$RSS = SSE = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n}\left[y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 x_i\right)\right]^2 = S_{yy} - \hat{\beta}_1 S_{xy}$. Residual sum of squares

$\hat{\beta}_1 = \underset{\beta_1 \in \mathbb{R}}{\operatorname{argmin}} RSS = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sum_{i=1}^{n}(x_i-\bar{x})^2} = \frac{S_{xy}}{S_{xx}}$; $\hat{\beta}_0 = \underset{\beta_0 \in \mathbb{R}}{\operatorname{argmin}} RSS = \bar{y} - \hat{\beta}_1 \bar{x}$.

$\{\varepsilon_i\}_{i\leq n} \sim iid\ N(0,\sigma^2) \rightarrow Var(\varepsilon) = \sigma^2 \rightarrow SE\left(\hat{\beta}_0\right)^2 = \sigma^2\left[\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2}\right]$; $SE\left(\hat{\beta}_1\right)^2 = \frac{\sigma^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2}$.

$\hat{\sigma} = RSE = \sqrt{\frac{RSS}{n-2}} = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}$; $p < n+1 \rightarrow RSE = \sqrt{\frac{1}{n-p-1}RSS}$. Residual square error.

$\hat{\sigma}^2 = \frac{RSS}{n-2} = \frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{n-2} = MSE$.

$CI\left(\hat{\beta}_1\right)$: $\hat{\beta}_1 \pm t_{\frac{\alpha}{2},n-2} * SE\left(\hat{\beta}_1\right) \rightarrow \hat{\beta}_1 \pm t_{\frac{\alpha}{2},n-2}\frac{\hat{\sigma}}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}} \rightarrow \hat{\beta}_1 \pm t_{\frac{\alpha}{2},n-2}\frac{\hat{\sigma}}{\sqrt{S_{xx}}}$.

$CI\left(\hat{\beta}_0\right)$: $\hat{\beta}_0 \pm t_{\frac{\alpha}{2},n-2} * SE\left(\hat{\beta}_0\right) \rightarrow \hat{\beta}_0 \pm t_{\frac{\alpha}{2},n-2}\hat{\sigma}\sqrt{\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2}}$.

$R^2 = \frac{TSS-RSS}{TSS} = 1 - \frac{RSS}{TSS}$; $TSS = \sum_{i=1}^{n}(y_i-\bar{y})^2$. Total sum of squares.

$r = Cor(X,Y) = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}} = \frac{S_{xy}}{(S_{xx}S_{yy})^{1/2}}$. $Q_{i,j} = Cov\left(X_i, X_j\right)$; $Cov(X,Y) = \frac{S_{xy}}{n-1}$.

H_0: $\beta_1 = \cdots = \beta_p = 0$ vs H_1: At least one is β_i non-zero . $F = \frac{(TSS-RSS)/p}{RSS/(n-p-1)}$. Reject H_0 if test statistic F is in rejection region: $F > F_{\alpha,df_1,df_2}$. Subset problem → H_0: $\beta_{p-q+1} = \beta_{p-q+2} = \beta_p = 0$. $F = \frac{(RSS_0-RSS)/q}{RSS/(n-p-1)}$. LASTLY, NOTE $RSS = \sum_{i=1}^{n}\left[y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right]^2$

$h_i = \frac{1}{n} + \frac{(x_i-\bar{x})^2}{\sum_{i'=1}^{n}(x_{i'}-\bar{x})^2}$. $VIF\left(\hat{\beta}_j\right) = \frac{1}{1-R^2_{X_j|X_{-j}}}$.

■β_0 is the intercept term – the expected value of Y when $X = 0$. β_1 is the slope – the average increase in Y associated with a one unit increase in X.
■Normal and uncorrelated implies independence.
■ Potential problems in linear modeling: non-linearity of response and predictor relationship, correlation of error terms, non-constant variance, outliers, high-leverage points, collinearity, multicollinearity.
■The RSE is the average amount that the response will deviate from the true regression line. The RSE is considered a measure of the lack of fit of the model to the data. If the predicted $\hat{y}_i$ is very close to the actual y_i for all i the RSE will be very small, and we can conclude the model fits the data well. But since it is measured in the units of Y, it is not always clear what constitutes a good/small RSE. The R^2 statistic provides an alternative measure of fit not dependent on scale. It takes the form of a proportion—the proportion of variance explained—and so it always takes on a value between 0 and 1 and is independent of the scale of Y . An R^2 statistic that is close to 1 indicates that a large proportion of the variability in the response has been explained by the regression. An R^2 close to 0 may have occurred because the linear model is wrong, or the inherent σ^2 is high, or both. Note that R^2 will always increase when predictors are added to the model, but this is overfitting the training data. The TSS measures the total variance in the response Y and can be thought of as the amount of variation inherent in the response before the regression is done. In contrast, the RSS measures the amount of variability that is left unexplained after performing the regression. Hence, TSS-RSS is the amount of variation explained and it is then standardized into a proportion by dividing by TSS. The R^2 is a measure of the linear relationship between X and Y . Recall that correlation of X and Y is also a measure of the linear relationship between X and Y . In simple linear regression $r^2 = R^2$. However, in multiple linear regression setting this relationship does not hold as r only quantifies the linear relationship between two random variables.
■We can look at the individual p-values to decide relevant variables, but if the number of predictors p is large, there are some low p-values by accident. Therefore, ideally, we can test all 2^p models and determine which has the best BIC or AIC . But for large p this is infeasible. Therefore, unless p is small we need an automated efficient algorithm to determine which predictors to include in the final model. The following three approaches are popular:
Forward selection: (Can always be used ; greedy approach that might include variables early that quickly become redundant) We begin with the null model—a model that contains an intercept but no predictors. We then fit p simple linear regressions and add to the null model the variable that results in the lowest RSS. We then add to that model the variable that results in the lowest RSS for the new two-variable model. This approach is continued until some stopping rule is satisfied. **Backward selection**: (Cannot be used for $p > n$) We start with all variables in the model and remove the variable with the largest p-value (least significant). The new $(p-1)$ variable model is fit, and the variable with the largest p-value is removed. This procedure continues until a stopping rule is reached, for instance, when all remaining variables p-values are below some threshold. **Mixed selection**: (Remedy the issues with forward and backward selection) This is a combination of forward and backward selection. We start with no variables in the model, and as with forward selection, we add the variable that provides the best fit. We continue to add variables one-by-one. The p-values for variables can become larger when new predictors are added to the model. Hence, if at any point the p-value for one of the variables in the model rises above a certain threshold, then we remove that variable from the model. We continue to perform these forward and backward steps until all variables in the model have a sufficiently low p-value, and all variables outside the model would have a large p-value if added to the model.
■**High-leverage points**: Observations with high leverage have an unusual value for x_i meaning the predictor value for this point is large relative to the other points. Removing high leverage points has a much more substantial impact on the least-squares line than removing outliers. High outliers have a sizable impact on the predicted line which is cause for concern if a few points have such heavy weight, for this reason it is important to identify high leverage points. They are easy to identify in simple linear regression but more difficult in multiple linear regression due to added dimensions. Hence, to quantify observations leverage we compute the leverage statistic. For simple linear regression this is: $h_i = \frac{1}{n} + \frac{(x_i-\bar{x})^2}{\sum_{i'=1}^{n}(x_{i'}-\bar{x})^2}$. It is clear that h_i increases with the distance of x_i from $\bar{x}$. The bounds are $\frac{1}{n} \leq h_i \leq 1$. The average leverage for all the observations is always equal to $\frac{p+1}{n}$. So, if a given observation has a leverage statistic that greatly exceeds $\frac{p+1}{n}$ then we can suspect the point has high leverage.
■**Collinearity**: Collinearity refers to the situation in which two or more predictor variables are closely related. This can make it difficult to separate the individual effects of the predictor on the response. Collinearity reduces the accuracy of the estimates of the regression coefficients, causing the standard error for $\hat{\beta}_j$ to grow. Therefore, effecting the test of H_0: $\hat{\beta}_j = 0$ and thus the power of the test (correctly detecting a non-zero coefficient) is reduced by collinearity. A simple way to detect collinearity is to look at the correlation matrix of predictors. An element in this matrix that has a high absolute value indicated a pair of highly correlated variables. Unfortunately, it is possible for collinearity to exist between three or more variables which cannot be seen in the covariance matrix (this is called **Multicollinearity**).
■**Multicollinearity**: Defined as collinearity which exists between three or more variables A better way to assess multicollinearity is to compute the Variance Inflation Factor (**VIF**). **VIF**: The smallest possible value of the VIF is 1, which indicates the complete absence of collinearity. Typically, in practice there is a small amount of collinearity among the predictors. As a rule of thumb, a VIF that exceeds 5 or 10 indicated a problematic amount of collinearity. $VIF\left(\hat{\beta}_j\right) = \frac{1}{1-R^2_{X_j|X_{-j}}}$. $R^2_{X_j|X_{-j}}$ is the R^2 from the regression of X_j onto all other predictors. If $R^2_{X_j|X_{-j}}$ is close to 1 then collinearity is present so the VIF will be large.

