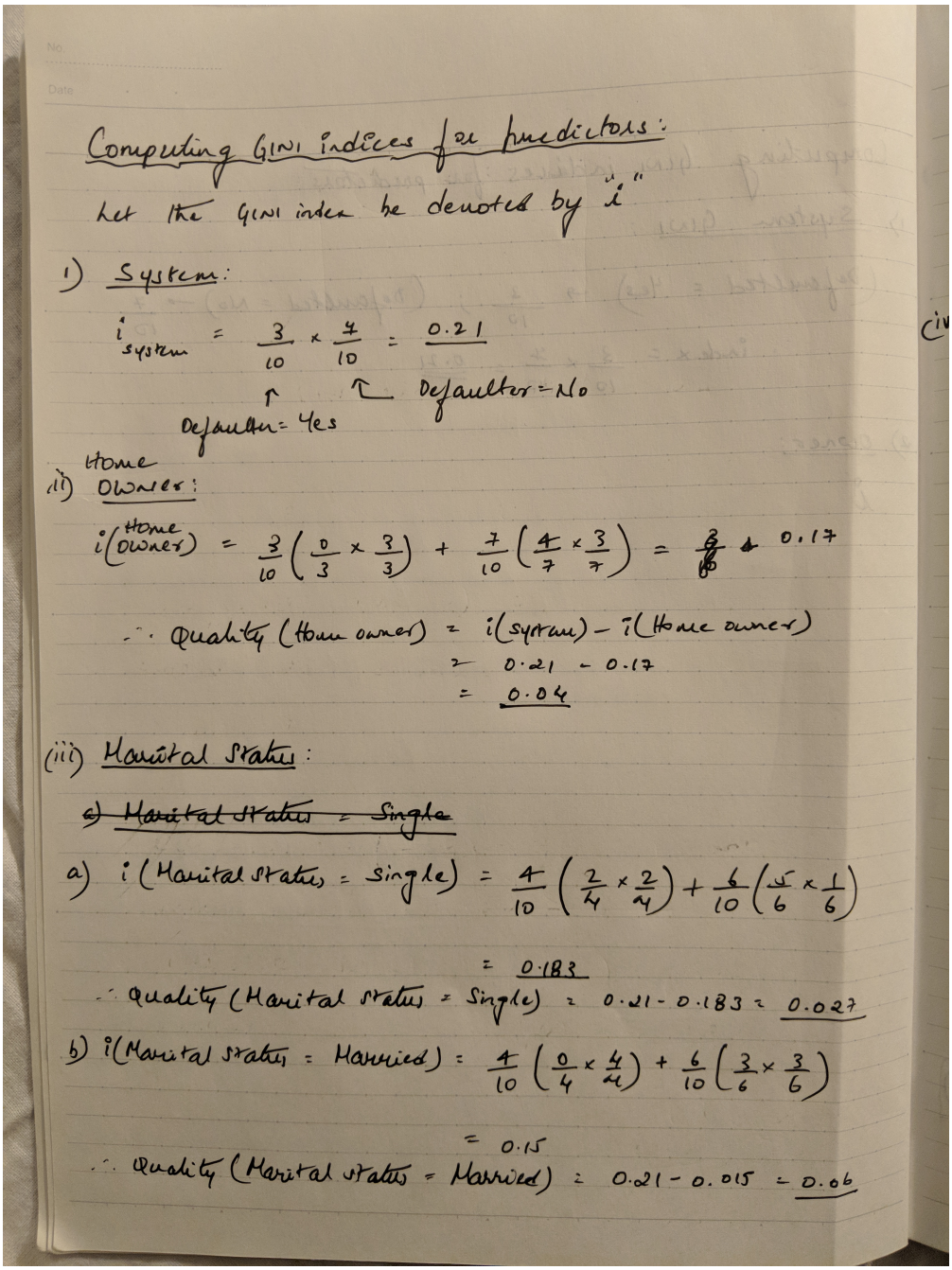
Solutions to Homework 4

Nakul Camasamudram

10/18/2017

Problem 1

a.



