Solution 1(a)
(i)
The paper introduces two statistical models' approaches proportional odds model and proportional hazards model for analyzing ordinal data, i.e., the data where target variable have a natural ordering but the intervals between adjacent categories are not necessarily equal.
Proportional Odds Model: It is based on assumption that the odds of falling into a particular category, or a higher category are proportional across different levels of the predictor variables i.e. the relationship between the predictors and the outcome is consistent across the different ordinal categories.

Proportional Hazards Model: It deals with time to event data and is used to assess the relationship between predictor variables and the hazard rate,i.e the instantaneous probability of the event occurring at a given time, while assuming that the hazard ratios are constant over time.
The proportional odds model and proportional hazards model have the same following general form.

$$\text{link}\{\gamma_j(\mathbf{x})\} = \theta_j - \boldsymbol{\beta}^{\mathrm{T}}\mathbf{x},$$

Where link is the complementary log-log function.

The paper introduces multiple examples for both, estimates the weights values and provides technique for Maximum likelihood Estimation for proportional odds and hazard model.

(ii)
Difference between likelihood function and odds ratio for ordinal regression and Multiclass classification is as follows.

Likelihood function

In **Ordinal Regression**
It is cumulative distribution of an ordinal response, such as the cumulative logit or probit distribution.
In **Multiclass classification**
It models the probability of each class or category within the response variable. Each class is treated as a separate category, and the likelihood is estimated using categorical distributions like the multinomial distribution.

Odds Ratio

In **Ordinal Regression**
It measures the odds of an observation being in or below a certain ordinal category relative to the odds of being in a higher category.
In **Multiclass classification**
It is not typically used as classes do not follow a particular order. Instead, the class probabilities are being used.

(iii) Difference between ordinal regression and other Regression algorithms like linear regression, polynomial regression etc.

Ordinal Regression	**Regression**
It includes cumulative odds ratios i.e., cumulative probabilities for each category relative to a reference category. It models how changes in predictor variables impact the cumulative odds of being in or below a specific category compared to higher categories.	It includes the coefficients corresponding to different features that represent the change in the expected value of the dependent variable for a one-unit change in the predictor variable.
It models ordered categorical variable	It models continuous variable
It makes assumption for constant odds ratio or constant hazard rate for proportional odds or proportional hazards model respectively.	It assumes linearity in case of linear regression.

1 (b) The proportional odds model as defined in paper is of the form

$$k_j(x) = k_j \exp(-\beta^T X)$$

or $$\log\left(\frac{\gamma_j(x)}{1-\gamma_j(x)}\right) = \theta_j - \beta^T X \quad (1 \le j < k)$$

where $\frac{\gamma_j(x)}{1-\gamma_j(x)}$ is the odds ratio for the event $Y \le j$.

& $\gamma_j(x) = \pi_1(x) + \pi_2(x) \cdots \pi_j(x)$

i.e $= \sum_{n=1}^{j} \pi_n(x)$.

Now as the contribution from a single multinomial observation $(n_1\ n_2 \cdots n_k)$ to the likelihood function is $\pi_1^{n_1}\ \pi_2^{n_2} \cdots \pi_k^{n_k}$ with probabilities satisfying $\text{link}(\gamma_{ij}) = (\theta_j - \beta^T x_i)/\tau_i$.

for the cumulative probabilities in ordinal regression, we define

$R_1 = n_1$ $\quad$ $Z_1 = R_1/n$

$R_2 = n_1 + n_2$ $\quad$ $Z_2 = R_2/n$

$\vdots$ $\quad$ $\vdots$

$R_k = \sum n_j = n$ $\quad$ $Z_k = R_k/n$.

1(b)

Now for Maximum likelihood Estimate we find

$$P(X/\theta).$$

i.e $\underset{\theta}{\operatorname{argmax}}\, P(X/\theta) = \prod_{i=1}^{K} P(X_i/\theta)$

$$P(X_i/\theta) = P(R_i : R_{i+1} - R_i \mid R_{i+1})$$

i.e probability that R_i gets divided in the ratio R_i & $R_{i+1} - R_i$

$$= \left(\frac{r_i}{r_{i+1}}\right)^{R_i} \left(\frac{r_{i+1} - r_i}{r_{i+1}}\right)^{R_{i+1} - R_i}.$$

$$\Rightarrow P(X_i/\theta) = \prod_{i=1}^{K} \left\{ \left(\frac{r_i}{r_{i+1}}\right)^{R_i} \left(\frac{r_{i+1} - r_i}{r_{i+1}}\right)^{R_{i+1} - R_i} \right\}$$

