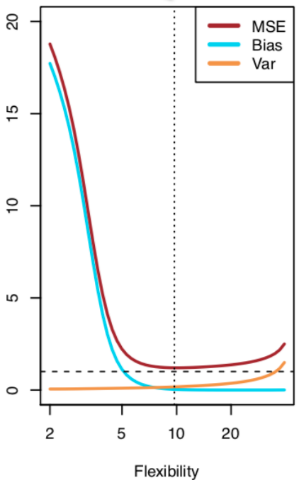
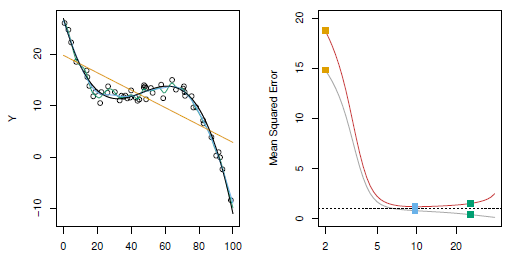
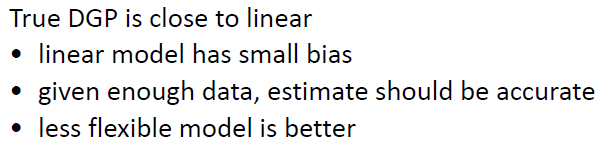
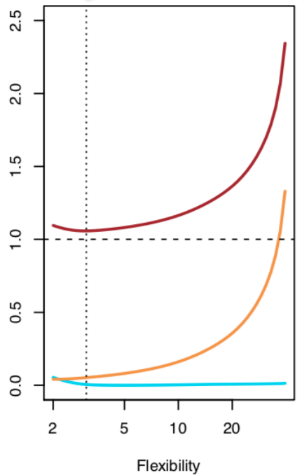
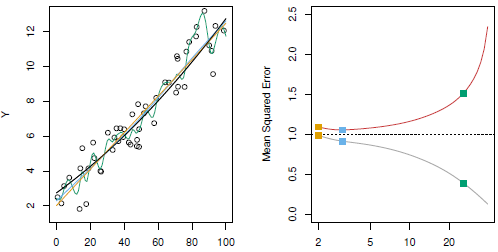
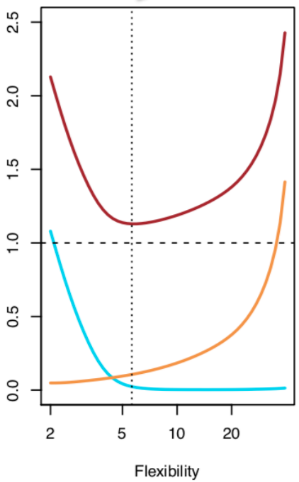
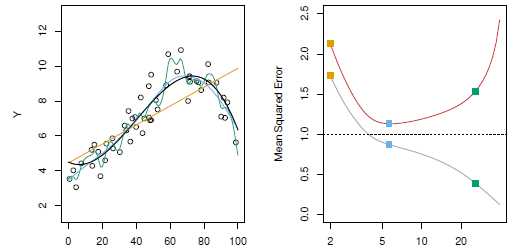
* Simple selecting a model with the lowest Train MSE is a wrong approach as low Train MSE does not correspond to low Test MSE. This happens because most models specifically estimate coefficients so as to minimise the Training MSE.
* As model flexibility increases, the Train MSE will decrease but Test MSE may not.
* When the model gives low Train MSE and high Test MSE, the model is said to be overfitting as it is now trying to find patterns in the training data too closely and may be picking up some patterns that are created by random chance.

**Bias – Variance Trade Off**

* The expected Test MSE can be broken down into 3 fundamental quantities :



* In order to reduce the expected Test MSE, we need a model that simultaneously achieves low variance and low bias.
  + Variance is inherently non-negative
  + Squared Bias is non-negative
  + Var(e) represents the irreducible error
* Even if we achieve zero variance and zero bias, we will still have the irreducible error. Hence, Test MSE cannot lie below Var(e).
* Variance refers to the amount by which f’() will change if we estimated it using a different training dataset.
  + Since training data is used to fit the model, different datasets will result in different f’().
  + If a method has high variance, then small changes in the training data can result ion large changes in f’().
  + More flexible models have high variance.
* Bias refers to the error introduced in f’() by approximating a complicated DGP with a simple f().
  + For example, fitting a Linear Regression model to a real-world non-linear dataset.
  + More flexible models have low bias.
* As model flexibility increases, the variance will increase while the bias will decrease.
  + Bias tends to decrease at a faster rate than variance.
    - This reduces the Test MSE.
  + As some point, increasing flexibility has little impact on bias but it starts significantly increasing the variance.
    - This starts to increase the Test MSE.



**Model Accuracy in Classification Setting**

* We now seek to estimate f() on the basis of training observations [ (x1,y1) , (x2,y2) , …… (xn,yn) ] where y1, y2 …. yn are now qualitative.
* The most common approach for measuring the accuracy is the Train Error Rate. It is the proportion of incorrect classifications.

Diagram, schematic

Description automatically generated

* We are more interested in the error rates that result from applying the classifier f’() to test observations. This is the Test Error Rate.

Text

Description automatically generated with low confidence

**K-Nearest Neighbours (KNN)**

k-Nearest neighbour is an example of **instance-based learning**, in which the training data set is stored, so that a classification for a new unclassified record may be found simply by comparing it to the most similar records in the training set

**Distance Function :**

* Property 1 : distance is always non-negative
* Property 2 : distance is commutative i.e. distance between A,B is same as B,A

A picture containing text

Description automatically generated

* A picture containing text, clock, watch

  Description automatically generatedProperty 3 : Triangular inequality hold true

**A picture containing text, clock, watch

Description automatically generatedEuclidean Distance :**

* When measuring distance certain attributes that have large values can overwhelm the influence of other attributes that are measured on a smaller scale. To avoid this, the data needs to be either **normalized or standardized**.

**Categorical Variables** :

* Cannot use Euclidean Distance for categorical variables
* A picture containing text, clock, gauge

  Description automatically generatedDefine a **Different Function**

**Table

Description automatically generatedExample** :

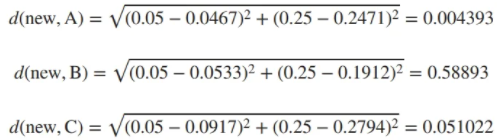
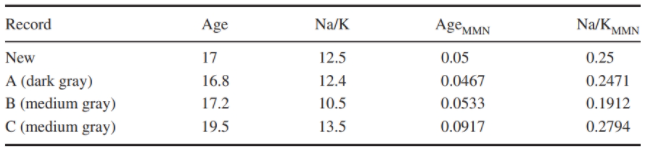
Assume that the age variable has 10 as minimum, 45 as mean, 15 as standard deviation and 50 as range.

* Age not normalized or standardized –
  + d(A,B) = sqrt( (50-20)2 + (0)2 ) = 30
  + d(A,C) = sqrt( (50-50)2 + (1)2 ) = 1
  + C is closer to A
* Age normalized –
  + d(A,B) = sqrt( (0.8-0.2)2 + (0)2 ) = 0.6
  + d(A,C) = sqrt( (0.8-0.8)2 + (1)2 ) = 1
  + B is closer to A
* Age standardized –
  + d(A,B) = sqrt( (0.33+1.67)2 + (0)2 ) = 2
  + d(A,C) = sqrt( (0.33-0.33)2 + (1)2 ) = 1
  + C is closer to A

The distance(x,y)and Min-Max Normalization functions produce values in the range [0, 1]. Hence, when calculating the distance between records containing both numeric and categorical attributes, the use of **Min-Max Normalization is preferred**.