c) ? (Marital status = Divorced) $= \frac{2}{10}\left(\frac{1}{2}\times\frac{1}{2}\right) + \frac{8}{10}\left(\frac{6}{8}\times\frac{2}{8}\right)$

$= 0.20$

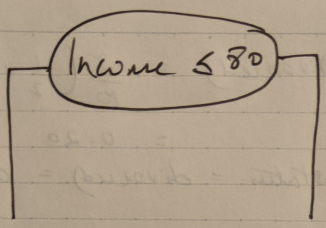
$\therefore$ Quality (Marital status = divorced) $= 0.21 - 0.20 = \underline{0.01}$

(ii) Income:

Income	Quality (split after) 0.21 −
60	$\frac{1}{10}\left(\frac{1}{1}\times\frac{0}{1}\right) + \frac{9}{10}\left(\frac{6}{9}\times\frac{3}{9}\right) = 0.01$
70	$\frac{2}{10}\left(\frac{2}{2}\times\frac{0}{2}\right) + \frac{8}{10}\left(\frac{5}{8}\times\frac{3}{8}\right) = 0.0225$
75	$\frac{3}{10}\left(\frac{3}{3}\times\frac{0}{3}\right) + \frac{7}{10}\left(\frac{3}{7}\times\frac{4}{7}\right) = \boxed{0.038}$
85	$\frac{4}{10}\left(\frac{3}{4}\times\frac{1}{4}\right) + \frac{6}{10}\left(\frac{2}{6}\times\frac{4}{6}\right) = 0.0016$
90	$\frac{5}{10}\left(\frac{3}{5}\times\frac{2}{5}\right) + \frac{5}{10}\left(\frac{1}{5}\times\frac{4}{5}\right) = 0.01$
95	$\frac{6}{10}\left(\frac{3}{6}\times\frac{3}{6}\right) + \frac{4}{10}\left(\frac{0}{4}\times\frac{4}{4}\right) = 0.06$
100	$\frac{7}{10}\left(\frac{4}{7}\times\frac{3}{7}\right) + \frac{3}{10}\left(\frac{3}{3}\times\frac{0}{3}\right) = 0.038$
120	$\frac{8}{10}\left(\frac{5}{8}\times\frac{3}{8}\right) + \frac{2}{10}\left(\frac{2}{2}\times\frac{0}{2}\right) = 0.022$
125	$\frac{9}{10}\left(\frac{6}{9}\times\frac{3}{9}\right) + \frac{1}{10}\left(\frac{1}{1}\times\frac{0}{1}\right) = 0.01$
220	$\frac{10}{10}\left(\frac{7}{10}+\frac{3}{10}\right) = 0.00$

Best overall GINI index is income in the range 75 − 85. By taking the midpoint of this range, we get

$\boxed{\text{income} \leq 80}$



Now, Quality has to be recomputed again for the left and right subtrees above.

LEFT:

$i(\text{Home Owner}) = \frac{3}{3}\left(\frac{0}{3} \times \frac{3}{3}\right) = 0$

$i(\text{Marital Status} = \text{Married}) = 0$

$i(\text{Marital Status} = \text{Single}) = 0$

$i(\text{Income}) = 0$

RIGHT:

(i) $i(\text{Home owner}) = \frac{3}{7}\left(\frac{3}{3} \times \frac{0}{3}\right) + \frac{4}{7}\left(\frac{3}{4} \times \frac{1}{4}\right) = \frac{3}{28}$