Taking log on both sides

$$\underset{\theta}{\operatorname{argmax}}\, L(\theta) = \sum_{i=1}^{K} \left\{ R_i \log\left(\frac{r_i}{r_{i+1}}\right) + (R_{i+1} - R_i) \log\left(\frac{r_{i+1} - r_i}{r_{i+1}}\right) \right\},$$

$$= \sum_{i=1}^{K} \left\{ R_i \log\left(\frac{r_i}{r_{i+1} - r_i}\right) - R_{i+1} \log\left(\frac{r_{i+1}}{r_{i+1} - r_i}\right) \right\}$$

$$= n \sum_{i=1}^{K} \left\{ z_i \phi_i - z_{i+1} g(\phi_i) \right\} \rightarrow (2)$$

where $z_i = R_i/n$

$$\phi_i = \log\left(\frac{r_i}{r_{i+1} - r_i}\right)$$

$$g(\phi_i) = \log\left(\frac{r_{i+1}}{r_{i+1} - r_i}\right).$$

1(b) Now putting

$$E\left[\frac{\partial L}{\partial \phi}\right] = 0.$$

we get $$E\left[\frac{Z_i}{Z_{i+1}}\right] = Z_{i+1} g'(\phi_i)$$

$$= \frac{Z_{i+1}}{r_{i+1}} + r_i$$

$$\Rightarrow \quad E[Z_i] = r_i .$$

Now as $\text{Var}(Z_i) = -\frac{\partial L}{\partial \theta^2}$ for one parameter exponential family for a glm

Therefore
$\Rightarrow$ differentiating equation (2) two times we get

$$\text{Var}\left(\frac{Z_i}{Z_{i+1}}\right) = \frac{Z_{i+1}\, g''(\phi_i)}{n}$$

$$\Rightarrow \quad \text{Var}\left(Z_i\right) = \frac{r_i(1-r_i)}{n}.$$

1 c.

Problem Statement
Given the various attributes of two type of wine data develop a ordinal regression and linear regression model and compare their performance as observed on the wine-quality prediction dataset.

Dataset
The dataset contains two csv files corresponding to red and white wine type and has 11 independent variables and target variable 'quality' is ordered and ranges between 3 to 9

Approach

1. Load the dataset and perform extrapolatory data analysis to understand the data and distribution of various features.
2. Perform one-way Anova test to understand if the if red and white wine are statistically significant from each other or not.
3. Check for presence of correlation among features and perform outlier detection and removal.
4. We then standardize the dataset and split the data into training and test set
5. We use Linear Regression and Ordinal Regression to train the model and compare it on following parameters.
    a. Log-Likelihood
    b. Akaike Information Criterion
    c. Bayesian Information Criterion
    d. Mean Squared Error (Macro Averaged)
    e. Mean Absolute Error (Macro Averaged)

Note: We have used Macro Averaged MSE and MAE as data is imbalanced

Results

We observe following results on red wine and white wine types.

Red Wine Data						
	Log Likelihood	**AIC**	**BIC**	**MSE**	**MAE**	**Accuracy**
Ordinal Regression	-1553.7	3129	3188	1.71	1.07	64.07
Linear Regression	-1278.6	2571	2607	1.51	1.06	NA

White Wine Data						
	Log Likelihood	**AIC**	**BIC**	**MSE**	**MAE**	**Accuracy**
Ordinal Regression	-5321.5	1.07E+04	1.07E+04	3.25	1.44	54.12
Linear Regression	-4321	8660	8716	3.346	1.48	NA

Observations

1. For White Wine Data Ordinal Regression slightly outperforms Linear Regression across multiple metrics.
2. For Red Wine Data Linear Regression has better Log Likelihood, AIC and BIC values indicating a better fit in case of Linear Regression, but Ordinal Regression MSE and MAE values are comparable to that of Linear Regression indicating a similar overall loss.

2(a) Likelihood function

$$f(x_1\ x_2\ \cdots\ x_n/w) = f(x_1/w)\cdot f(x_2/w)\cdots f(x_n/w)$$

$$= \prod_{i=1}^{n} f(x_i/w)$$

considering $x_1\ x_2\ \cdots\ x_n$ are i.i.d distributed.