$P(Y=y|X) = p(X) = \beta_0 + \beta_1 X \rightarrow p(X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}} \rightarrow \frac{p(X)}{1-p(X)} = e^{\beta_0+\beta_1 X}$; $p > 1 \rightarrow p(X) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$ (1)

ODDS: $\frac{p(X)}{1-p(X)} \in [0,\infty)$. e.g. on average 1 in 5 people with an *odds* of $\frac{1}{4}$ will default. This is because, $\frac{1/5}{1-(1/5)} = \frac{1}{4}$.

$$\text{log-odds or logit} \rightarrow \quad \ln\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X \tag{2}$$

$$\text{posterior probability that an observation X=x belongs to k}^{\text{th}}\text{ class} \rightarrow \quad p_k(x) = P(Y=k|X=x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K}\pi_l f_l(x)} \tag{3}$$

$$\text{prior pdf that X=x belongs to k}^{\text{th}}\text{ class w/ ASSUMPTION follows N}(\mu_k,\sigma_k^2) \rightarrow \hat{f}_k(x) = \frac{1}{\sigma_k\sqrt{2\pi}}\exp\left\{-\frac{1}{2\sigma_k^2}(x-\mu_k)^2\right\} \tag{4}$$

$$\text{LDA common covariance matrix} \rightarrow p_k(x) = P(Y=k|X=x) = \frac{\pi_k \frac{1}{\sigma\sqrt{2\pi}}\exp\left\{-\frac{1}{2\sigma^2}(x-\mu_k)^2\right\}}{\sum_{l=1}^{K}\pi_l \frac{1}{\sigma\sqrt{2\pi}}\exp\left\{-\frac{1}{2\sigma^2}(x-\mu_l)^2\right\}} \tag{5}$$

$$\text{LDA discriminant} \rightarrow \delta_k(x) = x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \ln(\pi_k) \tag{6}$$

$$\hat{\mu}_k = \frac{1}{n_k}\sum_{i:y_i=k} x_i \quad ; \quad \hat{\sigma}^2 = \frac{1}{n-K}\sum_{k=1}^{K}\sum_{i:y_i=k}(x_i-\hat{\mu}_k)^2 \ \textit{and}\ \hat{\pi}_k = \frac{n_k}{n} \tag{7}$$

$$\hat{\delta}_k(x) = x\frac{\hat{\mu}_k}{\hat{\sigma}^2} - \frac{\hat{\mu}_k^2}{2\hat{\sigma}^2} + \ln(\hat{\pi}_k) \tag{8}$$

3.5 Comparisons of Linear Regression with K-Nearest Neighbors

Linear regression is a parametric approach because it assumes a linear functional form. Not robust. Suspect results if subject to X,Y that are not linear relationship. A non-parametric approach does not explicitly assume the form that X,Y should take, thereby are more flexible. One such nonparametric approach is K-Nearest Neighbors (KNN regression). Given a value for K and a prediction point x_0, KNN regression first identifies the K training observations closest to x_0, represented by N_0. It then estimates $f(x_0)$ using the average of all the training responses in N_0. In other words, $f(x_0) = \frac{1}{K}\sum_{N_0} y_i$. When $K = 1$ we get a step function that is perfectly fit to data, i.e., step function. When K is larger, we see that it is still a step function but has much smaller regions of constant prediction (meaning that we have an average for a larger area surrounding x_0 to predict x_0), and consequently a smoother fit.
KNN performs slightly worse than linear regression when the relationship is linear, but much better than linear regression for non-linear situations. However, note that there is a decrease in performance as the dimension increases which is a common problem for KNN, and results from the fact that in higher dimensions there is effectively a reduction in sample size.