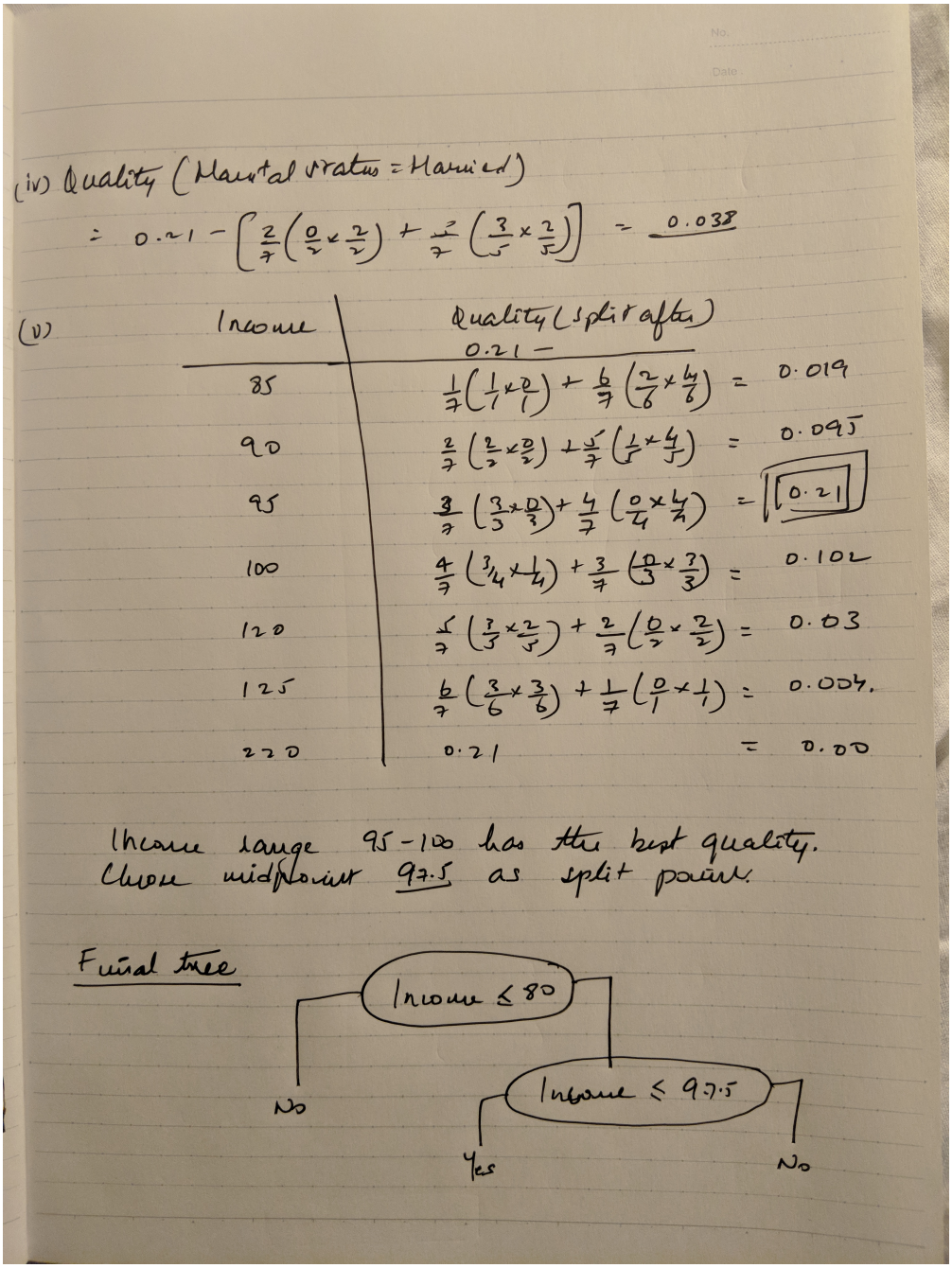
$\text{Quality}(\text{Home owner}) = 0.21 - \frac{3}{28} = 0.102$

(ii) $i(\text{Marital Status} = \text{Single}) = \frac{3}{7}\left(\frac{2}{3} \times \frac{1}{3}\right) + \frac{4}{7}\left(\frac{3}{4} \times \frac{1}{4}\right) = 0.202$

$\text{Quality}(\text{Marital Status} = \text{Single}) = 0.007$

(iii) $\text{Quality}(\text{Marital Status} = \text{Divorced}) = 0.21 - \left[\frac{2}{7}\left(\frac{1}{2} \times \frac{1}{2}\right) + \frac{5}{7}\left(\frac{2}{5} \times \frac{3}{5}\right)\right]$