**Combination Function** :

* After the distances have been calculated, they need to be combined to classify new data points
* Simple Unweighted Voting –
  + Step 1 : Decide on the value of K i.e. number of neighbours that will take part in the voting process
  + Step 2 : Compare the new record to and find K neighbours that have the minimum distance
  + Step 3 : Distance no longer matters for unweighted voting, each neighbour has equally weighed vote
  + As equal weightage is given to all neighbours, it **may lead to tie in case of even K**.
* Weighted Voting –
  + Closer neighbours have a larger voice in the classification decision than do more distant neighbours.
  + Influence of a particular record is inversely proportional to the distance of the record from the new record to be classified
  + Weighted voting also **makes it much less likely for ties to arise**.
  + Weighted Voting Example:

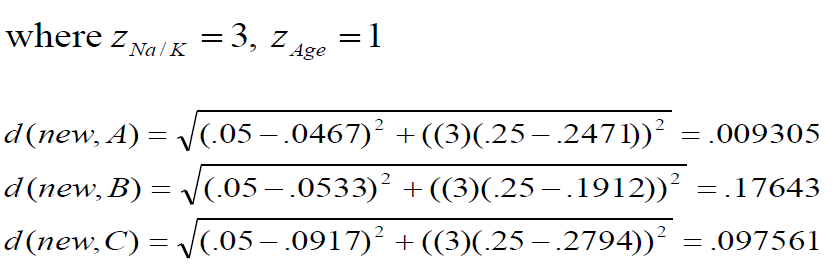
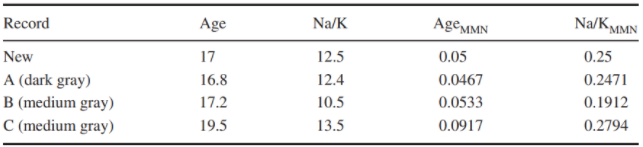


**Balanced Dataset** :

* The dataset needs to be balanced with rare classifications having sufficiently high number of instances. This is required to prevent the algorithm from predicting only the common classifications.
* One method to perform balancing is to reduce the proportion of records with more common classifications.

**Stretching Axis for Attribute Importance :**

* Stretching the Axes finds the coefficient zj by which to multiply the jth axis. Larger values of zj are associated with the more important variable axes



**KNN for Estimation and Prediction** :

* Can be done using **Locally Weighted Averaging**

Table

Description automatically generated

Text, whiteboard

Description automatically generated



**Choosing the value of K** :

* Smaller K = **HIGH VARIANCE**
  + May lead to overfitting
  + Noise or outliers may drastically affect the classification
* Larger K = **HIGH BIAS**
  + Will tend to smooth out obscure data values (outliers) in the training data
  + May overlook locally interesting neighbours

**LINEAR REGRESSION**

**Simple Linear Regression**

* It predicts a quantitative response Y on the basis of single predictor X.
* It assumes a linear relationship between X and Y



* β0 🡪 Intercept and β1 🡪 Slope
* Once the training data is used to estimate the slope and intercept coefficients, predictions can be made using :



**Estimating Coefficients**

* β0 and β1 are unknown. Before we use f(X) = β0 + β1.X , we must estimate the coefficients β’0 and β’1. This is done by Minimizing the Least Squares.
* Let y’i = β’0 + β’1.xi be the prediction for Y for the ith value of X.
  + Then, ith residual ei  = ( yi – y’i )
  + We then compute the Residual Sum of Squares (RSS)



* + When RSS is minimized, the coefficients are estimated as :

Text, letter

Description automatically generated

* If we calculated RSS at every possible value for β0 and β1, it has a bowl-shape with a unique minimum.
  + The red dot represents the best estimated value for these coefficients where RSS gets minimized. This is called the Ordinary Least Squares (OLS) method.

Chart, diagram

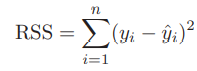
Description automatically generated

**Assessing Accuracy of Coefficient Estimates**

* For linear regression, the population regression line is given by
  + As the coefficients are unknown, we estimate β’0 and β’1 to define the Least Squares Line.
  + This line will be the best estimate of the population regression line but will never be an exact match.
* Graphical user interface, application

  Description automatically generatedStandard Error tells us the average amount that an estimate differs from its actual value. For Linear Regression coefficients, standard errors are given by :
  + Where σ2 = Var(e)
    - This value of σ2 is not known but can be estimated from the data. This estimation is called Residual Standard Error (RSE).
* A picture containing graphical user interface

  Description automatically generatedResidual Standard Error (RSE) measures the standard deviation of the residuals in a regression model.
* RSE is measured in terms and units of Y. Hence, it is not always clear what constitutes a good RSE. Here, R2 – Statistic provides an alternative measure of fit.
  + It is the proportion of variance of the response explained by the model.
  + It ranges between 0 and 1, and is independent of the scale of Y



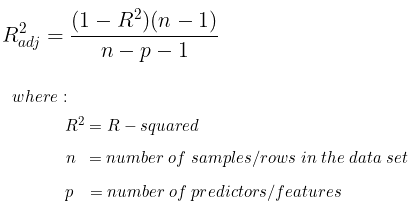
**Multiple Linear Regression**



* The parameters are estimated using the same least squares approach that was discussed in Simple Linear Regression.
  + We estimate coefficients β0 , β1 … βp so as to minimise the Residual Sum of Squares (RSS) which is given by:

Text, letter

Description automatically generated

* R2 – Statistic denotes the proportion of variance of response that can be explained by the model. The problem with it in case of multiple regression is that it will always increase when more variables are added to it, even if those variables are only weakly associated with the response
  + This happens because adding another variable to the least squares line allows for a better fit.
  + As a result, Adjusted – R2 is used. It penalizes the score as more predictors are added.

**Interaction Effects**

* One way of extending this model is to add an interaction effect. This third term added is called the Interaction Term.

A picture containing chart

Description automatically generated

* Sometimes, it is the case that an interaction term has a very small p-value, but the associated main terms do not.
* Hierarchy Principle 🡪 if we include an interaction term in the model, we should also include the main effects even if the p-values associated with their coefficients are not significant,
  + Interactions are hard to interpret in a model without main effects and their meaning gets changed.
* Consider a model with predictors Xi and Di
  + Let Di indicate whether the subject is a man (Di = 1) or a woman (Di = 0)
  + Let the model also have an interaction term between Xi and Di
* For women,

Box and whisker chart

Description automatically generated with medium confidence

* For men,

A picture containing text, clock, watch, orange

Description automatically generated

* A picture containing text, sky, day

  Description automatically generatedPlotting these two lines, we get

**Problems with Linear Regression**

1. **Non-Linearity of Data**
   * Linear Regression assumes that the relationship between X and Y is linear
   * Residual Plots help identify non-linearity
     + For simple linear regression, we plot residuals (yi – y’i) versus the predictor xi
     + For multiple linear regression, we plot residuals (yi – y’i) versus the fitted values y’i
   * Ideally, there should be no discernible pattern
   * Chart, scatter chart

     Description automatically generatedIf the residual plot indicates that there is some non-linear association, transformation of predictor variables can be done to minimise the non-linearity
2. **Non-Constant Variance of Error Terms**
   * An important assumption of Linear Regression is that error terms have a constant variance i.e. Variance(e) = σ2
   * This directly affects confidence intervals, standard errors, and hypothesis testing
   * Non-Constant error terms lead to Heteroscedasticity.
     + Present in the form of a funnel-shape in residual plots
   * One solution is to transform the response using a concave function like Log or sqrt
   * Another solution is to fit the model using weighted least squares where weights are proportional to the inverse of variances.

Chart, scatter chart

Description automatically generated

1. **Correlation of Error Terms**
   * Another important assumption of Linear Regression is that the error terms e1, e2 …. en are un-correlated
   * The Standard Errors computed for estimation of fitted values are based on this assumption
     + If there is correlation, the estimated standard errors would underestimate the true standard errors
     + Confidence and prediction intervals will be narrower
     + P-values associated with the model will be lower than they should be
       - Lead to an unwanted sense of confidence
   * Frequently observed in time-series data
2. **High Leverage Points**
   * Outliers are observations for which yi is unusual for a given xi
   * High Leverage Points have unusual xi
     + They inflate the strength of the regression relationship by reducing the p-value to increase the chance of a significant relationship
     + They have no impact on the coefficients because the point follows the predicted regression line
3. **Collinearity**
   * Collinearity refers to the situation in which two or more predictor variables are closely related to each other
   * Reduces the accuracy of the model and power of hypothesis testing
   * One way to diagnose is to look at the Correlation Matrix
   * Multi-Collinearity occurs when three or more variables are correlated without having a high correlation in pairs.
     + Cannot be detected using a correlation matrix
     + Variance Inflation Factor (VIF) is used
       - VIF = 1 🡪 Absence of multi-collinearity

VIF > 5 or 10 🡪 Problematic

**MODEL SELECTION – Reduce Variance**

**Model Selection Methods**

1. **Subset Selection**
   * Identify a subset of p predictors that we believe are related to the response
   * Fit with least squares on this reduced set
2. **Shrinkage Methods**
   * Fit a model with all p predictors, but with a method that shrinks some coefficients toward 0 (relative to OLS).
   * This shrinkage (or regularization) reduces model variance
   * Some shrinkage methods set some coefficients to exactly 0 (thus perform variable selection)
3. **Dimension Reduction**
   * Project the p predictors into a M-dimensional subspace, where M<p.
   * Achieved by computing M different linear combinations, or projections, of the predictors. Then fit on these M projections using OLS.