4.3 Logistic Regression AND 4.3.1 The Logistic Model AND 4.3.4 Multiple Logistic Regression (multiple not that useful because we prefer LDA usually)

■Logistic regression models the *probability* that Y belongs to a particular category. For example, if we consider the data of those who $Default \in \{Yes, No\}$ based on $Balance$ and $Income$. We may use logistic regression to find the probability that a creditor $Defaults = Yes$ conditioned on the creditors credit $Balance$, i.e., $P(Default = Yes|Balance) \in [0,1]$. We may then make that a creditor will default if $P(Default = Yes|Balance) > 0.5$ or we could be conservative and risk averse and claim that a creditor will default if $P(Default = Yes|Balance) > 0.1$.
■How should we model the relationship between a qualitative response Y and quantitative variable X . Assume we code Y generically such that $y \in \{0,1\}$. Previously it was discussed that a linear regression model, $P(Y=y|X) = \beta_0 + \beta_1 X$, is flawed as we can get a negative prediction which is difficult to interpret as a probability. Therefore, although it will work for the binary qualitative response it is not ideal.
■Anytime a straight line is fit to a binary response that is coded as 0 or 1, in principle we can always predict $P(Y=y|X) < 0$ for some values of X and $P(Y=y|X) > 1$ for others (unless the range of X is limited). To avoid this issue, we use a function that outputs values between 0 and 1, the logistic function. **EQUATION 1** .
■***Equation* (2)** is called the *log-odds* or *logit* . We see that the logistic regression equation given by (2) has a *logit* that is linear in X. Recall that β_1 gives the average change in Y associated with a one-unit increase in X. In contrast, in a logistic regression model, increasing X by one unit changes the *log-odds* by β_1, or equivalently, it multiplies the *odds* by e^{β_1} . However, because the relationship between $p(X)$ and X in is not a straight line, β_1 does not correspond to the change in $p(X)$ associated with a one-unit increase in X. The amount that $p(X)$ changes due to a one-unit change in X will depend on the current value of X. But regardless of the value of X, if β_1 is positive then increasing X will be associated with increasing $p(X)$. If β_1 is negative, then increasing X will be associated with decreasing $p(X)$.

4.4.0 Linear Discriminant Analysis AND 4.4.1 Using Bayes' Theorem for Classification

■The two-class logistic regression models have multiple-class extensions, but in practice they tend not to be used all that often. One of the reasons is that the method we discuss in the next section, discriminant analysis, is popular for multiple-class classification. Linear discriminant analysis is popular when we have more than two response classes. In this alternative approach, we model the distribution of the predictors X separately in each of the response classes (i.e. given Y), and then use Bayes' theorem to flip these around into estimates for $P(Y = k|X = x)$.
■ Suppose that now we are considering a case of multiple categories that the response Y can take. In other words, we wish to classify an observation into one of K classes, where $K \geq 2$. Let π_k represent the prior probability that a randomly chosen observation comes from the k^{th} class; this is the probability that a given observation is associated with the k^{th} category of the response variable Y. Let $f_k(x) = P(X = x|Y = k)$ denote the pdf for an object that comes from the k^{th} class. Then Bayes' Theorem states: **EQUATION 3** . In general, estimating π_k is easy, compute the fraction of observations that belong to the k^{th} class over all observations in the training data. Estimating $f_k(X)$ is difficult unless we assume a simple underlying distribution. We refer to $P(Y = k|X)$ as the posterior probability that an observation belongs to the k^{th} class. That is, it is the probability that the observation belongs to the k^{th} class, given the predictor value for that observation.

4.4.2 Linear Discriminant Analysis for $p = 1$

■For now, assume $p = 1$, i.e., we only have one predictor. We would like to obtain an estimate for $f_k(x)$ that we can plug into (3) to estimate $P(Y = k|X = x)$. We will then classify an observation to the class for which $P(Y = k|X = x)$ is greatest. To estimate $f_k(x)$, we will first make some assumptions about its form. ■Suppose we assume that $f_k(x)$ is normal. Therefore, ***EQUATION* 4** where μ_k and σ_k^2 are the mean and variance parameters for the k^{th} class. For now, let us further assume that there is equal variances across all K classes, $\sigma_1^2 = \cdots = \sigma_K^2 = \sigma^2$. If we plug in this assumed pdf into (3) we get, ***EQUATION 5*** . The Bayes' classifier involves assigning an observation $X = x$ to the class for which (5) is largest. Taking the log of (5) and rearranging the terms, it is not hard to show that this is equivalent to assigning the observation to the class for which, ***EQUATION 6*** .
■Example: $K = 2$. $\pi_1 = \pi_2$. The Bayes' classifier involves assigning an observation $X = x$ to the class for which (5) is largest or equivalently (6) is the largest. Which class is assigned? $\delta_1(x) = x\frac{\mu_1}{\sigma^2} - \frac{\mu_1^2}{2\sigma^2} + \ln(\pi_1)$; $\delta_2(x) = x\frac{\mu_2}{\sigma^2} - \frac{\mu_2^2}{2\sigma^2} + \ln(\pi_2)$. Therefore, class 1 is assigned if and only if $\delta_1(x) > \delta_2(x)$. The decision boundary is given when $\delta_1(x) = \delta_2(x)$.
■The *linear discriminant analysis* (LDA) method approximates the Bayes' classifier by plugging estimates for π_k, μ_k , and σ^2 into (6). In particular, the following estimates are used: ***EQUATION 7*** . Where n is the total number of training observations and n_k is the number of observations in the k^{th} class. The LDA classifier plugs in the estimated parameters and assigns an observation $X = x$ to the class for which (8) is largest.
■To reiterate, the LDA classifier results from the assumption that the observations within each class come from a normal distribution with a class-specific mean vector and a common variance σ^2, and plugging estimates for these parameters into the Bayes classifier.

LDA

■ Assumes that the observations within each class are drawn from a multivariate normal distribution with a class specific mean vector μ_k and a covariance matrix Q that is common to all classes.
■Training error rates will usually be lower than test error rates, the latter being what is of interest. This is because we adjust the parameters based on the training data.
■ LDA tends to be a better bet than QDA if there are relatively fewer training observations and so reducing variance is crucial.
■ LDA is a special case of QDA . NOTE that in LDA the functions are estimates obtained by assuming $X_1, \ldots, X_p$ are normal whereas logistic regression the coefficients are chosen to maximize the likelihood.
■ LDA outperforms logistic regression when the normality assumption holds and logistic regression performs better when this assumption is violated.