$= -0.03$



c

Below is the pseudocode to find the optimal subtree

- Construct a full tree on the full dataset
- Compute $\{\alpha_1, \alpha_2, \ldots\}$
- Obtain $\{T_1, T_2, \ldots, t\}$ for each α
- Define sequence $\{\beta_j = \sqrt{\alpha_{j-1} \cdot \alpha_j}\}$
- Divide data into folds
- Within each fold:
  - Build a sequence of trees over β_j
  - Compute prediction error for left-out obs.
- For each β_j, sum # misclassifications over folds
- Select β_j with min # mis-classifications (option: up to one SE)
- Report the β_j sub-tree of the full data

Problem 2

Data Initialization

```r
# Imports
library(dplyr)

rm(list=ls())
set.seed(123)

# Read the data
saheart <- read.table("SAheart.txt", sep = ",", header = TRUE) %>% select(-row.names)

# Separate input and output
X <- saheart[-c(5, 10)]
X <- as.matrix(X/max(X)) # Normalized matrix for faster convergence
y <- saheart$chd
```

Use existing logistic regression implementation

```r
logreg.fit <- glm(y ~ X, family = binomial(link = "logit"))

# Parameters
coef(logreg.fit)
```

```
> (Intercept)        Xsbp    Xtobacco        Xldl  Xadiposity      Xtypea
>  -6.0668643   1.2297098  15.8519799  41.9631907   3.7204906   8.8218215
>    Xobesity    Xalcohol        Xage
> -12.6290125   0.3151876  11.0417720
```

```r
# Prediction accuracy
logreg.pred <- predict(logreg.fit, newdata = saheart[-c(5, 10)], type='response')
logreg.pred <- ifelse(logreg.pred > 0.5, 1, 0)
acc <- 1 - mean(logreg.pred != saheart$chd)
acc
```

```
> [1] 0.7229437
```

Perceptron as logistic regression

```r
# Sigmoid activation
sigmoid <- function(z){1.0/(1.0+exp(-z))}

# Perceptron
perceptron <- function(X, y, lr)
{
  delta <- 1e-8  # Allowed difference between consecutive parameter estiamtes
  X_mat <- cbind(1, X)
  w <- matrix(0, nrow=ncol(X_mat))  # Initialize weights as zeros
  w_prev <- matrix(1, nrow=ncol(X_mat))
  w_diff <- abs(w - w_prev)

  # Batch gradient descent
  while(length(w_diff[w_diff > delta]) > 1)  # Until convergence
  {
    residual <- sigmoid(X_mat %*% w) - y

    del <- t(X_mat) %*% as.matrix(residual, ncol=nrow(X_mat)) *  (1/nrow(X_mat))  # Derivative

    w_prev <- w
    w <- w - (lr*del)  # New weights
    w_diff <- abs(w - w_prev)
  }

  return(w)
}

weights <- perceptron(X, y, 5)

# Parameters
weights
```

```
>                 [,1]
>           -6.0668514
> sbp        1.2297108
> tobacco   15.8519876
> ldl       41.9621180
> adiposity  3.7206486
> typea      8.8218265
> obesity  -12.6290528
> alcohol    0.3151795
> age       11.0417533
```