**Subset Selection**

* **Best Subset Selection**
  + Text, letter

    Description automatically generatedFit a separate OLS regression for each combination of p predictors and pick the best combination of predictors
  + In step 3 , we cannot use R2 or RSS because as k increases, RSS will decrease while R2 will increase due to addition of predictors.
  + For classification models, we use Deviance to rank model performance.
    - Lower the better
  + Drawbacks :
    - computationally burdensome (2p models for p predictors)
    - cannot apply when p is very large
    - when p is large, the models might perform well on the training data (due to overfitting) but poorly on test data
* **Forward Stepwise Selection**
  + Start with no predictors, add the predictor with the biggest additional improvement in fit

Text, letter

Description automatically generated

* + Advantages :
    - Computationally easier
    - Can be applied when p>n
  + Drawback :
    - Because the algorithm adds one at a time, if it “makes a mistake” early, it cannot correct 🡪 not guaranteed to find the best model
* **Backward Stepwise Selection**
  + it begins with the full least squares model containing all p predictors, and then iteratively removes the least useful predictor, one-at-a-time.

Text, letter

Description automatically generated

* + Advantages :
    - Computationally easier
    - Might arrive at a different model than Forward Stepwise Selection
  + Drawbacks :
    - Not guaranteed to find the best model
    - Cannot apply when p>n

**Selecting the Optimal Model**

* **Mallow’s Cp**
  + It is a way to assess the fit of a multiple regression model.
  + The technique then compares the full model with a smaller model with “p” parameters and determines how much error is left unexplained by the partial model.
  + Diagram

    Description automatically generatedPenalizes as number of predictors increase
* **Akaike Information Criterion (AIC)**
  + It is a measure of fit that can be used to assess models.
  + This measure uses the log-likelihood, but also adds a penalizing term associated with the number of variables.
  + Lower the Better

Diagram, text

Description automatically generated

* **Bayesian Information Criterion (BIC)**
  + The BIC resolves this problem by introducing a penalty term for the number of parameters in the model. The penalty term is larger in BIC than in AIC.
  + Text

    Description automatically generatedLower the Better

**Shrinkage Methods**

* Instead of fitting a model with a just a subset of p predictors, we can use all p predictors but constrain the coefficient estimates by shrinking them toward zero
* This biases the coefficients but reduces the variance
* **Ridge Regression**
  + Linear Regression estimates the coefficients by minimizing the RSS
  + Ridge Regression learns weights and biases using the same logic as OLS but adds a penalty for large variations in w parameters using a penalty term (Shrinkage Penalty).
  + A picture containing diagram

    Description automatically generatedThe addition of a parameter penalty is called **Regularization**. It prevents overfitting by restricting the model. Ridge Regression uses **L2 Regularization**.
  + The influence of regularization term is controlled by the tuning parameter λ. Higher value of tuning parameter leads to more regularization and hence simpler models.
  + This shrinkage penalty is only applied to the coefficients but not to the intercept.
  + Due to the presence of the penalty term, Ridge Regression is not scale-invariant. Hence, if we multiply the predictors by c, the coefficients will not scale (in OLS the coefficients would have been scaled by 1/c)
  + Drawbacks :
    - Includes all predictors as they shrink towards zero but do not become zero
    - At first when variance is reduced, it performs better than OLS. But after a certain point, Bias starts to increase a lot and the performance becomes poor.
* **Lasso Regression**
* Lasso Regression over comes the drawback of Ridge Regression where in coefficients shrink to zero but never become equal to zero.
* A picture containing text, clock

  Description automatically generatedThe only difference is that the β2 j term in the ridge regression penalty has been replaced by |βj | in the lasso penalty.
* Lasso Regression uses **L1 Regularization** to set the parameter weights in w for the least influential variables.
* As with ridge regression, the lasso shrinks the coefficient estimates towards zero. However, in the case of the lasso, the L1 penalty has the effect of forcing some of the coefficient estimates to be exactly equal to zero when the tuning parameter λ is sufficiently large.
  + - Hence, the lasso performs variable selection.
    - Lasso yields sparse models—that is, sparse models that involve only a subset of the variables.

**Relationship Between Ridge, LASSO and Best Subset**

* Diagram

  Description automatically generated with medium confidenceGraphical user interface, text

  Description automatically generatedRidge regression and LASSO can be seen as computationally feasible versions of best subset (especially LASSO, because it performs variable selection)

**High Dimensional Data**

* Data sets containing more features than observations are often referred to as high-dimensional.
* Classical approaches such as least squares linear regression are not appropriate in this setting.
  + Role of Bias - Variance trade off leading to overfitting
* When the number of features p is as large as, or larger than, the number of observations n, least squares cannot be performed.
  + Regardless of whether or not there truly is a relationship between the features and the response, least squares will yield a set of coefficient estimates that result in a perfect fit to the data, such that the residuals are zero
  + The model R2 increases to 1 as the number of features included in the model increases, and correspondingly the training set MSE decreases to 0 as the number of features increases, even though the features are completely unrelated to the response.
  + The MSE on an independent test set becomes extremely large as the number of features included in the model increases, because including the additional predictors leads to a vast increase in the variance of the coefficient estimates.
  + Unfortunately, the Cp, AIC, and BIC approaches are not appropriate in the high-dimensional setting, because estimating σ2 is problematic.
* Less Flexible models tend to work well when p > n
  + Forward Stepwise Selection
  + Ridge Regression
  + LASSO
  + Ridge
    - Avoid overfitting by imposing constraints on the fit (a less flexible fit) compared to least squares

**NON-LINEARITY – Reduce Bias**

**Polynomial Regression**

* Extends Linear Regression by adding extra predictors, obtained by raising each of the original predictors to a power

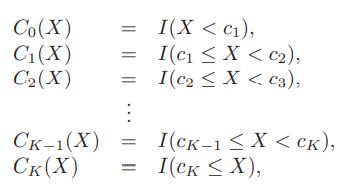


* For a large enough d (degree of polynomial regression), the curve becomes highly non-linear
  + It is unusual to use d greater than 3 or 4 as the polynomial curve can become overly flexible and can take on some very strange shapes
* Text, letter

  Description automatically generatedStandard error at each point can be computed :

**Step Functions**

* Using polynomial functions of the features as predictors in a linear model imposes a global structure on the non-linear function of X.
  + We can instead use step functions in order to avoid imposing such a global structure.
  + Here step function we break the range of X into bins and fit a different constant in each bin.
  + This amounts to converting a continuous variable into an ordered categorical variable.
* We create c1,c2….cK cut points in the range of X. This results in (K+1) dummy variables :



* + For any value of X, C0(X) + C1(X) + …. Ck(X) = 1 as X must be in exactly one of these intervals.
  + We then use Least Squares to fit a linear model using C0(X) , C1(X) , …. Ck(X) as predictors:

**Basis Functions**

* Polynomial Regression and Step Functions ( or Piecewise- constant Regression) models are special cases of a Basis Function Approach.
* Text, letter

  Description automatically generatedThe idea is to have a family of functions or transformations that can be applied to X. So, instead of fitting a linear model, we fit the following model :

**Regression Splines**

* Instead of fitting a polynomial regression over the entire range of data, we now fit a polynomial over different regions of X. This is a more flexible class of Basis Functions.
* These are of two types:
  + Piecewise Polynomial Splines
  + Smoothing Splines

**Piecewise Polynomial Splines**

* **Piecewise Polynomials**
  + Instead of fitting a high-degree polynomial over the entire range of X, we fit separate low-degree polynomials over different regions of X.
  + ****Cubic Polynomial :
  + Piecewise Cubic Polynomial :