QDA

■ Assumes that the observations within each class are drawn from a multivariate normal distribution with a class specific mean vector μ_k and a class specific covariance matrix Q_k ,i.e., an observation from the k^{th} class is of the form $X \sim N(\mu_k, Q_k)$. Under this assumption, the Bayes classifier assigns an observation $X = x$ to the class for which $\delta_k(x) = -\frac{1}{2}(x-\mu_k)^T Q_k^{-1}(x-\mu_k) - \frac{1}{2}\ln|Q_k| + \ln\pi_k$ is the largest.
■ The quantity x appears as a quadratic function in the discriminant function for QDA, $\delta_k(x)$. This is where QDA gets its name.
■ Why would one prefer QDA or LDA? The answer lies in the bias-variance trade off . When there are p predictors, then estimating a SINGLE covariance matrix requires estimating $p(p+1)/2$ parameters. QDA however, estimates a separate covariance matrix for each class, for a total of $Kp(p+1)/2$ parameters. If instead we assume that the K classes share a common covariance matrix, the LDA becomes linear in x, which means there are Kp linear coefficients to estimate.
■ Due to fewer coefficients to estimate, LDA is less flexible than QDA, and has a lower variance. In theory, this can lead to improved prediction performance. However, there is a tradeoff as if the assumption of LDA that there is a common covariance matrix is badly mistaken, then the LDA can suffer from high bias.
■ QDA is recommended if the training set is large, so that the variance of the classifier is not a major concern, or if the assumption of a common covariance matrix for the K classes is clearly untenable.
■ QDA includes multiplicative terms therefore, QDA has potential to be more accurate in setting where interactions among the predictors is important in discriminating between classes.

Naïve Bayes Classifier

■ Using Bayes theorem, (given K classes) we have an expression for the posterior probability $p_k(x) = P(Y = k|X = x)$ in terms of $\pi_1, \ldots, \pi_K$ and $f_1(x), \ldots, f_K(x)$. Estimating $\pi_1, \ldots, \pi_K$ is straightforward for example $\hat{\pi}_k$ can be estimated using the proportion of training data in k^{th} class over total # of observations . However, estimating $f_1(x), \ldots, f_K(x)$ from data is very difficult unless we know that $f_j(x)$ has a specific form, e.g. normal, in which case we only need to know the mean and variance.
■ Instead of assuming $f_1(x), \ldots, f_K(x)$ follow normal like in LDA and QDA, in Naïve Bayes Classifier we make a different (but equally strong) assumption **within the k^{th} class the p predictors are independent** . This means that the joint density is simply the product of the single densities, $f_{in}(x_1, \ldots, x_p) = f_{k1}(x_1) * \ldots * f_{kp}(x_p)$ where f_{kj} is the density function of the j^{th} predictor among the observations in the k^{th} class.
■ Why is this assumption so powerful? Normally, we could not only have to estimate each marginal density, but we could also have to estimate the joint density; now we do what is easier and not both.
■ Under the naïve bayes classifier the expression for the posterior probability is given by $P(Y = k|X = x) = \frac{\pi_k * f_{k1}(x_1) * \ldots * f_{kp}(x_p)}{\sum_{l=1}^{K}\pi_l * f_{l1}(x_1) * \ldots * f_{lp}(x_p)}$ for $k = 1, \ldots, K$.
■ To estimate the one-dimensional density function f_{kj} using training $x_{ij}, \ldots, x_{n_j}$ we have two options: **OPTION 1** → we assume that within each class the j^{th} predictor is drawn from a univariate normal distribution, i.e., $X_j|Y \sim N(\mu_{jk}, \sigma_{jk}^2)$. This may sound a lot like QDA, but there is a key difference: here we are assuming that the predictors are independent. This amounts to assuming that the **covariance matrix Q is diagonal**. **OPTION 2** → We simply count the proportion of training observations for the j^{th} predictor corresponding to each class. For example, suppose $X_j \in \{1,2,3\}$ and we have 100 observations in the k^{th} class. Suppose that the j^{th} predictor takes on values 1, 2, 3 in 32, 55, 13 of those observations respectively. Then we have $\widehat{f_{kj}} = \begin{cases} 0.32\,, if\ x_j = 1 \\ 0.55\,, if\ x_j = 2 \\ 0.13\,, if\ x_j = 3 \end{cases}$.
■ First: Any classifier with a linear boundary is a special case of naïve bayes. Second: If we model $f_{kj}(x_j)$ in the naïve bayes classifier using a one dimensional normal distribution $N(\mu_{jk}, \sigma_j^2)$ then we end up with $g_{kj}(x_j) = b_{kj}x_j$ where $b_{kj} = \left(\mu_{kj} - \mu_{kj}\right)/\sigma_j^2$. In this case naïve bayes is a special case of LDA. Third: neither LDA or naïve Bayes is a special case of the other.

Assign x to the class that maximizes or $P(Y = k|X = x)$ $\log\left(\frac{P(Y=k|X=x)}{(Y=K|X=x)}\right)$ equals $a_k + \sum_{j=1}^{p} b_{kj}x_j$ and $a_k + \sum_{j=1}^{p} b_{kj}x_j + \sum_{j=1}^{p}\sum_{l=1}^{p} c_{kjl}x_j x_l$ and $a_k + \sum_{j=1}^{p} g_{kj}(x_j)$ for LDA, QDA, and Naïve Bayes respectively.

Poisson Regression Model: $\lambda(X_1, \ldots, X_p) = \exp\{\beta_0 + \beta_1 X + \cdots + \beta_p X_p\}$. We estimate β_j using MLE. Likelihood $L(\beta_0, \beta_1, \ldots, \beta_p) = \prod_{i=1}^{n}[(\exp\{-\lambda(x_i)\}\lambda(x_i)^{y_i})/y_i!]$. Interpretation: an increase of $X_j$by one unit is associated with a change of $E[Y] = \lambda$ by a factor of $\exp\{\beta_j\}$ .Mean-variance relationship s.t. mean=variance. Lastly, there are no negative predictions using the Poisson regression model. Note that we take $\ln[\lambda(X_1, \ldots, X_p)]$ to be linear rather than λ itself.
Polynomial Regression Model : $y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \cdots + \beta_d x_i^d + \varepsilon_i$. Can be estimated using least squares by assigning $X, X^2, \ldots, X^d$ as predictors. In practice polynomials of degree 3 or 4 don't work well due to overfitting.
Regression Splines : We use cubic polynomials with knots. The points where coefficients change are called knots. If we use K knots we fit $K + 1$ cubic polynomials.
■**Poisson Regression Example** : Let X =average daily traffic volume ; Y = # of car accidents. λ denotes the average number of accidents per day. β_0 is the intercept representing the expected number of accidents when traffic volume is 0. β_1 is the coefficient representing how much the expected number of accidents changes with a one unit increase in the volume of traffic. If $\beta_0 = 0.005$, it means that for every additional 100 cars in daily traffic, the expected number of accidents increases by 0.5% . This is because $\exp\{0.005\} = 1.005$. To predict the number of accidents on a day with a traffic volume of 2000 cars, we use $\lambda = \exp\{\beta_0 + \beta_1 X\} = \exp\{\beta_0 + (0.005)(2000)\}$

So far we have studied three types of regression models: linear, logistic, Poisson. These approaches share some common characteristics. Each approach has predictors $X_1, \ldots, X_p$ to predict a response Y where we assume that $Y|X_1, \ldots, X_p$ belongs to a certain family of distributions. For linear regression we assume $Y|X_1, \ldots, X_p$ follows normal. For logistic regression we assume Bernoulli. For Poisson we assume Poisson. These are all members of the exponential family.