Considering a Gaussian distribution, likelihood function becomes

(i) $$L(w; \phi(x_n)) = \prod_{n=1}^{n=N} \frac{1}{\sqrt{2\pi\sigma_n^2}} \exp\left(-\frac{\theta_n (t_n - w^T \phi(x_n))^2}{2\sigma_n^2}\right)$$

where θ_n = weight associated to n^{th} data pt.

σ_n^2 = variance associated to n^{th} error data pt.

t_n = actual value

x_n = input for the n^{th} data pt.
i.e n^{th} data pt.

(ii) Assuming a Gaussian prior, for a weight vector w prior is

$$p(w) = \frac{1}{\sqrt{(2\pi)^N |\Sigma_w|}} \exp\left(\frac{-1}{2}(w-\mu_w)^T \Sigma_w^{-1} (w-\mu_w)\right)$$

where N = dimension of weight vector w.

μ_w = mean vector of weight vector w.

$|\Sigma_w|$ = determinant of covariance matrix among elements of w.

Σ_w^{-1} = Inverse of covariance matrix.

2(b).

(ii) MLE parameter Estimation

Assuming a Gaussian distribution, likelihood function is

$$P(X/w) = \prod_{n=1}^{n=N} \frac{1}{\sqrt{2\pi\sigma_n^2}} \exp\left(-\frac{\theta_n (t_n - w^T(x_n))^2}{2\sigma_n^2}\right)$$

To estimate parameters we find $\underset{w}{\operatorname{argmax}} L(w; \phi(x_n))$

Therefore

$$L(w; \phi(x_n)) = \underset{w}{\operatorname{argmax}} \prod_{n=1}^{n=N} \frac{1}{\sqrt{2\pi\sigma_n^2}} \exp\left(-\frac{\theta_n (t_n - w^T(x_n))^2}{2\sigma_n^2}\right)$$

Taking Log on both sides

$$\underset{w}{\operatorname{argmax}} \log L(w; \phi(x_n))$$

$$= \underset{w}{\operatorname{argmax}} \sum_{n=1}^{n=N} \left\{ -\frac{1}{2} \log(2\pi\sigma_n^2) + \frac{1}{2}\left(\frac{-\theta_n (t_n - w^T(\phi_0(x_n))^2}{\sigma_n^2}\right)\right\}$$

$$= \underset{w}{\operatorname{argmax}} \sum_{n=1}^{n=N} \left\{ \frac{-1}{2} \frac{\theta_n (t_n - w^T\phi(x_n))^2}{\sigma_n^2} \right\}$$

$$= \underset{w}{\operatorname{argmin}} \sum_{n=1}^{n=N} \left\{ \frac{1}{2} \frac{\theta_n (t_n - w^T\phi(x_n))^2}{\sigma_n^2} \right\} \rightarrow (1)$$

Hence equation (1) is the expression for MLE estimation

where σ_n^2 = variance of error for n^{th} data point

θ_n = weight associated with n^{th} data point

t_n = actual value ~~with~~ of n^{th} data point

x_n = features for n^{th} data point.

2. (b) (ii) MAP Estimation.

$P(w/x) = P(x/w) \cdot P(w)$.

Taking log on both sides.

$$\underset{w}{\operatorname{argmax}}\, P(w/x) = \underset{w}{\operatorname{argmax}} \log(P(X/w)) + \underset{w}{\operatorname{argmax}} \log(P(w))$$

$$= \underset{w}{\operatorname{argmax}} \log(P(X/w)) + \underset{w}{\operatorname{argmax}} \log(P(w))$$

I II

Now from eq-① we know that

$$\underset{w}{\operatorname{argmax}}\, P(X/w) = \underset{w}{\operatorname{argmin}} \frac{1}{2} \left\{ \frac{\theta_n (t_n - w^T(\phi(x_n))^2}{\sigma_n^2} \right\}.$$

Also for II we assume a Gaussian Prior for a weight vector w as in 2(a)(ii).

$$\underset{w}{\operatorname{argmax}}\, P(w) = \frac{1}{\sqrt{(2\pi)^N |\Sigma_w|}} \exp\left(\frac{-1}{2} (w - \mu_w)^T \Sigma_w^{-1} (w - \mu_w) \right)$$