A picture containing diagram

Description automatically generated

* + The points where the coefficients change (c) are called Knots. Using more knots leads to a more flexible piecewise polynomial. If we place K knots over the entire range of X, we get K+1 different piecewise polynomials. When the degree is 0, it becomes Stepwise Function
* **Adding Constraints**
  + Without constraints, the piecewise polynomial can be discontinuous at the Knots.
  + We can constrain the polynomial to be continuous over regions of X or to have continuous first and second derivatives at each knot.
    - Graphical user interface, chart

      Description automatically generatedEach constraint removes one degree of freedom by reducing the complexity of the fit.
* **Cubic Splines**
  + Cubic Splines are sufficiently flexible to consistently estimate the smooth regression function f(). We can use higher degree splines but there wont be a significant increase in the performance.
  + To use a cubic Spline, we only need to set the number of Knots.
  + A Cubic Spline with K knots is as complex as a polynomial with (k+3) degree :
  + In total, we get K+4 degrees of freedom :

1. Intercept
2. X
3. X2
4. A picture containing text

   Description automatically generatedX3
5. K knots :

* **Choosing the Number and Location of Knots**
  + One option is to place more knots in places where we feel the function might vary most rapidly, and to place fewer knots where it seems more stable.
  + While this option can work well, in practice it is common to place knots in a uniform fashion.
    - One way to do this is to specify the desired degrees of freedom, and then have the software automatically place the corresponding number of knots at uniform quantiles of the data.
  + Use Cross-Validation to determine the number of Knots
    - Hold out test sample
    - Fit on training sample with K knots
    - Predict on test
    - Repeat multiple times and compute overall MSE
    - Repeat for different K
    - Pick K with lowest MSE
* **Comparison Between Splines and Polynomial Regression**
  + Polynomial Regression improves fit by increasing the degree
    - Places a global structure
  + Splines improve fit by increase the number of knots
    - Place a local structure
* **Degrees of Freedom**
  + DOF capture the complexity of a regression model. For different models, the DOF (excluding the intercept) are :

Graphical user interface

Description automatically generated with medium confidence

**Smoothing Splines**

* Text

  Description automatically generatedIn Smoothing Splines, we want to fit a smooth curve g(x) that fits the data well. This g(x) tries to minimize the RSS as follows :
* If we don’t constraint this g(), then the smooth spline will just interpolate the data points and lead to overfitting.
  + This is taken care of by adding a Penalty Term

Text

Description automatically generated with medium confidence

* + The Smoothing Spline that minimizes the above RSS is essentially a Shrunken version of the Natural Cubic Spline
    - Piecewise Cubic Polynomial with Knots at each xi (a knot at each training observation)
    - Continuous first and second derivatives
    - Fewer DOF due to the presence of the Penalty Term.
* Choosing the Smoothing Parameter λ
  + λ controls the roughness of the spline and determines the Effective DOF
  + As λ increases from 0 to ∞ , the effective DOF dfλ decreases from n to 2
  + DOF refer to the number of free parameters i.e. number of coefficients
    - For a smoothing spline, these n parameters are heavily shrunk down due to λ
    - As a result we measure dfλ
      * Higher the dfλ , more flexible the Smoothing Spline is
  + The value of optimal λ can be computed using Cross-Validation
    - Text

      Description automatically generatedWhen we apply LOOCV, we find that a single equation can be used to find the optimal value:

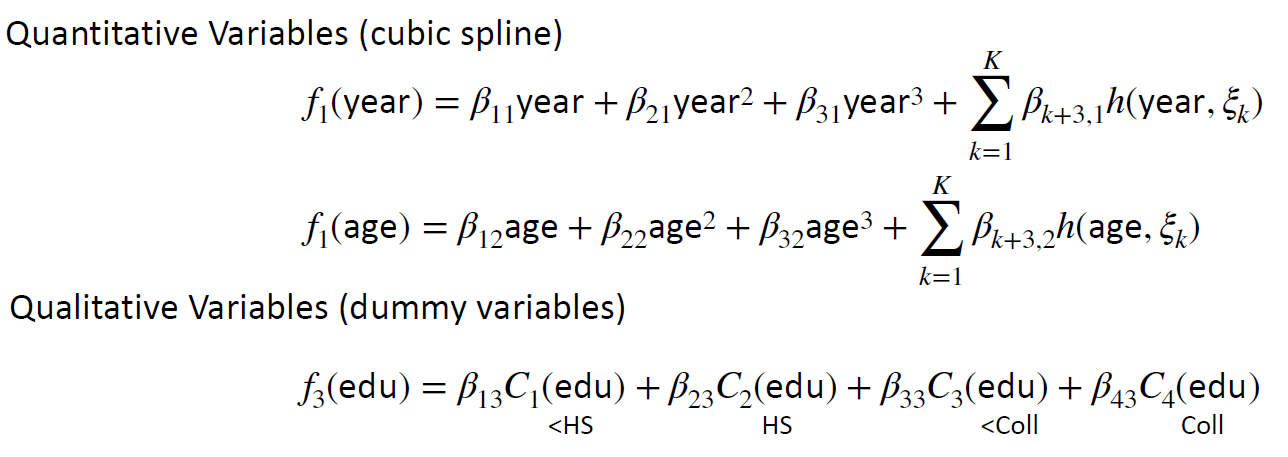
**Generalized Additive Models (GAMs)**

* GAMs provide a general framework for extending a standard linear model by allowing non-linear functions of each of the variables.
* It is an extension of Multiple Linear Regression

A picture containing text, antenna

Description automatically generated

* For example :



* Advantages :
  + Allows us to fit a non-linear fj() to each Xj , so that we can automatically model non-linear relationships that standard linear regression will miss. This means that we do not need to manually try out many different transformations on each variable individually
  + The non-linear fits can potentially make more accurate predictions for the response Y
  + Because the model is additive, we can still examine the effect of each Xj on Y individually while holding all of the other variables fixed. Hence if we are interested in inference, GAMs provide a useful representation.
* Drawback :
  + The main limitation of GAMs is that the model is restricted to be additive. With many variables, important interactions can be missed. However, as with linear regression, we can manually add interaction terms to the GAM model by including additional predictors of the form Xj × Xk
* Diagram

  Description automatically generatedCan be extended to Classification Problems as well :

**LOGISTIC REGRESSION**

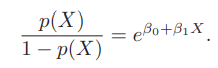
**Simple Logistic Regression**

* Logistic Regression models the probability that response Y belongs to a particular category
* It is a linear method as inputs X are combined linearly using coefficients
* The Logistic Function (also called Sigmoid Function) is given by :

Chart, box and whisker chart

Description automatically generated

* Upon manipulating the Logistic Function, we get



**Odds**

* Taking the log on both sides, we get the Log-Odds or Logit form

A picture containing text

Description automatically generated

**Coefficients in Logistic Regression**

* The coefficients β0 and β1 are unknown and must be estimated based on the available Training Data. This is done using the Maximum-Likelihood Estimation.
* The parameter values are found such that they maximise the likelihood that the process described by the model produced the data that were actually observed
* Maximum Likelihood requires an assumption about the data’s distribution such as normal distribution.
* In case of normal distribution, different values of mean (u) and standard deviation (σ) give different curves.
  + If the curve is wrong, the distribution will be far from the mean (u) of the actual curve.
  + Hence, the probability or likelihood of observing those far-off measurements will be low.
  + As the curve gets closer to the actual distribution, the likelihood increases
  + Once we have figured out the maximum likelihood estimation for the mean, we do the same process for different standard deviations of the curve

Table

Description automatically generated**Confusion Matrix**

Diagram, text, letter

Description automatically generated

A picture containing diagram

Description automatically generated

Text

Description automatically generated

**Logistic Regression Over Linear Regression**

* Linear Regression can produce probabilities lower than 0 and higher than 1
* For categorical variables (nominal but with numeric labels) , Linear Regression would assume an order that may not make sense

**TREE-BASED METHODS**

* Trees-Based Methods operate by dividing up the feature space into rectangles. Each rectangle can be considered as a neighbourhood in KNN algorithm.
* Predictions are based on the average value (regression) or the most common class (classification) in each rectangle.
* Decision Tree is collection of **decision nodes**, connected by **branches**, extending downward from **root node** to terminating **leaf nodes**. Beginning with root node, attributes tested at decision nodes, and each possible outcome results in branch.