Linear regression: $E\left[Y|X_1, \ldots, X_p\right] = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$
Logistic regression : $E\left[Y|X_1, \ldots, X_p\right] = \exp\{\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p\}/(1 + \exp\{\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p\})$
Poisson regression $E\left[Y|X_1, \ldots, X_p\right] = \exp\{\beta_0 + \beta_1 X + \cdots + \beta_p X_p\}$

Question 6.1: Let X have a multivariate normal distribution $N(\mu, Q)$. Give a condition on the matrix Q that guarantees X has a density. Write a formula for the density. **Answer 6.1** : The conditions on Q that guarentees that X has a density are such that Q must be a square, symmetric, and positive semi-definite matrix, have inverse and determinant. Many of these conditions imply the other and only first three conditions are sufficient. If these conditions for Q are met, then the pdf exists and is given by:

$$X \sim N_k(\mu, Q) \equiv f_X(x) = \frac{1}{(2\pi)^{k/2}(\det Q)^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu)^T Q^{-1}(x-\mu)\right\}$$

Question 6.2: Recall that Ridge Regression shrinks the regression coefficients by imposing a penalty on their size. Indeed, the ridge coefficients minimize a penalized residual sum of squares,

Equation (1) : $\hat{\beta}^{\text{ridge}} = \underset{\beta}{\text{argmin}} \left[\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{p} x_{i,j}\beta_j\right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2\right]$

Show that the equivalent formulation of (1) is the following:

Equation (2) : $\hat{\beta}^{\text{ridge}} = \underset{\beta}{\text{argmin}} \left[\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{p} x_{i,j}\beta_j\right)^2\right]$ subject to $\sum_{j=1}^{p} \beta_j^2 \le t$.

Answer 6.2 : Using the Lagrange multiplier function, $\mathcal{L}(x, \lambda) = f(x) + \lambda * g(x)$. We convert the optimization problem in equation (2) into an unconstrained one: $\mathcal{L}(\beta, \lambda) = \sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{p} x_{i,j}\beta_j\right)^2 + \lambda\left(\sum_{j=1}^{p} \beta_j^2 - t\right)$. Minimizing $\mathcal{L}(\beta, \lambda)$ with respect to β (drops relevance of $-\lambda t$), $\hat{\beta}^{\text{ridge, unconstrained}} = \underset{\beta}{\text{argmin}} \sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{p} x_{i,j}\beta_j\right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$. ■

Question 6.3: If we write equation (1) in matrix form it is $RSS(\lambda) = (y - X\beta)^T(y - X\beta) + \lambda\beta^T\beta$. Show that the ridge regression solutions can be seen to be, $\hat{\beta}^{\text{ridge}} = (X^TX + \lambda I)^{-1}X^Ty$, where I is a $p \times p$ identity matrix. (Note that the solution adds a positive constant to the diagonal of X^TX before inversion. This makes the problem nonsingular, even if X^TX is not of full rank.)

Answer 6.3 : $\frac{d}{d\beta}\{RSS(\lambda)\} = \frac{d}{d\beta}\{(y - X\beta)^T(y - X\beta) + \lambda\beta^T\beta\} = \frac{d}{d\beta}\{(y - X\beta)^T(y - X\beta) + \lambda\beta^T\beta\} = \frac{d}{d\beta}\{(y^T - (X\beta)^T)(y - X\beta) + \lambda\beta^T\beta\} = \frac{d}{d\beta}\{(y^T - \beta^TX^T)(y - X\beta) + \lambda\beta^T\beta\} = \frac{d}{d\beta}\{y^Ty - y^TX\beta - \beta^TX^Ty + \beta^TX^TX\beta + \beta^T\beta\} = -2X^TY + 2X^TX\beta + 2\lambda\beta$

Finding minimum. $\frac{d}{d\beta}\{RSS(\lambda)\} = 0 \rightarrow (X^TX + \lambda I)2\beta = 2X^TY \rightarrow \hat{\beta}^{\text{ridge}} = (X^TX + \lambda I)^{-1}X^Ty$. ■

Question 6.4: In linear regression we have a vector β of coefficients of the p predictors. Let $\theta = \alpha^T\beta$ be a linear combination of the parameters. The least squares estimate of $\alpha^T\beta$ is $\hat{\theta} = \alpha\hat{\beta} = \alpha^T(X^TX)^{-1}X^Ty$. Consider X to be fixed, this is a linear function of c_0^Ty of the response vector y. Assume the linear model is correct and show that $\alpha\hat{\beta}$ is unbiased for $\alpha^T\beta$.

Answer 6.4 : $E[y] = X\beta \rightarrow E[\hat{\theta}] = E[\alpha\hat{\beta}] = E[\alpha^T(X^TX)^{-1}X^Ty] = \alpha^T(X^TX)^{-1}X^TE[y] = \alpha^T(X^TX)^{-1}X^TX\beta = \alpha^T\beta$.

Question 6.5: The Gauss-Markov theorem states that if we have any other linear estimator $\hat{\theta} = c^Ty$ that is unbiased for $\alpha^T\beta$, that is $E[c^Ty] = \alpha^T\beta$, then $Var(\alpha^T\hat{\beta}) \le Var(c^Ty)$. Prove the Gauss-Markov theorem.