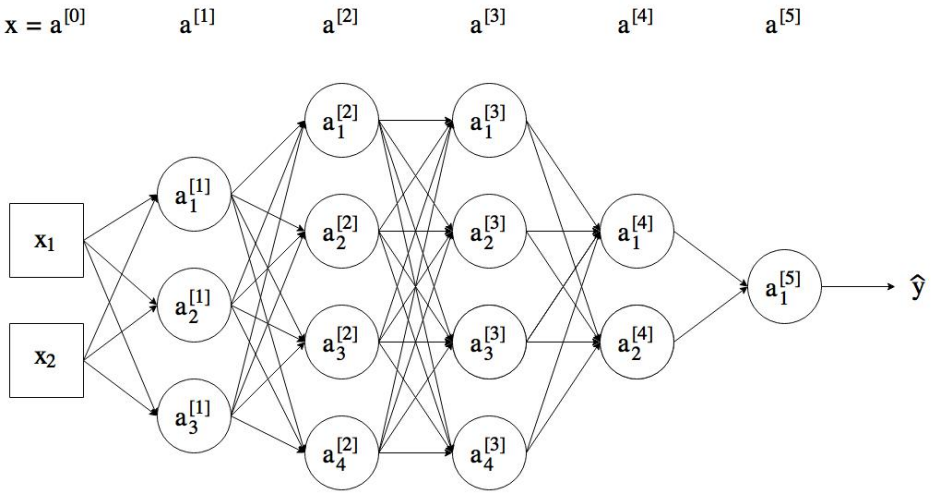
```
X <- as.matrix(saheart[-c(5, 10)])
X <- cbind(1, X)
result <- sigmoid(X %*% weights)
result <- ifelse(result > 0.5, 1, 0)
acc <- 1 - mean(logreg.pred != saheart$chd)

# Prediction accuracy
acc
```

```
> [1] 0.7229437
```

Both the methods above have the exact same coefficients and prediction accuracy.

Problem 3

a.



- Superscripts and subscripts refer to the layers and activation units respectively. For example, $a_2^{[1]}$ refers to the second unit of layer 1.
- When the subscript is not present, the term is a vector that represents all the activation units of a layer. For example, $a^{[1]} = [a_1^{[1]}\ a_2^{[2]}\ a_3^{[3]}]^T$.

b.

Number of parameters = (3 x 3) + (4 x 4) + (4 x 5) + (2 x 5) + (1 x 3) = 9 + 16 + 20 + 10 + 3 = 58

c.

Let's introduce a few notations so that the below equations are better understandable.

- $z^{[l]}$ is a column vector of length n_l where n_l is the number of activation units in the l^{th} layer. It is a weighted sum of the activation outputs of layer l - 1 and a bias term $b^{[l-1]}$
- $b^{[l]}$ is a column vector of length n_l that represents the bias input to l^{th} activation layer.
- $a^{[l]}$ is the activation vector of length n_l such that $a^{[l]} = \sigma(z^{[l]})$ where σ, the activation function, is applied element-wise on $z^{[l]}$.

- $W^{[l]}$ is a matrix whose elements $W_{ij}^{[l]}$ denote the weight associated with the connection between unit j in layer $l-1$, and unit i of layer l.
- The dimensions of various terms below are represented by their subscripts. Example, $z_{(m\times n)}^{[l]}$ has m rows and n columns.

Layer 1:

$$z_{(3\times1)}^{[1]} = W_{(3\times2)}^{[1]} a_{(2\times1)}^{[0]} + b_{(3\times1)}^{[1]}$$
$$a_{(3\times1)}^{[1]} = \sigma(z_{(3\times1)}^{[1]})$$

Layer 2:

$$z_{(4\times1)}^{[2]} = W_{(4\times3)}^{[2]} a_{(3\times1)}^{[1]} + b_{(4\times1)}^{[2]}$$
$$a_{(4\times1)}^{[2]} = \sigma(z_{(4\times1)}^{[2]})$$

Layer 3:

$$z_{(4\times1)}^{[3]} = W_{(4\times4)}^{[3]} a_{(4\times1)}^{[2]} + b_{(4\times1)}^{[3]}$$
$$a_{(4\times1)}^{[3]} = \sigma(z_{(4\times1)}^{[3]})$$

Layer 4:

$$z_{(2\times1)}^{[4]} = W_{(2\times4)}^{[4]} a_{(4\times1)}^{[3]} + b_{(2\times1)}^{[4]}$$
$$a_{(2\times1)}^{[4]} = \sigma(z_{(2\times1)}^{[4]})$$

Layer 5:

$$z_{(1\times1)}^{[5]} = W_{(1\times2)}^{[5]} a_{(2\times1)}^{[4]} + b_{(1\times1)}^{[5]}$$
$$a_{(1\times1)}^{[5]} = \sigma(z_{(1\times1)}^{[5]})$$
$$\hat{y}_{(1\times1)} = a_{(1\times1)}^{[5]}$$

d.