Diagram

Description automatically generated

**Requirements for Decision Trees** :

* Decision tree algorithms represent supervised learning, and as such require pre-classified target variables.
  + A training data set must be supplied which provides the algorithm with the values of the target variable
* This training dataset should be rich and varied, providing the algorithm with a healthy cross section of the types of records for which classification may be needed in the future.
  + Decision trees learn by example, and if examples are systematically lacking for a definable subset of records, classification and prediction for this subset will be problematic or impossible

**Pure Leaf Nodes and Diverse Leaf Nodes :**

* When no further splits can be made, the decision tree algorithm stops growing new nodes.
* Pure Leaf Nodes are nodes where the target variable is unary for the records in that node (e.g., each record in the leaf node is a good credit risk). Then no further splits are necessary, so no further nodes are grown.
* Diverse Leaf Nodes have mixed values for the target attribute.
  + For example, subset of records has Savings= “High”, Income<= $30,000, and Assets= “Low”. Leaf node contains 2 “Good Risk”, and 3 “Bad Risk” records. All records contain same predictor values. No way to split further leading to pure leaf node
  + In this case, the decision tree may report that the classification for such customers is “bad,” with 60% confidence, as determined by the three-fifths of customers in this node who are bad credit risks.

**Building a Tree**

* Algorithm :
  + **STEP 1** : Divide the predictor space into J distinct and non-overlapping regions R1,R2…. RJ
  + **STEP 2** : For every observation that calls in RJ, make the prediction as the mean value of that region.
* The goal is to minimize RSS across regions :

Text

Description automatically generated

* Multiple combinations of regions are possible. As a result, we use the Recursive Binary Splitting Approach to construct the Regions.
  + Top Down : Start at the top of the tree
  + Greedy : At each step, take the best split, rather than looking ahead and picking a split that might lead to a better tree.

Chart

Description automatically generated with low confidence

**Recursive Binary Splitting**

Diagram, text, letter

Description automatically generated

**Tree Pruning**

* Recursive Binary Splitting will overfit the data because the resulting tree T0 will be too complex as the tree would have multiple splits.
* A smaller tree with fewer splits might lead to lower variance and better interpretation at the cost of little bias.
* One alternative is to build the tree only so long as the decrease in the RSS due to each new split exceeds some high threshold.
  + This will result in smaller trees
  + The issue would be that sometimes a seemingly worthless split at the start might be followed by a very good split later on
* A better strategy is to grow a very large tree T0 and then **prune** it back in order to obtain a subtree.
  + Pruning refers to eliminating some of the branches
  + The selected subtree should lead to the lowest Test Error Rate
    - Given a subtree, we can run cross-validation to evaluate the overall Test Error Rate
    - This would be computationally infeasible for all possible subtrees
    - Cost Complexity Pruning (or Weakest Link Pruning) allows us to take care of this

**Cost Complexity Pruning**

* A picture containing text, clock

  Description automatically generatedRather than considering every possible subtree, we consider a sequence of trees indexed by a non-negative tuning parameter α

Text

Description automatically generated

Diagram

Description automatically generated with medium confidence

**Classification Trees**

* A classification tree is very similar to a regression tree, except that it is classification used to predict a qualitative response rather than a quantitative one.
* For a classification tree, we predict that each observation belongs to the most commonly occurring class of training observations in the region to which it belongs.
* We can’t use RSS as the criterion to make binary splits. Instead, we use the Classification Error Rate.
* A picture containing diagram

  Description automatically generatedClassification Error Rate (E) is given by the formula mentioned below. Here, pmk is the proportion of training observations in the mth region belonging to the kth class
* This Classification Error Rate (E) is not sensitive enough for tree growing. Instead, we use two other measures :
* **Gini Index (G)**
  + Measure of total variance across the K classes

Text

Description automatically generated

* + It is sensitive enough as it takes a small value when pmk is close to either 0 or 1 (extreme points for pmk). Represents the node purity that indicates that a node contains predominantly observations from a single class
    - The minimum value of the Gini Index is 0. This happens when the node is pure, this means that all the contained elements in the node are of one unique class. Therefore, this node will not be split again.
    - Thus, the optimum split is chosen by the features with less Gini Index.
    - Chart, diagram, box and whisker chart

      Description automatically generatedMoreover, it gets the maximum value when the probability of the two classes is the same.
* **Cross Entropy (D)**
  + A picture containing text, clock

    Description automatically generatedEntropy is a measure of information that indicates the disorder of the features with the target.
  + A picture containing chart

    Description automatically generatedSimilar to the Gini Index, the optimum split is chosen by the feature with less entropy. It gets its maximum value when the probability of the two classes is the same and a node is pure when the entropy has its minimum value, which is 0

**Gini Index vs Entropy**

* Gini Index has values in the range [0 , 0.5]
* Entropy has values in the range [0 , 1]
* Computationally, entropy is more complex since it makes use of logarithms and consequently, the calculation of the Gini Index will be faster.

Chart

Description automatically generated

**Trees vs Linear Models**

* If the relationship between the features and the response is well approximated by a linear model, then an approach such as linear regression will likely work well and will outperform a method such as a regression tree that does not exploit this linear structure.
* If instead there is a highly non-linear and complex relationship between the features and the response as indicated by model (8.9), then decision trees may outperform classical approaches.

Chart

Description automatically generated

**Advantages of Tree-Based Methods**

* Trees are easily interpretable as they closely mirror the human decision-making process.
* They are easy to display graphically
* They can handle qualitative (ordinal and nominal) variables easily without creating dummy variables
* They handle missing values well without dropping observations
  + When a value is missing, they split on a surrogate variable: E.g., if a user's years of job experience is missing, they'll split on an optimally chosen correlated variable like age

**Disadvantages of Tree-Based Methods**

* Trees generally do not have the same level of predictive accuracy as some of the other regression and classification approaches
  + Can be improved by Bagging, Forests and Boosting of trees
* Decision trees tend to have high variance. A small change in the training data can produce big changes in the estimated Tree.

**Bagging**

* Decision Trees are highly variable as small changes in the data can lead to very different trees
* One way to reduce this is to average over independent samples.
  + The issue is that we only have 1 training dataset
  + As a result, we use Bagging which utilizes Bootstrapping to overcome this issue
* The idea is to create several subsets of data from training sample chosen randomly with replacement.
  + Now, each collection of subset data is used to train their decision trees. As a result, we end up with an ensemble of different models.
  + Average of all the predictions from different trees are used which is more robust than a single decision tree.
* Similarly for Classification trees, we take a majority vote of predicted class over all bootstrap samples
* Assume we generate B different bootstrapped training datasets
  + We then fit the tree on the bth bootstrapped dataset to get f’\*b(x)
  + Text

    Description automatically generatedAveraging all the predictions for all such bth trees, we get :

**Out-of-Bag Error Estimation**

* Instead of using Cross-Validation to estimate Test Error of the Bagged tree, we can use a straightforward approach
* In bagging, the trees are repeatedly fit to bootstrapped subsets of the observations.
  + On average, each bagged tree makes use of 2/3 of the observations. The remaining 1/3 are not used to fit that specific tree. These observations are called Out-of-Bag (OOB) Observations.

Text, letter

Description automatically generated

**Interpreting Bagged Trees**

* When we bag over a large number of trees, it is no longer possible to represent the result using a single tree, so we lose the feature of interpretability but gain predictive power.
* However, it is still possible to get a measure of overall variable importance
  + Calculate the total amount that RSS is decreased due to splits on a given predictor, averaged over B trees
  + Can be done for Gini index in case of classification trees

**Random Forests**

* It takes one extra step over Bagging where in addition to taking the random subset of data, it also takes the m random selection of features rather than using all features to grow trees.
  + When you have many random trees. It’s called Random Forest
  + Shape

    Description automatically generatedTypically we choose
* This process de-correlates the bagged trees
  + In building a random forest, at each split in the tree, the algorithm is not even allowed to consider a majority of the available predictors.
  + Suppose that there is one very strong predictor in the data set, along with a number of other moderately strong predictors. Then in the collection of bagged trees, most or all of the trees will use this strong predictor in the top split. Consequently, all of the bagged trees will look quite similar to each other.
  + Random forests overcome this problem by forcing each split to consider only a subset of the predictors.