Answer 6.5 : Since $\hat{\theta} = c^Ty$ is an unbiased estimator of $\alpha^T\beta$, it follows that $E[\hat{\theta}] = c^TX\beta = \alpha^T\beta$, implying that $c = X(X^TX)^{-1}\alpha + v$ holds for some v satisfying $v^TX = 0$. Next, the variance of $\hat{\theta}$ is given by $Var(\hat{\theta}) = Var(c^Ty) = c^TVar(y)c = \sigma^2c^TIc = \sigma^2\|c\|_2^2 = \sigma^2\alpha^T(X^TX)^{-1}\alpha + \sigma^2\|v\|_2^2$ where the equality holds because $v^TX = 0$. Finally, using the covariance matrix of $\hat{\beta}$, we have $Var(\hat{\theta}) = \sigma^2\alpha^T(X^TX)^{-1}\alpha + \sigma^2\|v\|_2^2 \ge \sigma^2\alpha^T(X^TX)^{-1}\alpha = Var(\alpha^T\hat{\beta})$ ■

Question 6.6: Suppose a data set contains a higher number of predictor variables than the number of observations. Suppose in addition multicollinearity is suspected in the multiple regression data. **(Question A)** Explain what is meant by multicollinearity **(Question B)** In the above circumstance what would you choose? Simple linear regression, Ridge Regression, or LASSO? **(Question C)** Suppose the number of significant parameters is relatively small and the others are close to zero, so that in effect only a few predictors influence the response. Which of the three (simple linear, Ridge, or LASSO) is the best to use?

Answer 6.6A : Multicollinearity occurs when predictor variables are highly correlated, making it difficult to determine their individual effects on the response. It can also lead to unstable coefficient estimates and inflated standard errors.

Answer 6.6B : In the presence of multicollinearity and more predictors than observations, Ridge Regression is preferred as it stabilizes coefficient estimates by imposing a penalty on their magnitude. Formally speaking, ridge regression can make the matrix $X^TX + \lambda I$ in (1) nonsingular by adding a small positive constant to its diagonal elements.

Answer 6.6C : If only a few predictors significantly influence the response, LASSO is the best choice because it performs variable selection by shrinking some coefficients to exactly zero.

Question 5.1: This problem relates to the QDA model, in which the observations within each class are drawn from a normal distribution with a class specific mean vector and a class specific covariance matrix. We consider the simple case where $p = 1$; that is, there is only one feature. Suppose that we have K classes and that if an observation belongs to the k^{th} class, then X comes from a one-dimensional normal distribution with $X \sim N(\mu_k, \sigma_k^2)$. Prove that in this case the Bayes classifier is not linear, but it is actually quadratic.

Answer 5.1: Given $p = 1 \rightarrow Y = \beta_0 + \beta_1X$. π_k = prior probability that a randomly chosen observation comes from the k^{th} class. Furthermore, $f_k(x) \equiv P(X = x|Y = k)$ is the density function of X for an observation that comes from the k^{th} class.

Using Bayes' Theorem, we get $P(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}$. To estimate $f_k(x)$ we assume $f_k(x) \sim N(\mu_k, \sigma_k^2)$, therefore an estimate of $f_k(x)$ is given by $\hat{f}_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k}\exp\left\{-\frac{1}{2\sigma_k^2}(x - \mu_k)^2\right\}$. If we do **not** assume $\sigma_1^2 = \cdots = \sigma_K^2 = \sigma^2$ as observations within each class are drawn from a normal distribution with a **class-specific** mean vector and a **class specific** co- variance matrix, then plugging in $\hat{f}_k(x)$ for $f_k(x)$ yields $P(Y = k|X = x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma_k}\exp\left(-\frac{1}{2\sigma_k^2}(x-\mu_k)^2\right)}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\sigma_l}\exp\left(-\frac{1}{2\sigma_l^2}(x-\mu_l)^2\right)} =$

$\frac{\frac{\pi_k}{\sigma_k}\exp\left(-\frac{1}{2\sigma_k^2}(x-\mu_k)^2\right)}{\sum_{l=1}^{K} \frac{\pi_l}{\sigma_l}\exp\left(-\frac{1}{2\sigma_l^2}(x-\mu_l)^2\right)}$. Then taking the advantage of the properties of natural log we can simplify this to,

$\ln\left(P(Y = k|X = x)\right) = \ln\left(\frac{\frac{\pi_k}{\sigma_k}\exp\left(-\frac{1}{2\sigma_k^2}(x-\mu_k)^2\right)}{\sum_{l=1}^{K} \frac{\pi_l}{\sigma_l}\exp\left(-\frac{1}{2\sigma_l^2}(x-\mu_l)^2\right)}\right) = \ln \pi_k - \ln \sigma_k - \frac{1}{2\sigma_k^2}x^2 + \frac{\mu_k}{\sigma_k^2}x - \ln\left(\sum_{l=1}^{K} \frac{\pi_l}{\sigma_l}\exp\left\{-\frac{1}{2\sigma_l^2}(x - \mu_l)^2\right\}\right)$.

Since we seek to drop all terms independent of k to find an equivalent objective function to $\ln\left(P(Y = k|X = x)\right)$ given by $\delta_k(x)$ that is maximized for some observation in the k^{th} class. To do this we drop the term $-\ln\left(\sum_{l=1}^{K} \frac{\pi_l}{\sigma_l}\exp\left\{-\frac{1}{2\sigma_l^2}(x - \mu_l)^2\right\}\right)$ from the above equation. $\delta_k(x) = \ln \pi_k - \ln \sigma_k - \frac{1}{2\sigma_k^2}x^2 + \frac{\mu_k}{\sigma_k^2}x - \frac{\mu_k^2}{2\sigma_k^2}$. We see that $\delta_k(x)$ is quadratic as it has a term of 2nd order. ■

Question 5.2: In this exercise, we examine the difference between LDA and QDA. **(Question A)** If the Bayes decision boundary is linear, do we expect LDA or QDA to perform better on the training set? On the test set? **(Question B)** If the Bayes decision boundary is non-linear, do we expect LDA or QDA to perform better on the training set? On the test set? **(Question C)** In general, as the sample size n increases, do we expect the test prediction accuracy of QDA relative to LDA to improve, decline , or be unchanged? Why? **(Question D)** True or False: Even if the Bayes decision boundary for a problem is linear, we will probably achieve a superior test error rate using QDA rather than LDA because QDA is flexible enough to model a linear decision boundary. Justify your answer.

Answer 5.2A : QDA is expected to perform better on the training set as it is more flexible, but it will suffer from overfitting on the test set. Therefore, QDA> LDA for training set and QDA<LDA for test set.

Answer 5.2B : QDA is expected to perform better in the training data and test set when the decision boundary is non-linear.