To represent multiple observations, a couple of extra notations have to be introduced. The below notations assume that there are m training examples.

- $A^{[l]} = [a^{[l](1)}\ a^{[l](2)}\ a^{[l](3)} \cdots a^{[l](m)}]$ represents the l^{th} activation layer output and column vector $a^{[l](i)}$ is the l^{th} activation layer output for the i^{th} training example.
- $Z^{[l]} = [z^{[l](1)}\ z^{[l](2)}\ z^{[l](3)} \cdots z^{[l](m)}]$ is the input to the l^{th} activation layer and $z^{[l](i)}$ is the input to the to the l^{th} activation layer for the i^{th} training example.
- $B^{[l]} = [b^{[l](1)}\ b^{[l](2)}\ b^{[l](3)} \cdots b^{[l](m)}]$ is a matrix where the i^{th} column is the bias input to the l^{th} layer for the i^{th} training example.

Layer 1:

$$Z_{(3\times m)}^{[1]} = W_{(3\times2)}^{[1]} A_{(2\times m)}^{[0]} + b_{(3\times m)}^{[1]}$$
$$A_{(3\times m)}^{[1]} = \sigma(Z_{(3\times m)}^{[1]})$$

Layer 2:

$$Z_{(4\times m)}^{[2]} = W_{(4\times3)}^{[2]} A_{(3\times m)}^{[1]} + B_{(4\times m)}^{[2]}$$
$$A_{(4\times m)}^{[2]} = \sigma(Z_{(4\times m)}^{[2]})$$

Layer 3:

$$Z^{[3]}_{(4\times m)} = W^{[3]}_{(4\times 4)} A^{[2]}_{(4\times m)} + B^{[3]}_{(4\times m)}$$
$$A^{[3]}_{(4\times m)} = \sigma(Z^{[3]}_{(4\times m)})$$

Layer 4:

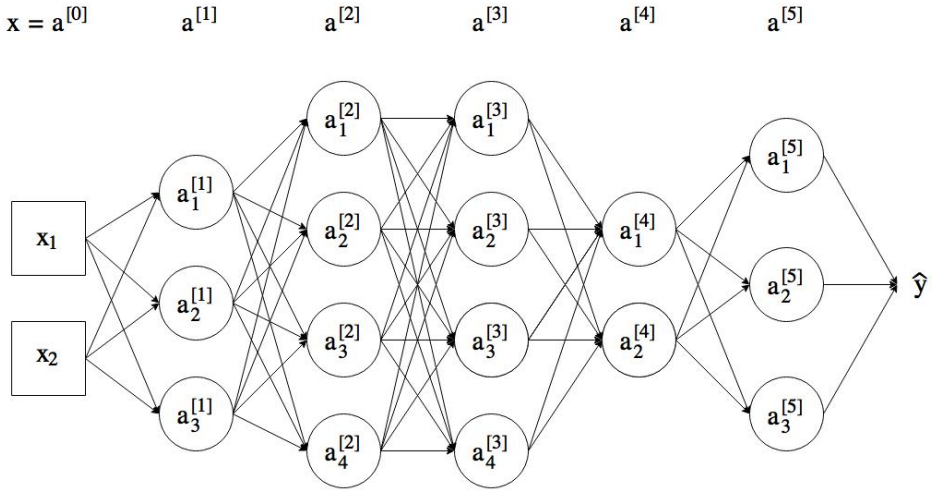
$$Z^{[4]}_{(2\times m)} = W^{[4]}_{(2\times 4)} A^{[3]}_{(4\times m)} + B^{[4]}_{(2\times m)}$$
$$A^{[4]}_{(2\times m)} = \sigma(Z^{[4]}_{(2\times m)})$$

Layer 5:

$$Z^{[5]}_{(1\times m)} = W^{[5]}_{(1\times 2)} A^{[4]}_{(2\times m)} + B^{[5]}_{(1\times m)}$$
$$A^{[5]}_{(1\times m)} = \sigma(Z^{[5]}_{(1\times m)})$$
$$\hat{y}_{(1\times m)} = A^{[5]}_{(1\times m)}$$

e.

a.