**SUPPORT VECTOR MACHINES**

**Hyperplanes**

* In a p-dimension space, a hyperplane is a flat affine subspace of dimension (p-1)
  + For a 2-dimension space, the hyperplane is a line
  + For a 3-dimension space, the hyperplane is a plane
* In 2-D, the hyperplane is defined by the equation :
* In a p-dimensions, the hyperplane is defined by the equation :

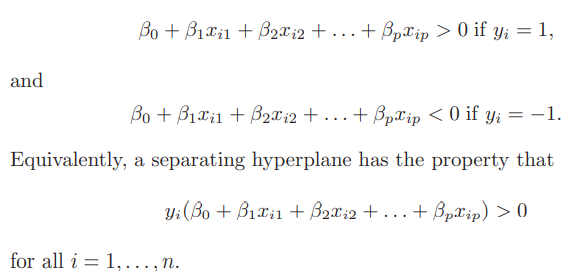
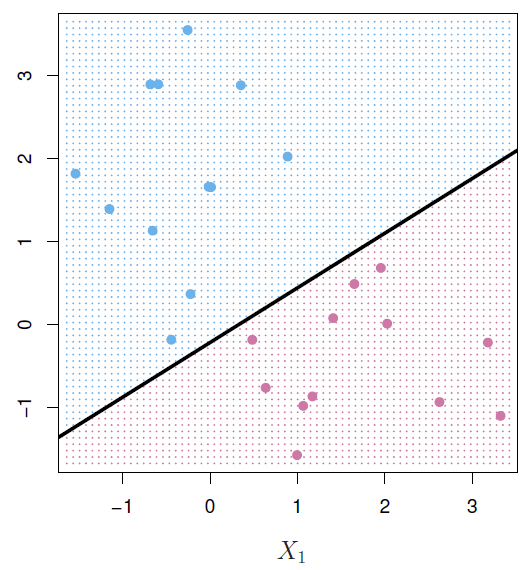
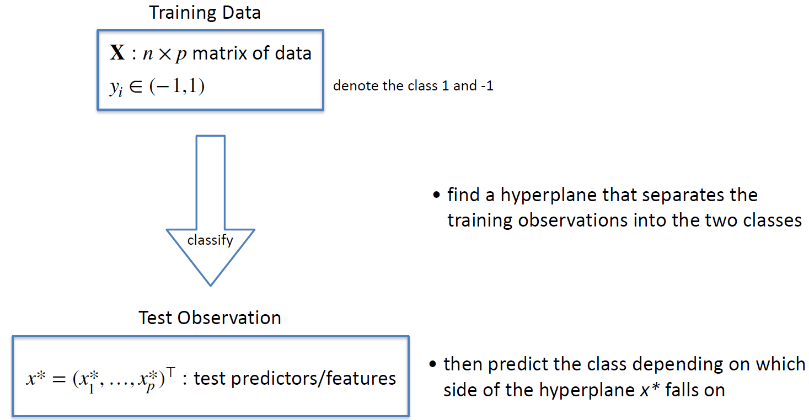


* This hyperplane would divide the p-dimensional space into two halves :

Text, letter

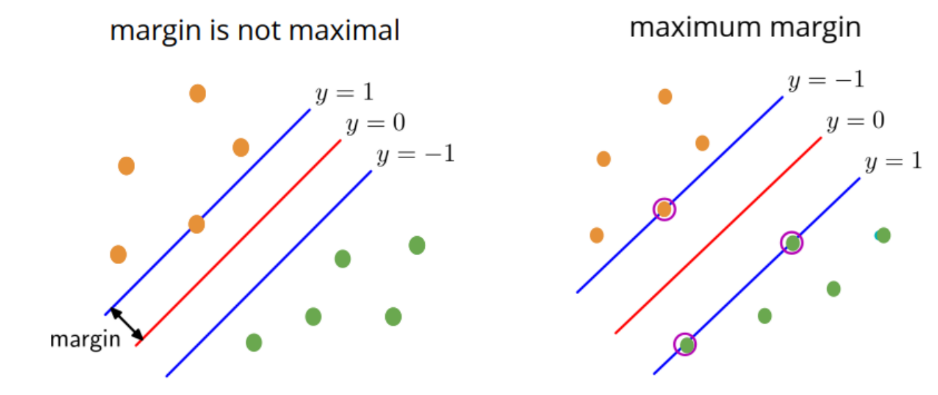
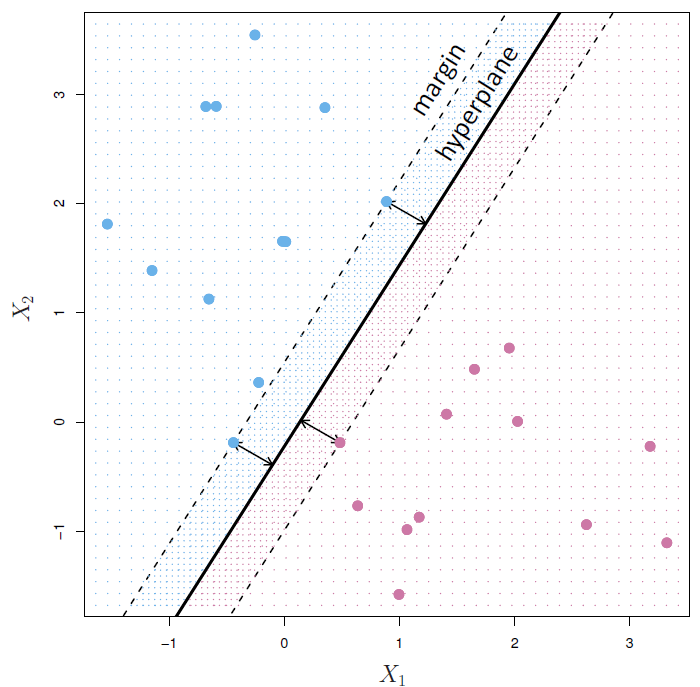
Description automatically generated

**Classification with Hyperplanes**



**Maximal Margin Classifier**

* In general, if our data can be perfectly separated using a hyperplane, then there will exist an infinite number of such hyperplanes.
  + This is because a given separating hyperplane can usually be shifted a tiny bit up or down, or rotated, without coming into contact with any of the observations.
* In order to construct a classifier based upon a separating hyperplane, we must decide which of the infinite possible separating hyperplanes to use.
* A natural choice is the Maximal Margin Hyperplane which is the separating hyperplane that is farthest from the training observations. This is done using **Margin**.
* **Margin** is the perpendicular distance from a hyperplane to the nearest observation.
* **Maximal Margin Hyperplane** is the hyperplane with the largest margin
  + We can then classify a test observation based on which side of the maximal margin hyperplane it lies. This is known as the maximal margin classifier.
  + Although the maximal margin classifier is often successful, it can also lead to overfitting when p is large.



* **Support Vectors** are the observations that lie on the margin. These help identify the margin in the first place.
  + If these points are moved, the classifier is changed
  + Chart, scatter chart

    Description automatically generatedIf other points are moved, the classifier is unchanged.

**Construction of the Maximal Margin Classifier**

* Consider n training observations x1, x2 …. Xn
* Consider their associated labels y1, y2 … yn with values in {-1,+1}
* Graphical user interface, text, application

  Description automatically generatedThen, the Maximal Margin Classifier is simply the solution to the optimization problem :

**Non-Separable Observations**

* The maximal margin classifier is a very natural way to perform classification, if a separating hyperplane exists.
* However, in many cases no separating hyperplane exists, and so there is no maximal margin classifier.
  + In this case, the optimization problem has no solution with M > 0.
* We can extend the concept of a separating hyperplane in order to develop a hyperplane that almost separates the classes, using a so-called **Soft Margin**.
* The generalization of the maximal margin classifier to the non-separable case is known as the **Support Vector Classifier**.

Chart, scatter chart

Description automatically generated

**Support Vector Classifiers**

* Even if a separating hyperplane does exist, then there are instances in which a classifier based on a separating hyperplane might not be desirable.
* A classifier based on a separating hyperplane will necessarily perfectly classify all of the training observations.
  + This can lead to sensitivity to individual observations. The addition of a single observation may lead to a dramatic change in the maximal margin hyperplane.
    - The resulting maximal margin hyperplane may not satisfactory—for one thing, it may only a tiny margin.
    - This is problematic because the distance of an observation from the hyperplane can be seen as a measure of our confidence that the observation was correctly classified.
  + Moreover, the fact that the maximal margin hyperplane is extremely sensitive to a change in a single observation suggests that it may have overfit the training data.
* In this case, we might be willing to consider a classifier based on a hyperplane that does not perfectly separate the two classes, in the interest of :
  + **Greater robustness to individual observations**
  + **Better classification for most of the observations**
* The **Support Vector Classifier** (or Soft Margin Classifier) allows some observations to be on the incorrect side of the margin or even on the incorrect side of the hyperplane.
  + The margin is considered to be Soft because it is being violated by some of the training observations.