Answer 5.2C : QDA is recommended if the training set is very large, so that the variance of the classifier is not a major concern, or if the assumption of a common covariance matrix for the K classes is clearly untenable. Therefore, as n becomes large, the test prediction accuracy of QDA relative to LDA is expected to **improve**.

Answer 5.2D : False. If the Bayes decision boundary is linear, due to the QDAs flexibility it is expected to perform better on the training set but will yield a worse test error rate due to overfitting compared to LDA.

Question 5.4: Suppose that in the regression framework there are relatively few training observations and so reducing variance is crucial. Which would you prefer LDA or QDA?

Answer 5.4: LDA as if n is small, LDA tends to be a better bet than QDA as reducing variance is crucial.

Question 5.5: Suppose that in the regression framework the training set is large, so the variance of the classifier is not a major concern. Which would you prefer LDA or QDA?

Answer 5.5: QDA is recommended if the training set is very large, so that the variance of the classifier is not a major concern .

Question 5.6: Suppose the assumption of a common covariance matrix for the K classes is clearly untenable. LDA or QDA?

Answer 5.6: QDA. If we were to use LDA this assumption is wrong and creates far too much bias.

Question 4.4: Classifying an observation to the class which $p_k(x) = \frac{\pi_k \frac{1}{\sigma\sqrt{2\pi}}\exp\left\{-\frac{1}{2\sigma^2}(x-\mu_k)^2\right\}}{\sum_{l=1}^{K} \pi_l \frac{1}{\sigma\sqrt{2\pi}}\exp\left\{-\frac{1}{2\sigma^2}(x-\mu_l)^2\right\}} = \frac{\pi_k \exp\left\{-\frac{1}{2\sigma^2}(x-\mu_k)^2\right\}}{\sum_{l=1}^{K} \pi_l \exp\left\{-\frac{1}{2\sigma^2}(x-\mu_l)^2\right\}}$ is largest is equivalent to classifying an observation to for which $\delta_k(x) = x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \ln(\pi_k)$ is largest. In other words, under the assumption that the k^{th} classes are drawn from a $N(\mu_k, \sigma^2)$ distribution, the Bayes classifier assigns an observation to the class for which the discriminant function is maximized. Prove this.

Answer 4.4: Maximizing $p_k(x)$ is equivalent to maximizing the numerator of $\ln p_k(x)$ as the denominator of $p_k(x)$ does not depend on k. Therefore, we seek to maximize $\ln \pi_k - \frac{1}{2\sigma^2}(x - \mu_k)^2$. Dropping all terms with no dependence to k yields, $\delta_k(x) = x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \ln(\pi_k)$ ■

4.5 A Comparison of Classification Methods.

When is LDA, QDA, Logistic Regression, KNN best choice? Though their motivations differ, the logistic regression and LDA methods are closely connected. Consider the two-class setting with $p = 1$ predictor and let $p_1(x)$ and $p_2(x) = 1 - p_1(x)$ be the probabilities that the observation $X = x$ belong to class 1 and class 2 , respectively. In the LDA framework, we can see that the log-odds are given by $\ln\left(\frac{p_1(x)}{1-p_1(x)}\right) = \ln\left(\frac{p_1(x)}{p_2(x)}\right) = c_0 + c_1x$ where c_0, c_1 are functions of μ_1, μ_2, and σ^2 . We know that in logistic regression, $\ln\left(\frac{p_1}{1-p_1}\right) = \beta_0 + \beta_1x$. Both equations are linear functions of x hence they both produce linear decision boundaries. The only difference between the two approaches lies in the fact that β_0 and β_1 are estimated using MLE, whereas c_0 and c_1 are computed using the estimated mean and variance from a normal distribution. This connection between logistic regression and LDA also holds for $p > 1$.

■LDA/ logistic regression. Since LDA and logistic regression differ only in their fitting procedures, one might expect the two approaches to give similar results. This is often but not always the case. LDA assumes the observations are drawn from a Gaussian distribution with a common covariance matrix in each class, and so can provide some improvements over logistic regression when this assumption approximately holds. Conversely, logistic regression can outperform LDA if these Gaussian assumptions are not met.

■KNN. In order to make a prediction for an observation $X = x$, the K training observations that are closest to x are identified. Then x is assigned to the class to which the plurality of these observations belong. Hence, KNN is a completely nonparametric approach: no assumptions are made about the shape of the decision boundary. Therefore, we can expect KNN approach to dominate LDA and logistic regression when the decision boundary is highly non-linear. On the other hand, KNN does not tell us which predictors are important.

■QDA. Finally, QDA serves as a compromise between the non-parametric KNN method and the linear LDA and logistic regression approaches. Since QDA assumes a quadratic decision boundary, it can accurately model a wider range of problems than the linear methods. Though not as flexible as KNN, QDA can perform better in the presence of a limited number of training observations because it does make some assumptions about the form of the decision boundary.

■EXAMPLES. To illustrate the performances of these four classification approaches, we generated data from six different scenarios. In each of the six scenarios, there were $p = 2$ predictors. The scenarios were as follows:

■*Scenario 1 (Linear):* There were 20 training observations in each of two classes. The observations in each class were uncorrelated random normal variables with a different mean in each class. LDA performed well in this setting, as one would expect since this is the model assumed by LDA. KNN performed poorly because it paid the price in terms of variance that was not offset by a reduction in bias. QDA also performed worse than LDA, since it fit a more flexible classifier than necessary. Since logistic regression assumes a linear decision boundary, its results were only slightly inferior to those of LDA.

■*Scenario 2 (Linear):* Details are those in Scenario 1, except that within each class, the two predictors had a correlation of -0.5. The test yielded little change in the relative performances of the methods as compared to the previous scenario. Naïve Bayes performs poorly as the assumption of independence is violated.

■*Scenario 3 (Linear):* We generated X_1 and X_2 from the t distribution, with 50 observations per class. The t distribution has a similar shape to the normal distribution, but it has a tendency to yield more extreme points—that is, more points that are far from the mean. In this setting, the decision boundary was linear and so fit into the logistic regression framework. The set-up violated the assumptions of LDA, since the observations were not drawn from a normal distribution. Logistic regression outperformed LDA, though both methods were superior to the other approaches. In particular, the QDA results deteriorated considerably as a consequence of non-normality. Naïve Bayes again performed poorly as the assumption of independence was violated.

■*Scenario 4 (non-linear)*: The data were generated from a normal distribution, with a correlation of 0.5 between the predictors in the first class, and correlation of -0.5 between the predictors in the second class. This setup corresponded to the QDA assumption and resulted in quadratic decision boundaries. QDA outperformed all other approaches. Naïve Bayes assumptions violated and hence did poor.