The notations used here will be consistent with those used in the above subproblems.

b.

Number of parameters = (3 x 3) + (4 x 4) + (4 x 5) + (2 x 5) + (3 x 3) = 9 + 16 + 20 + 10 + 9 = 64

c.

Layer 1:

$$z^{[1]}_{(3\times 1)} = W^{[1]}_{(3\times 2)} a^{[0]}_{(2\times 1)} + b^{[1]}_{(3\times 1)}$$
$$a^{[1]}_{(3\times 1)} = \sigma(z^{[1]}_{(3\times 1)})$$

Layer 2:

$$z^{[2]}_{(4\times1)} = W^{[2]}_{(4\times3)} a^{[1]}_{(3\times1)} + b^{[2]}_{(4\times1)}$$
$$a^{[2]}_{(4\times1)} = \sigma(z^{[2]}_{(4\times1)})$$

Layer 3:

$$z^{[3]}_{(4\times1)} = W^{[3]}_{(4\times4)} a^{[2]}_{(4\times1)} + b^{[3]}_{(4\times1)}$$
$$a^{[3]}_{(4\times1)} = \sigma(z^{[3]}_{(4\times1)})$$

Layer 4:

$$z^{[4]}_{(2\times1)} = W^{[4]}_{(2\times4)} a^{[3]}_{(4\times1)} + b^{[4]}_{(2\times1)}$$
$$a^{[4]}_{(2\times1)} = \sigma(z^{[4]}_{(2\times1)})$$

Layer 5:

$$z^{[5]}_{(3\times1)} = W^{[5]}_{(3\times2)} a^{[4]}_{(2\times1)} + b^{[5]}_{(3\times1)}$$
$$a^{[5]}_{(3\times1)} = \sigma(z^{[5]}_{(3\times1)})$$
$$\hat{y}_{(1\times1)} = max(a^{[5]}_1, a^{[5]}_2, a^{[5]}_3)$$

$a^{[5]}_i$ has the dimensions (1×1).

d.

Layer 1:

$$Z^{[1]}_{(3\times m)} = W^{[1]}_{(3\times2)} A^{[0]}_{(2\times m)} + B^{[1]}_{(3\times m)}$$
$$A^{[1]}_{(3\times m)} = \sigma(Z^{[1]}_{(3\times m)})$$

Layer 2:

$$Z^{[2]}_{(4\times m)} = W^{[2]}_{(4\times3)} A^{[1]}_{(3\times m)} + B^{[2]}_{(4\times m)}$$
$$A^{[2]}_{(4\times m)} = \sigma(Z^{[2]}_{(4\times m)})$$

Layer 3:

$$Z^{[3]}_{(4\times m)} = W^{[3]}_{(4\times4)} A^{[2]}_{(4\times m)} + B^{[3]}_{(4\times m)}$$
$$A^{[3]}_{(4\times m)} = \sigma(Z^{[3]}_{(4\times m)})$$

Layer 4:

$$Z^{[4]}_{(2\times m)} = W^{[4]}_{(2\times4)} A^{[3]}_{(4\times m)} + B^{[4]}_{(2\times m)}$$
$$A^{[4]}_{(2\times m)} = \sigma(Z^{[4]}_{(2\times m)})$$

Layer 5:

$$Z^{[5]}_{(3\times m)} = W^{[5]}_{(3\times2)} A^{[4]}_{(2\times m)} + B^{[5]}_{(3\times m)}$$
$$A^{[5]}_{(3\times m)} = \sigma(Z^{[5]}_{(3\times m)})$$
$$\hat{y}_{(1\times m)} = max(A^{[5]}_1, A^{[5]}_2, A^{[5]}_3)$$

$A^{[5]}_i$ has the dimensions $(1 \times m)$.