**Construction of Support Vector Classifiers**

* Graphical user interface, text, application

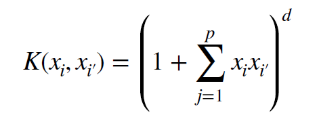
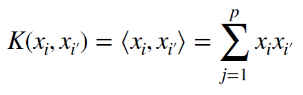
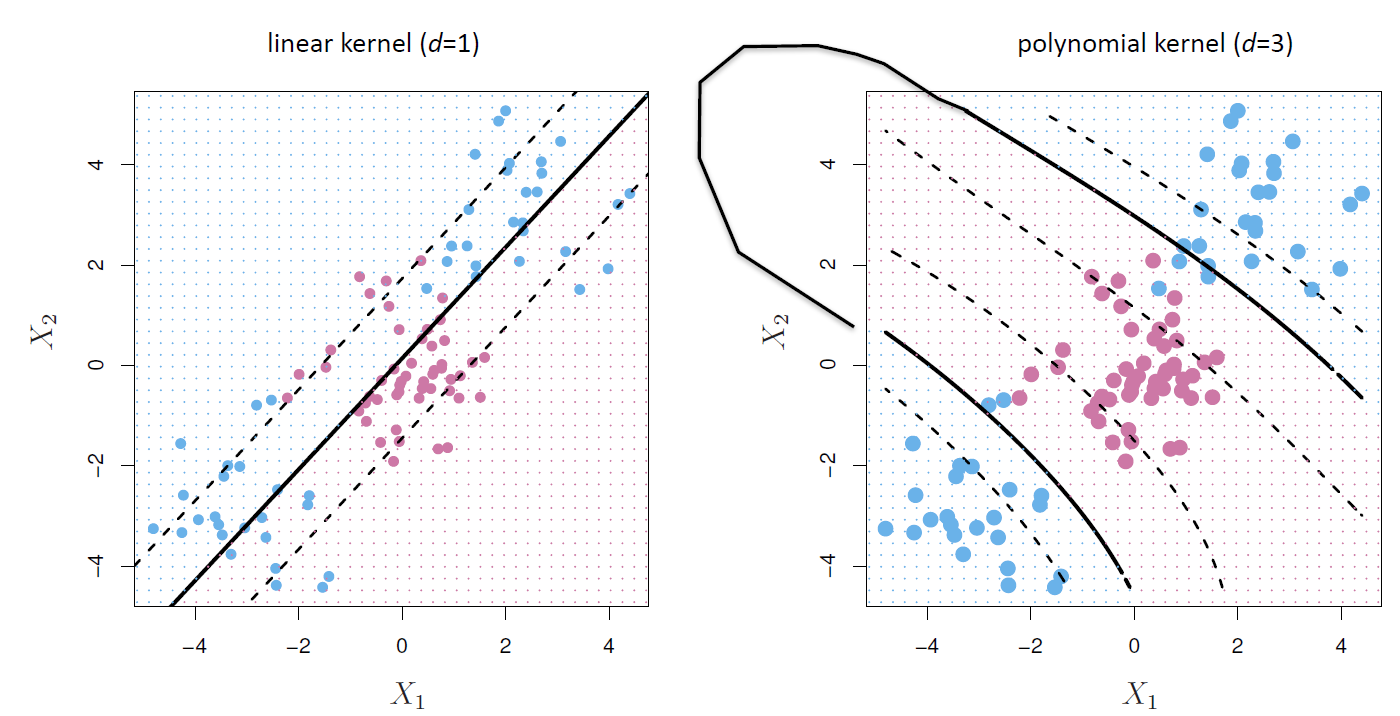
  Description automatically generatedThe Support Vector Classifier is simply the solution to the updated optimization problem :
* The **slack variables** are used to allow observations on the wrong side of the hyperplane.
  + If **ei = 0** , the ith observation is on the correct side of the hyperplane
  + If **ei > 0** , the ith observation is on the wrong side of the margin 🡪 margin violation
  + If **ei > 1** , the ith observation is on the wrong side of the hyperplane
* **Tuning Parameter C** bounds the sums of slack variables. In essence, it determines the number and severity of the violations to the margin that we want to tolerate.
  + As C increase, we allow more observations to be on the wrong side of the hyperplane
  + Controls the Bias-Variance Trade-Off
    - Small C 🡪 Classifier that is highly fit to the data 🡪 High Variance
    - Large C 🡪 Less data being fit closely 🡪 High Bias
  + Generally chosen via Cross Validation

**Support Vector Machines**

* **Support Vector Machines (SVM)** are an extension of the support vector support vector machine classifier that results from enlarging the feature space in a specific way, using kernels.
* The main idea is that we may want to enlarge our feature space in order to accommodate a non-linear boundary between the classes.
* Text

  Description automatically generatedThe solution to the Support Vector Classifier problem depends only on the inner product (dot product) of the observations. Hence, the Linear Support Vector Classifier can then be represented as :
* Text

  Description automatically generatedIn SVM, replace the inner product with a generalization **(Kernel)**
* A kernel is a function that quantifies the similarity between two observations.



**CLUSTERING**

* Clustering is the grouping of records, observations, or cases into classes of similar objects
* A cluster is a collection of records that are similar to one another and dissimilar to records in other clusters
* The task does not try to classify, estimate, or predict the value of a target variable
  + Unsupervised Learning – No target variable
* Seeks to segment the entire data set into homogeneous subgroups
  + Elements in cluster should be as similar as possible
* The similarity of records within the cluster is maximized and the similarity to records outside the cluster is minimized

**Goal of Clustering :**

To construct clusters of records such that the between-cluster variation is large compared to the within-cluster variation

Diagram

Description automatically generated

**Distance Measurement :**

Text, letter

Description automatically generated

**Hierarchical Clustering :**

* Results in a Dendrogram
  + Tree like structure
  + Created through recursively dividing or combining records
* Agglomerative Methods
  + Initialize each observation to be a tiny cluster of its own
  + Combine existing clusters to create tree
* Divisive Methods
  + Begin with all records in one big cluster
  + Split off most dissimilar records into separate clusters

**Determining distance between Clusters :**

* Single Linkage
  + Based on Nearest Neighbour
    - Searches for minimum distance between records
  + Forms long slender clusters
* Complete Linkage
  + Based on Farthest Neighbour
    - Searches for maximum distance between records
  + Forms compact sphere-like clusters
* Average Linkage
  + Based on the average distance between all records in one cluster to all records in another cluster
  + Reduces dependence on extreme values
  + Used in k-Means

**NEURAL NETWORKS**

* Other names for Feedforward Neural Networks:
  + Multi-Layer Perceptrons
  + Deep Forward Networks
  + Dense Neural Networks
  + Fully Connected Neural Networks
* A **Perceptron** is the smallest unit of a Neural Network. It is a simple representation of a biological neuron in an artificial neural network, but it does not represent the way human mind works and functions. It essentially maps the inputs through a linear function and then through a non-linear activation function to an output.