■*Scenario 5 (non-linear)*: Within each class, the observations were generated from a normal distribution with uncorrelated predictors. However, the responses were sampled from the logistic function using X_1^2 , X_2^2, and $X_1 \times X_2$ as predictors. Consequently, there is a quadratic decision boundary. QDA performs best, closely followed by KNN. The linear methods had poor performance.

■*Scenario 6 (non-linear)*: Details are as in the previous scenario, but the responses were sampled from a more complicated non-linear function. As a result, even the quadratic decision boundaries of QDA could not adequately model the data. KNN performs best. QDA did slightly better than the linear models. But $K = 1$ gave the worst results; this highlights the fact that even when the data exhibits a complex non-linear relationship, a non-parametric method such as KNN can still give poor results if the level of smoothness is not chosen correctly.

■*Scenario 7*: observations come from normal distribution with a different diagonal covariance matrix for each class. The sample size is small, $n = 6$. The Naïve bayes does well as the assumptions are met (diagonal matrix). QDA does slightly worse due to small sample size which led to too much variance in estimating the correlation between predictors in each class.

■These six examples illustrate that no one method will dominate the others in every situation. When the true decision boundaries are linear, then the LDA and logistic regression approaches will tend to perform well. When the boundaries are moderately non-linear, QDA may give better results. Finally, for much more complicated decision boundaries, a non-parametric approach such as KNN can be superior. But the level of smoothness for a non-parametric approach must be chosen carefully.

■**Definition**: An $\mathbb{R}^n$-valued random variable $X = (X_1, \ldots, X_n)$ is Gaussian or Multivariate Normal if every linear combination $\sum_{j=1}^{n} a_jX_j$ has a one-dimensional Normal distribution.

■**Theorem**: X is an $\mathbb{R}^n$-valued random variable if and only if its characteristic function has the form $Var_X(u) = \exp\left\{i\langle u, \mu\rangle - \frac{1}{2}\langle u, Qu\rangle\right\}$ where $\mu \in \mathbb{R}^n$ and Q is an $n \times n$ symmetric nonnegative semi-definite matrix. Q is the covariance matrix of X and μ is the mean of X.

■**Example** : Let $X_1, \ldots, X_n$ be an $\mathbb{R}^n$-valued **independent** random variables with laws $N(\mu_j, \sigma_j^2)$. Then $X = (X_1, \ldots, X_n)$ is Multivariate Normal as $Var_X(u_1, \ldots, u_n) = \prod_{j=1}^{n} Var(u_j) = \exp\left\{i\langle u, \mu\rangle - \frac{1}{2}\langle u, Qu\rangle\right\}$.

■**Theorem**: Let $X_1, \ldots, X_n$ be an $\mathbb{R}^n$-valued Multivariate Normal random variable. The components X_j are independent if and only if the covariance matrix Q of X is diagonal.

Method, Key Context, Main Strength, Main Limitation.
Simple Linear Regression, linear relationships ; small datasets, easy to interpret, cannot model non-linear.
Cubic Splines, non-linear but smooth relationships, flexible fitting, risk of overfitting which yields poor predictive performance.
Ridge Regression, handles multicollinearity well; high-dimensional , reduces overfitting, doesn't perform feature selection.
LASSO, high-dimensional ; feature selection, shrinks and selects predictors, struggles with correlated features.

Ridge Regression (uses L2 penalty): $RSS + \lambda \sum_{j=1}^{p} \beta_j^2 = \sum_{i=1}^{n}\left[y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij}\right]^2 + \lambda \sum_{j=1}^{p} \beta_j^2$

LASSO Regression (uses L1 penalty) : $RSS + \lambda \sum_{j=1}^{p} |\beta_j| = \sum_{i=1}^{n}\left[y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij}\right]^2 + \lambda \sum_{j=1}^{p} |\beta_j|$

If $n \gg p$, then the least squares estimates tend to have low variance; good predictive performance. If $n > p$ (and not $n \gg p$) , then there can be a lot of variability in least squares leading to overfitting ; poor predictive performance. If $n < p$, there is no longer a unique least squares estimate ; the variance is infinite so the method cannot be used. HOWEVER, if we constrain or shrink the coefficients, we can sometimes reduce the variance at the cost of a tiny increase in bias.

■ It is often the case that some or even many of the variables used in a multiple regression model are in fact not associated with the response. By shrinking the coefficients of these variables to 0 we can obtain a model that is more easily interpreted. Least squares does not give 0 coefficients, that is where the penalty function comes in, i.e., shrinkage methods: Two shrinkage methods**: LASSO** and **Ridge Regression**

■Ridge Regression: $RSS + \lambda \sum_{j=1}^{p} \beta_j^2 = \sum_{i=1}^{n}\left[y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij}\right]^2 + \lambda \sum_{j=1}^{p} \beta_j^2$. The quantity $\lambda \ge 0$ is called a tuning parameter. $\lambda \sum_{j=1}^{p} \beta_j^2$ is called the shrinkage penalty; its small when $\beta_1, \ldots, \beta_p$ are close to 0 so it has the effect of shrinking the estimates of β_j towards 0. When $\lambda = 0$ the penalty has no effect and it is a least squares problem. As $\lambda \to \infty$, the impact of the shrinkage penalty grows sending the ridge regression estimates towards 0 . For each value of λ we get a different set of the estimates of the β_j ; selecting a good value for λ is crucial. With least squares we have only one estimate. How does ridge improve over LS? It is all about the bias variance trade-off. As λ increases, leads to decreased variance but increased bias.

■LASSO Regression : $RSS + \lambda \sum_{j=1}^{p} |\beta_j| = \sum_{i=1}^{n}\left[y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij}\right]^2 + \lambda \sum_{j=1}^{p} |\beta_j|$. ADVANTAGE; lets us completely shrink some variables to 0, whereas Ridge Regression always includes all p variables. LASSO performs **Variable selection** .

■ **LASSO** shrinks all coefficients towards zero by a similar amount, and sufficiently small coefficients are shrunken all the way to zero. **Ridge Regression** shrinks every dimension of the data by the same proportion .

■ Principal Component Analysis (PCA) is a popular approach for deriving a low-dimensional set of features from a large set of variables.

LOOCV = Leave One Out Cross Validation . LASSO =Least Absolute Shrinkage and Selection Operator

Attain tuning parameter through cross validation. LOOCV leaves one data point out trains the rest. Iterate. This is bad for large data. K-fold cross validation is used with more data and splits data into k groups.