$\Rightarrow$

$$\underset{w}{\operatorname{argmax}}(P(w/x) = \underset{w}{\operatorname{argmin}} \frac{1}{2} \left\{ \frac{\theta_n (t_n - w^T(\phi(x_n))^2}{\sigma_n^2} \right\}.$$

$$+ \underset{w}{\operatorname{argmax}} \left\{ \frac{-1}{2} \log(2\pi |\Sigma_w|) - \frac{1}{2} (w - \mu_w)^T \Sigma_w^{-1} (w - \mu_w) \right\}.$$

$$= \underset{w}{\operatorname{argmin}} \frac{1}{2} \left\{ \frac{\theta_n (t_n - w^T(\phi(x_n))^2}{\sigma_n^2} \right\} +$$

$$\underset{w}{\operatorname{argmin}} \left\{ \frac{1}{2} (w - \mu_w)^T \Sigma_w^{-1} (w - \mu_w) \right\}.$$

Assuming mean vector $\mu_w = 1$ the above equation further reduces to

$$= \underset{w}{\operatorname{argmin}} \frac{1}{2} \left\{ \left(\frac{\theta_n (t_n - w^T (\phi(x_n))^2}{\sigma_n^2} \right) + w^T \Sigma_w^{-1} w \right\}$$

$$= \underset{w}{\operatorname{argmin}} \frac{1}{2} \left\{ \theta_n \frac{(t_n - w^T \phi(x_n))^2}{\sigma_n^2} + w^T \Sigma_w^{-1} w \right\}.$$

where Σ_w^{-1} = inverse of covariance matrix ~~across~~ created by amoung elements of w vector w.

μ_w = mean of vector w.

θ_n = weight associated with nth data point

σ_n^2 = variance of error associated with n^{th} data point.

2(c).

(i) MLE objective of linear regression in heteroscedastic setting

$$= \underset{w}{\operatorname{argmin}} \frac{1}{2} \left\{ \frac{\theta_n (t_n - w^T \phi(x_n))^2}{\sigma_n^2} \right\}$$

$$= \underset{w}{\operatorname{argmin}} \frac{1}{2} \left\{ r_n (t_n - w^T (\phi(x_n))^2 \right\} \rightarrow (2)$$

where weighing factor $r_n = \frac{\sigma_n}{\theta_n} \frac{\theta_n}{\theta_n} \frac{\theta_n}{\sigma_n^2}$ for n^{th} data point.

Hence the MLE estimate reduces to minimization of equation (2)

Hence sum of squares error function is

$$E_D(w) = \frac{1}{2} \sum_{n=1}^{n=N} r_n (t_n - w^T (\phi(x_n))^2 \rightarrow (3)$$

(ii) To minimize $E_D(w)$ we find $\nabla_w E_D(w)$

$$\nabla_w E_D w = 2 \cdot \frac{1}{2} \sum_{n=1}^{n=N} r_n (t_n - w^T \phi(x_n)) (-\phi(x_n))$$

$$= -\sum_{n=1}^{n=N} r_n t_n \phi(x_n) + \sum_{n=1}^{n=N} r_n w^T \phi(x_n) \cdot \phi(x_n).$$

putting $\nabla_w E_D(w) = 0$ we get

$$\sum_{n=1}^{n=N} r_n w^T \phi(x_n) \cdot \phi(x_n) = \sum_{n=1}^{N} r_n t_n \phi(x_n).$$

$$\Rightarrow \sum_{n=1}^{N} r_n \phi(x_n)^T w \phi(x_n) = \sum_{n=1}^{n=N} r_n t_n \phi(x_n).$$

$$\Rightarrow w = \frac{\sum_{n=1}^{n=N} r_n t_n \phi(x_n)}{\sum_{n=1}^{n=N} r_n \phi(x_n)^T \phi(x_n)} \rightarrow (4)$$

Hence eq (4) is the expression for w which minimizes the error function in equation (3).

3(a)(i).