Diagram

Description automatically generated

* **Inputs (xi)** : Input features or outputs from previous layers
* **Weights (wj)** : Learnable real-valued parameters, can be considered as **slope** from LR
* Chart

  Description automatically generated**Bias (b)** : constant input which can be considered as the **intercept** from LR
* **Input Layer** : It is a layer made up of input features and number of perceptrons in this layer is equal to total number of inputs to the model
* **Hidden Layer** : It represents the perceptrons in any non-input / non-output layer
* **Output Layer** : It represents perceptrons used to generate the final output
* **Depth** : Number of layers except the input layer
* **Width of a layer** : Number of perceptrons in a layer

Table

Description automatically generatedDiagram

Description automatically generated

Table

Description automatically generatedDiagram

Description automatically generated

Diagram

Description automatically generated

Graphical user interface, table

Description automatically generated

**Activation Function**

* It is a **non-linear function** that is used to map the input of a perceptron to its output.
* It is important that this activation function is non-linear because if it is linear, then the entire string of linear operations can be compressed into a single operation. This wouldn’t give any extra explanatory power. Hence, **this non-linearity gives the Neural Network good learning power**.
* Chart, line chart

  Description automatically generatedTheoretically, we can apply a different activation function to each hidden layer perceptron. Hence, the more perceptrons we have in the hidden layers, the more activation functions we can apply to the neural network.
* For Linear, Positive Real-Valued Outputs, we use the following activation functions :

A picture containing graphical user interface

Description automatically generated

* For Binary, Categorical Outputs, we use the following activation functions :

Chart

Description automatically generated

**Optimization**

* Optimization is the mathematical problem of finding a set of values that maximizes or minimizes a particular function.
* This function that is being maximized/minimized is called the **Objective Function**. It is also called the Loss Function / Cost Function.
* **Optimization problems are hard because** during optimization, we aim to find the global minima. This would correspond to the overall minimum value. In most cases, it is not possible to find the global minima and hence we settle for a local minima as there would infinite spaces to choose from.
* Diagram

  Description automatically generatedIt becomes easy to find the global minima only in the case of **purely convex functions**.
* Few ways to find these optimal values are :
  + Simplex
  + Newton’s Method
  + Genetic Algorithms
  + Stochastic Gradient Descent

**Gradient Descent**

* The intuition behind gradient descent is that take a step in the direction of the negative gradient (negative derivative in case of 1-D).
* The step size needs to be reduced with every step otherwise the step can overshoot and miss the minima.

Graphical user interface, text, application

Description automatically generated

Diagram, venn diagram

Description automatically generated

**Stochastic Gradient Descent**

* Most problems in statistical learning can be written as minimizing an objective/loss function. It can be calculated as the average of loss at each data point:

Graphical user interface, text, application

Description automatically generated

* Text

  Description automatically generatedIn case of gradient descent, we update the estimate of θ using all data points at each iteration. It can be written in the following form :
* In case of Stochastic Gradient Descent, we update the estimate of θ using a mini-batch of data points at each iteration. It can be written in the following form :

Text

Description automatically generated

* Using mini batches than using all the data points at one time is beneficial because this reduces computational and time complexity. It may also help with the model training in terms of robustness based on the type of problem and data. On massive datasets, Stochastic Gradient descent can converge faster as the updation of θ is more frequent.
* With Stochastic Gradient Descent, there is always a trade-off:
  + Faster updates but error function is not as well minimized as in the case of Gradient Descent (mini-batch = 1)
  + Less frequent updates but a more accurate gradient (mini-batch = size of dataset)
* A smaller mini-batch size can have certain effects such as regularization in certain problem domains.
* Small mini-batch size – Loss is noisy due to greater influence of individual data points
* Larger mini-batch size – Smoother Convergence
* Key hyperparameters of Stochastic Gradient Descent:

Graphical user interface, text

Description automatically generated

**RESAMPLING METHODS**

**Introduction**

* Test Error Rate is the average error that results from using a method to predict a new observation
* The test error can be easily calculated if a designated test set is available. Unfortunately, this is usually not the case.
* Cross-Validation is used to provide measures of test error by splitting the sampled data.

**Validation Set Approach**

* It involves randomly dividing the available set of observations into two parts, a Training Set and a Validation Set or hold-out set.
* The model is fit on the Training Set, and the fitted model is used to predict the responses for the observations in the Validation Set.
* The resulting validation set error rate—typically assessed using MSE in the case of a quantitative response—provides an estimate of the test error rate.

Timeline

Description automatically generated with medium confidence

* Drawbacks :
  + The validation estimate of the test error rate can be highly variable, depending on precisely which observations are included in the training set and which observations are included in the validation set
  + In the validation approach, only a subset of the observations—those that are included in the training set rather than in the validation set—are used to fit the model. Since statistical methods tend to perform worse when trained on fewer observations, this suggests that the validation set error rate may tend to overestimate the test error rate for the model fit on the entire data set.

**Leave-One-Out Cross-Validation (LOOCV)**

* It is closely related to the validation set approach, but it attempts to address the method’s drawbacks
* Like the validation set approach, LOOCV involves splitting the set of observations into two parts. However, instead of creating two subsets of comparable size, a single observation is used for the validation set, and the remaining observations make up the training set.
* The statistical learning method is fit on the (n – 1) training observations, and a prediction y’1 is made for the excluded observation, using its value x’1. Since (x1, y1) was not used in the fitting process, MSE1 = (y’1 – y1)2.
  + We can repeat this approach n times to produce n squared errors MSE1, MSE2 … MSEn.
  + The LOOCV estimate for test MSE is the average of these n MSEs

A picture containing text, watch

Description automatically generated

* Advantages :
  + Less Bias 🡪 does not overestimate the test error as much as validation set approach
  + No randomness in the training/validation splits
* Drawback :
  + It has the potential to be expensive to implement, since the model has to be fit n times. This can be very time consuming if n is large, and if each individual model is slow to fit.

A picture containing chart

Description automatically generated

**k-Fold Cross-Validation**

* This approach involves randomly k-fold CV dividing the set of observations into k groups, or folds, of approximately equal size.
* The first fold is treated as a validation set, and the model is fit on the remaining (k – 1) folds.
* The mean squared error, MSE1, is then computed on the observations in the held-out fold.
* This procedure is repeated k times; each time, a different group of observations is treated as a validation set.
  + This results in k estimates of test error, MSE1, MSE2 …. MSEk
  + A picture containing text, clock, watch

    Description automatically generatedThe k-Fold CV estimate is then computed by averaging these values :
* LOOCV is a special case of k-fold CV in which k is set to equal n.
* In practice, one typically performs k-fold CV using k = 5 or k = 10.

Chart

Description automatically generated

**Bias-Variance Trade-Off for Cross-Validation**

* LOOCV 🡪 Lowest Bias
* Validation Set Approach 🡪 Highest Bias
* For k-Fold CV , one performs it using k = 5 or k = 10, as these values have been shown empirically to yield test error rate estimates that suffer neither from excessively high bias nor from very high variance.

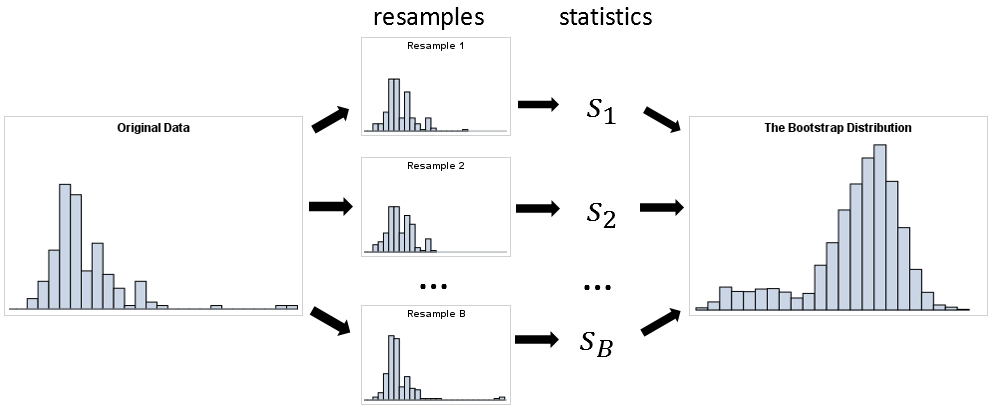
**Cross-Validation for Classification Problems**

* In this setting, cross-validation works the same as before, except that rather than using MSE to quantify test error, we instead use the number of misclassified observations.
* LOOCV error rate is given by :

A picture containing text, watch

Description automatically generated

**Bootstrapping**

* The bootstrap is used to quantify the uncertainty associated with a given estimator or statistical learning method.
* In linear regression, we derived a formula for the standard error of regression coefficients But in many other cases, such formulas are not known. Instead, we can use the bootstrap
* Rather than repeatedly obtaining independent data sets from the population, we instead obtain distinct data sets by repeatedly sampling observations from the original data set.
* We randomly select n observations from the data set in order to produce a bootstrap data set. The sampling is performed with replacement, which means that the replacement same observation can occur more than once in the bootstrap data set.