For logistic regression likelihood function

$$L(w) = \sum_{n=1}^{n=N} \left\{ y_i \log\left(\frac{1}{1+\exp(-w^T x)}\right) + (1-y_i)\log\left(1 - \frac{1}{1+\exp(-w^T x)}\right) \right\}.$$

Gradient

(ii) $$\nabla_w L(w) = \sum_{n=1}^{N} \left\{ 1 - \frac{y_i}{1+\exp(-w^T x)} \right\} \cdot x$$

(iii) Hessian

$$\nabla_w^2 L(w) = \sum_{n=1}^{n=N} \left[\frac{-\exp(-w^T x)}{(1+\exp(-w^T x))^2} \right] \cdot x_i \cdot x_i^T.$$

(iv). update equation

$$w_{new} = w_{old} - \frac{\nabla_w L(w)}{\nabla_w^2 L(w)}$$

$$\Rightarrow w_{new} = w_{old} - \frac{\sum_{n=1}^{N} \left\{ 1 - \frac{y_i}{1+\exp(-w^T x)} \right\} x.}{\sum_{n=1}^{n=N} \left\{ \frac{-\exp(-w^T x)}{(1+\exp(-w^T x))^2} \right\} x_i \cdot x_i^T.}$$

3(a)

Algorithm of Newton-Raphson optimization technique

1. Initialize the parameter vector w
   convergence threshold $\epsilon = 0.001$
   Number of Iterations T_{max}
   change in weights $\Delta w = \infty$
   current iteration number $t = 0$.

2. while $t < T_{max}$ && $||\Delta w|| > \epsilon$:

   3.a compute $\nabla_w L(w) = \sum_{n=1}^{n=N} \left(y_i - \frac{1}{1+\exp(-w^T x)} \right)$

   &

   $$\nabla_w^2 L(w) = \sum_{n=1}^{n=N} \left(\frac{-\exp(-w^T x)}{(1+\exp(-w^T x))^2} \right) \cdot x_i \cdot x_i^T.$$

   3.b update weight by

   $$w_{new} = w_{old} - \frac{\nabla_w L(w)}{\nabla_w^2 L(w)}$$

   3.c $\Delta w = w_{new} - w_{old}$.

   3.d $w_{old} = w_{new}$.

   3.e $t = t + 1$.

3. Return w_{new}.

3(i). Weighted least Squares minimizes the following objective error function

$$E(w) = \frac{1}{2} \sum_{n=1}^{N} r_n (t_n - w^T(\phi(x_n))^2$$

with update equation

$$w_{new} = w_{old} - \theta_n (y\, t_n - w^T \phi(x_n)) \cdot x_n \quad \rightarrow (1)$$

where θ_n = weight associated with n^{th} data point

t_n = actual value of n^{th} data point

w = parameter to estimate.

Newton Raphson's method tries to minimize the error function by iteratively updating weights using second order derivative with update equation as

$$w_{new} = w_{old} - (\nabla_w^2 L(w))^{-1} (\nabla_w L(w)) \rightarrow (2)$$

equation (2) is analogous to equation (1) as

$\nabla_w^2 L(w)^{-1}$ i.e Hessian^{-1} in Newton Raphson's method corresponds to weight matrix in WLS.

$\nabla_w L(w)$ = gradient in Newton Raphson's method corresponds to error term i.e residual in WLS.

(ii). Newton Raphson's method iteratively reweights data points by updating the weight vector in each iteration.

Due to this reweighing of data points, it is also called iterative least squares method.

(c) Error function of logistic regression

$$E(w) = -\sum_{n=1}^{N} \{ t_n \log a_n + (1-t_n) \log (1-a_n) \}.$$

where $a_n = \sigma(w^T x)$.

gradient of $E(w)$

$$\nabla_w E(w) = \sum_{n=1}^{N} (a_n - t_n) x_n = X^T (a - t)$$

$$\nabla_w^2 E(w) = \sum_{n=1}^{N} a_n (1-a_n) x_n \cdot x_n^T$$

$$= X R X^T \longrightarrow (1)$$

where R_{nxn} is a diagonal matrix of $N \times N$ elements i.e $R_{nxn} = a_n(1-a_n)$ for n^{th} row.

from equation (1)

$$H = x_n R x_n^T$$

Multiplying by v^T & v from both sides in above equation, where v is a random vector such that $v \neq 0$.

$$\Rightarrow v^T H v = v^T x_n R x_n^T v$$

$$= v^T x_n R (v^T x_n)^T$$

$$= \| v^T x_n R^{1/2} \|^2$$

$$\Rightarrow v^T H v > 0$$

Therefor H is positive definite

{using Theorem:- If $xAx > 0$ for a random non zero vector x, then A is +ve definite}.

⇒ Error function $E(w)$ is a convex function

⇒ There exist a unique solution. i.e unique minima.